UNIVERSIDADE FEDERAL FLUMINENSE

CAMILA ELEUTÉRIO GUSMÃO

Explorando a Generalização de Classificadores de Notícias Falsas em Português Baseados em Modelos de Linguagem

NITERÓI

2024

CAMILA ELEUTÉRIO GUSMÃO

Explorando a Generalização de Classificadores de Notícias Falsas em Português Baseados em Modelos de Linguagem

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: CIÊNCIA DA COMPUTAÇÃO

Orientadora:
ALINE MARINS PAES CARVALHO

Co-orientadora:
THAIANE MOREIRA DE OLIVEIRA

NITERÓI

2024

Ficha catalográfica automática - SDC/BEE
Gerada com informações fornecidas pelo autor

G982e Gusmão, Camila Eleutério
Explorando a Generalização de Classificadores de Notícias Falsas em Português Baseados em Modelos de Linguagem / Camila Eleutério Gusmão. - 2024.
121 f.: il.

Orientador: Aline Marins Paes Carvalho.
Coorientador: Thaiane Moreira de Oliveira.
Dissertação (mestrado)-Universidade Federal Fluminense, Instituto de Computação, Niterói, 2024.

1. Aprendizado de máquina. 2. Processamento de linguagem natural. 3. Fake news. 4. Generalização de modelos de linguagem. 5. Produção intelectual. I. Carvalho, Aline Marins Paes, orientadora. II. Oliveira, Thaiane Moreira de, coorientadora. III. Universidade Federal Fluminense. Instituto de Computação. IV. Título.

CDD - XXX

Bibliotecário responsável: Debora do Nascimento - CRB7/6368

CAMILA ELEUTÉRIO GUSMÃO

Explorando a Generalização de Classificadores de Notícias Falsas em Português Baseados em Modelos de Linguagem

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: CIÊNCIA DA COMPUTAÇÃO

Aprovada em Outubro de 2024.

BANCA EXAMINADORA

Profa. ALINE MARINS PAES CARVALHO - Orientadora, UFF

Profa. THAIANE MOREIRA DE OLIVEIRA - Coorientadora, UFF

Profa. FLAVIA CRISTINA BERNARDINI, UFF

Profa. ANA PAULA COUTO DA SILVA, UFMG

Prof. RONALDO RIBEIRO GOLDSCHMIDT, IME

Niterói
2024

À Nana e Quinho, os gatos mais dóceis deste mundo

À André e Cecilia, que partiram cedo demais, o meu abraço apertado

Agradecimentos

Muitos caminharam para que este momento chegasse. Desde imigrantes de aldeias pobres até meus pais, no começo de tudo, lavradores de café em solo capixaba. Vieram para o Rio de Janeiro dentre outras razões por uma chance que não dependesse apenas da terra, da chuva, e porque não, da sorte. Mesmo sem entenderem o que era a vida acadêmica, sempre cultivaram em mim, da maneira deles, a ideia de que sem estudar eu não teria chance alguma de mudar a minha realidade.

O desejo pelo aprendizado grudou feito carrapicho, daqueles que grudam na perna da gente quando corremos no mato. Se instalou, e nunca mais saiu. Porém eles, meus pais, não foram os únicos incentivadores. Esta vontade somente se manteve viva, dando origem ao sonho de lecionar um dia - mesmo com toda a propaganda contrária existente - porque tive em meu caminho excelentes professores.

Aqui faço um agradecimento especial aos professores que gentilmente fizeram parte desta caminhada, cada um com o seu estilo de trabalhar e de se comunicar com os alunos. Vocês foram em muitos momentos um reduto acolhedor, uma fonte de inspiração e de esperança; esta conquista não seria possível sem vocês.

Não posso deixar de citar a minha orientadora Aline, que aceitou me orientar e me deu um voto de confiança para conduzir a pesquisa deste tema tão desafiador que são as *fake news*. Os seus ensinamentos e revisões foram cruciais para a qualidade do que foi apresentado. Estendo este agradecimento à banca de avaliação, por todos os comentários e sugestões gentilmente ofertados, refinando esta dissertação.

Agradeço também aos meus amigos, familiares, e ao meu namorado Renato, por toda a compreensão nos momentos em que estive ausente para me dedicar a este projeto, além de todas as palavras de encorajamento quando eu duvidava que seria possível chegar ao final.

Por fim, agradeço à Deus por não desistir desta filha e colocar tantas pessoas incríveis no meu caminho e tornar este sonho algo possível e, sobretudo, real.

Resumo

A pesquisa sobre a geração automática de classificadores de notícias falsas tem sido amplamente investigada para combater a desinformação global, amplificada pela Internet. No entanto, a maioria dos estudos se concentra na língua inglesa. Apesar dos classificadores mostrarem boa generalização em experimentos empíricos com uma mesma base de dados sendo usada para treinamento e avaliação, sua eficácia no mundo real, com notícias de variados assuntos e estilos, ainda é incerta. Além disso, não está claro quais características das bases de dados contribuem para melhores classificadores. Esta dissertação investiga a capacidade de generalização de classificadores baseados em modelos de linguagem na detecção de notícias falsas escritas em português. Para tanto, foram selecionados modelos de linguagem monolíngue e multilíngue incorporando variações da Arquitetura Transformer. Foram selecionadas 14 bases de dados em português, onde foram observadas características que podem influenciar o aprendizado, como estilo de escrita, assuntos, taxa de balanceamento e padrão na rotulação. Para investigar a generalização dos classificadores em diferentes bases, propomos duas estratégias: (i.) a generalização pós-treinamento dos classificadores, quando os modelos são testados com uma base diferente da que foram treinados (*cross-data*) e (ii.) a generalização do modelo de linguagem pré-treinado, com experimentos no estilo (*zero-shot*) que remontam à tarefa intermediária de completação de textos. Os experimentos *zero-shot* ainda não se mostram capazes de classificar notícias falsas, porém o uso de LLMs como o Sabiá-3 podem auxiliar na distinção de textos com alegações verificáveis de sentenças de opinião. Já nos experimentos *cross-data*, dentre os *encoders* os classificadores baseados no BERTimbau obtiveram os melhores resultados, com dez deles atingindo F1 macro igual ou superior a 70% na avaliação de pelo menos uma base não utilizada no treinamento. O modelo mT5 gerou os classificadores com o maior alcance sobre dados não vistos durante o treinamento, porém necessitam de uma oferta maior de exemplos para obterem bons resultados.

Palavras-chave: desinformação, modelos de linguagem, generalização de modelos, inferência de modelos.

Abstract

Research into the automatic generation of fake news classifiers has been widely investigated to combat global disinformation, amplified by the Internet. However, most studies focus on the English language. Although the classifiers show good generalization in empirical experiments with the same dataset used for training and evaluation, their effectiveness in the real world, with news on various topics and styles, is still uncertain. Furthermore, it is unclear which characteristics of the datasets contribute to better classifiers. This article investigates the generalization capacity of classifiers based on language models in the detection of fake news written in Portuguese. For this purpose, monolingual and multilingual language models incorporating variations of the Transformer architecture were selected. 14 Portuguese data sets were chosen, in which characteristics that could influence learning were observed, such as writing style, topics, balance rate and labeling pattern. To investigate the generalization of classifiers on different datasets, we propose two strategies: (i) post-training generalization of classifiers, in which the models are tested on a different dataset to the one on which they were trained (cross-data), and (ii) generalization of the pre-trained language model, with zero-shot style experiments that resemble the intermediate task. The *zero-shot* experiments have not yet proven capable of classifying fake news, but the use of LLMs such as Sabiá-3 can help distinguish texts with verifiable claims from opinion statements. In the *cross-data* experiments, among the *encoders* the classifiers based on BERTimbau obtained the best results, with ten of them achieving F1 macro equal to or greater than 70% in the evaluation of at least one base not used in training. The mT5 model generated the classifiers with the greatest reach on data not seen during training, but they need a larger supply of examples to achieve great results.

Keywords: fake news, misinformation, language models, zero-shot, datasets cross-data validation.

Lista de Figuras

Lista de Tabelas

Sumário

1 Introdução

A circulação de informação falsa não é algo novo, com o registro de inúmeros episódios ao longo da história da humanidade (MAGAZINE, 2016; POSETTI; MATTHEWS, 2018). Um dos casos mais antigos e famosos que se tem notícia remonta ao início do século XIX, quando o jornal norte-americano *The Sun* publicou uma série de artigos que não só afirmavam a existência de uma civilização na Lua, como também davam detalhes sobre as criaturas que lá habitavam, o que ficou conhecido como *The Great Moon Hoax* (THORNTON, 2000). Entretanto, a disseminação de notícias falsas nos moldes atuais, as chamadas "fake news", não é comparável aos seus precedentes históricos. Se no passado elas se limitavam basicamente a rumores locais e estórias sensacionalistas na imprensa (através da chamada *yellow press*[1]), o alcance e a velocidade de propagação das redes sociais as tornaram muito mais nocivas para a sociedade (OLAN et al., 2024).

Anteriormente, quando notícias eram publicadas apenas por meio de veículos de comunicação, a credibilidade de uma notícia era avaliada com base na reputação de quem a publicou e no cumprimento de padrões jornalísticos estabelecidos, com a apuração dos fatos e uma distinção clara entre informação e opinião (O'NEIL; GEDDES, 2015; VALENTINI; DAMASIO, 2016). Porém, com o advento das redes sociais, qualquer pessoa pode publicar e disseminar informações sem a obrigatoriedade do emprego destes filtros, comprometendo a qualidade e a veracidade dos conteúdos compartilhados (COOKE, 2017). Isto se reflete em uma maior dificuldade dos leitores em distinguir informações seguras das duvidosas, porque elas passaram a se apresentar de maneira bastante similar (CENTER FOR INFORMATION TECHNOLOGY AND SOCIETY, 2024).

Há várias pesquisas que buscam compreender por que as pessoas se tornam canais involuntários na disseminação de notícias falsas, com a investigação dos aspectos psicológicos e cognitivos em conjunto com a dinâmica das redes sociais (ECKER et al., 2022; OLAN et al., 2024; PENNYCOOK; RAND, 2021; COOKE, 2017). Em Cooke (2017), os pesquisadores trazem a preocupação do jornalismo em combater as *fake news* após as

[1] https://www.britannica.com/topic/yellow-journalism

eleições estadunidenses de 2016, contra o que foi chamado de **era "pós-verdade"** (em inglês *post-truth*, eleita como a palavra daquele ano[2]), na qual o público tende a acreditar mais em informações que apelam às emoções ou crenças pessoais do que em informações factuais ou objetivas, dificultando assim o combate às *fake news*. Por exemplo, o impacto do **viés de confirmação** é amplificado com a criação das chamadas "bolhas sociais", no qual os usuários tendem a buscar e acreditar em informações que confirmam suas próprias visões de mundo, ignorando fontes conflitantes, justamente por se cercarem apenas de conteúdos e perfis que reforcem estas ideias (COOKE, 2017).

Além da pré-existente inclinação subjetiva, a sobrecarga de informações disponíveis também contribui com este cenário. Em Case e Given (2016 apud COOKE, 2017), ressalta-se que "conforme o número de itens de informação aumenta — ou conforme a quantidade de tempo disponível diminui — as pessoas recorrem a regras mais simples e menos confiáveis para fazer escolhas para encurtar seu tempo de pesquisa" (tradução nossa).

Embora este fenômeno se dê essencialmente nas redes sociais, as consequências para o mundo *offline* são catastróficas e, em muitos casos, irreversíveis. Em 2018, a versão editada de um vídeo de uma campanha paquistanesa de combate ao sequestro de crianças "viralizou" na Índia. O conteúdo adulterado foi interpretado como evidência de um sequestro real, levando à morte de inocentes julgados como autores de um ato inventado[3]. No Brasil, um caso similar e de grande repercussão foi o linchamento de Fabiane Maria de Jesus em maio de 2014, após ser confundida com uma suposta sequestradora de crianças, cujo retrato falado havia sido feito dois anos antes e circulava em postagens no Facebook na época do crime[4].

O dano causado pelas *fake news* também ocorre de maneira coletiva, tendo como principais alvos a saúde e a política. A profusão de notícias falsas durante a pandemia de Covid-19 levou à morte milhares de pessoas que se recusaram a vacinar ou adotaram tratamentos comprovadamente ineficazes[5]. As *fake news* descredibilizando o resultado de eleições instigaram ataques à democracia, gerando episódios como a invasão ao Capitólio em 6 de janeiro de 2021[6] e a invasão às sedes do poder brasileiro em 8 de janeiro de 2023[7].

[2] https://languages.oup.com/word-of-the-year/2016/
[3] https://tinyurl.com/whatsapp-india-killings
[4] https://tinyurl.com/mulher-linchada-apos-fake-news
[5] https://www.buzzfeednews.com/article/janelytvynenko/coronavirus-fake-news-disinformation-rumors-hoaxes
[6] https://www.cnnbrasil.com.br/internacional/invasao-ao-capitolio-completa-um-ano-relembre-o-ataque-a-democracia-dos-eua/
[7] https://www.bbc.com/portuguese/articles/cye7egj6y1no

Existem muitas iniciativas no jornalismo dedicadas a combater as *fake news*, apurando fatos e mostrando incoerências. Porém, esta é uma corrida desleal, uma vez que os disseminadores de notícias falsas não seguem normas ou processos editoriais para garantir a precisão e a credibilidade das informações, o que os dá uma ampla vantagem na escala de divulgação (LAZER et al., 2018). Soma-se a isso a existência cada vez mais preponderante de "social bots" que simulam ações de usuários em publicações para engajar conteúdos de modo que eles alcancem um contingente maior de usuários reais (VAROL et al., 2017).

1.1 Problema de pesquisa

Métodos de detecção automática de informações falsas baseados em Inteligência Artificial (IA) são essenciais como uma tentativa de equilibrar o ecossistema de notícias, dada a escala de conteúdo que é produzido e propagado a cada instante nas redes sociais e nos mais diferentes formatos (AIMEUR; AMRI; BRASSARD, 2023; SHARMA et al., 2019). Contudo, a construção destes métodos ainda enfrenta percalços que os impedem de serem adotados no mundo real para impedir a propagação das notícias falsas (WĘCEL et al., 2023; PAWLICKA et al., 2024).

Se o treinamento ocorre com dados desatualizados, o modelo se torna defasado para lidar com novas informações e formas de se comunicar (HAMED; AB AZIZ; YAAKUB, 2023). Além disso, se os dados não abordarem diferentes tópicos, o modelo acaba se especializando na detecção de teor falso apenas em conteúdos de domínios específicos. Ademais, a maioria dos métodos requer dados de treinamento contendo notícias rotuladas como verdadeiras ou falsas, para induzirem métodos que as discriminem.

O processo de rotulação manual não é escalável, enquanto métodos que automaticamente coletam notícias verdadeiras ou falsas podem estar sujeitos a erros, ou conter algum tipo de viés. Estes pontos mostram que os classificadores de notícias falsas podem enfrentar dificuldades para serem *generalizáveis* ao ponto de classificarem corretamente uma notícia pertencente a uma distribuição distinta daquelas em que foram treinados.

Esta suposição, se comprovada verdadeira, dificultaria a adoção de classificadores automáticos para classificar notícias do mundo real, que podem variar em estilo de escrita e assunto das bases de dados utilizadas para treiná-los. Por outro lado, algumas bases de dados podem ter passado por um processo de curadoria mais cuidadoso, o que pode influenciar na qualidade dos classificadores. Entretanto, avaliá-los no mesmo conjunto de dados, ainda que este seja um conjunto de teste, não esclarece a influência da qualidade

da base no desempenho do classificador.

A dificuldade de generalização naturalmente também se estende ao idioma: a maioria das bases de dados e, por consequência, dos classificadores de *fake news* têm como foco a língua inglesa (SILVA, R. M. et al., 2020; FISCHER et al., 2022). Contudo, existem mais de 7.000 outros idiomas no mundo, que estão sendo negligenciados pela supremacia das bases de dados na língua inglesa (RUDER, 2020), o que agrava potencialmente a divulgação de notícias falsas ao redor do mundo. A língua portuguesa, em particular, é o quarto idioma mais usado no mundo[8].

O Brasil, por exemplo, é um país de proporções continentais e enfrenta muitos problemas com o avanço das notícias falsas, como os exemplos já mencionados aqui evidenciam. Assim, construir métodos automáticos que possam ser usados no mundo real pode ser um desafio ainda maior para a língua portuguesa, se comparado aos métodos que têm como foco línguas com maior disponibilidade de dados, como a inglesa.

Atualmente, a maioria dos métodos automáticos de detecção de notícias falsas são construídos a partir de modelos de linguagem neurais. Embora em menor quantidade que para a língua inglesa, existem modelos de linguagem pré-treinados tanto especificamente para português (SOUZA; NOGUEIRA; LOTUFO, 2020) como para múltiplas línguas que incluem o português (XUE et al., 2020). Entretanto, a capacidade de generalização de tais modelos para uma variedade de bases de *fake news* em português e a seleção do modelo de melhor desempenho para a tarefa foram pouco explorados na literatura.

1.2 Objetivos

Este trabalho contribui com uma investigação empírica do desempenho e das habilidades de generalização dos classificadores construídos a partir de modelos de linguagem neurais para a tarefa binária de classificação de *fake news* em português. O trabalho se concentra em tentar responder se tais classificadores poderiam ser usados no mundo real para dados em que eles não foram treinados, sobrepujando mudanças na distribuição dos dados (QUIÑONERO-CANDELA et al., 2022). Até onde sabemos, não há outros trabalhos na literatura contendo um levantamento de datasets de *fake news* em português.

Para tanto, a investigação seleciona modelos de linguagem neurais monolíngues e multilíngues com diferentes arquiteturas e experimenta os classificadores usando abordagens *in-data* – em que os classificadores são treinados e testados na mesma base de dados,

[8] https://tinyurl.com/lingua-portuguesa-influencias

apenas para efeito de sanidade dos resultados – e *cross-data* – em que os classificadores são testados em uma base diferente da que foram treinados.

1.3 Metodologia

Além da investigação da capacidade de generalização de classificadores mediante o aperfeiçoamento dos modelos de linguagem via *fine-tuning*, este trabalho verifica a capacidade de modelos de linguagem de diferentes arquiteturas em classificar *fake news* utilizando somente o conhecimento adquirido durante o pré-treinamento, por meio da abordagem *zero-shot*. Tais abordagens foram utilizadas na tentativa de responder as seguintes perguntas de pesquisa:

- **RQ1.** Que bases de dados existem para a classificação de notícias falsas em português e quais são as suas características principais que podem influenciar no desempenho dos classificadores? Para responder a esta pergunta, selecionamos as bases de dados disponibilizadas na literatura até o ano de 2022 e elencamos características de estilo de escrita, assuntos, classes e estratégias de rotulação, tamanho, diversidade léxica e semântica.

- **RQ2.** Generalização de modelos pré-treinados: modelos de linguagem pré-treinados em uma tarefa intermediária incluem como habilidade emergente a identificação da veracidade das notícias? Para responder a esta pergunta, propomos testar os modelos usando uma configuração que remonta à tarefa intermediária, ou seja, completar textos ou substituir máscaras em sentenças.

- **RQ3.** Generalização *in-data*: qual o desempenho dos classificadores de diferentes tipos de instâncias de Transformers ao serem treinados e testados com dados da mesma base de dados? Qual o papel das bases de dados no desempenho dos classificadores? Para responder a esta pergunta, foram treinados classificadores usando a estratégia de *fine-tuning*, e seus resultados foram examinados sob a luz das características das bases de dados.

- **RQ4.** Generalização *cross-data* de modelos treinados: qual o desempenho desses mesmos classificadores ao serem testados com dados de fora da sua base original, de forma similar a como poderiam ser usados no mundo real? Para responder a esta pergunta, os classificadores foram treinados com uma base D_i e testados com outra base $D_{j,j\neq i}$.

1.4 Contribuições

A partir da metodologia proposta para responder às questões de pesquisa apresentadas, o trabalho contribui com:

- Um catálogo e análise de vários conjuntos de dados sobre *fake news* na língua portuguesa, apresentando as suas características e mostrando quais são os pontos críticos observados acerca da qualidade dos dados.
- Metodologias de verificação de generalidade para testar classificadores e modelos de linguagem utilizados para identificar notícias falsas.
- Resultados empíricos que mostram o desempenho de classificadores com diferentes regimes de treinamento que lidam com textos de notícias falsas, indicando qual arquitetura, método de treinamento, ou processo de construção de base de dados melhor se adequa a este tipo de tarefa para o caso do português.

1.5 Organização do texto

O restante da dissertação está organizado como segue: o Capítulo 2 aborda os principais conceitos do ponto de vista teórico que deram o embasamento necessário para a elaboração e construção dos experimentos, já o Capítulo 3 indica trabalhos recentes que atuaram em problema correlatos e as relações com as nossas questões de pesquisa. O Capítulo 4 expõe a metodologia proposta para o estudo, o Capítulo 5 mostra as principais características dos datasets[9] selecionados a partir das análises efetuadas. O Capítulo 6 apresenta os resultados dos experimentos e o Capítulo 7 traz as conclusões, limitações e possibilidades de trabalhos futuros.

[9] Ao longo do texto usamos os termos dataset, conjunto de dados e bases de dados com o mesmo significado.

2 Fundamentação teórica

Neste capítulo são apresentados os conceitos necessários para a compreensão deste trabalho. Na Seção 2.1 será abordada a definição ampla do termo *fake news* e o papel da verificação de fatos. Em seguida, na Seção 2.2 serão apresentados os conceitos-base dos modelos e técnicas computacionais que proporcionaram o surgimento dos modelos aqui utilizados. Já na Seção 2.3, será introduzida a forma de treinamento inicial de tais modelos, bem como a prática de inferências a partir deles. Por fim, a última seção deste capítulo mostra como estes modelos podem ser utilizados para a classificação de notícias falsas.

2.1 Fake News e Verificação de Fatos

2.1.1 Definição de Fake News

O termo *fake news* se popularizou nas eleições estadunidenses de 2016[1], porém já existia há muito mais tempo, com registros que datam da década de 1890[2]. Embora ele não tenha surgido naquela disputa eleitoral, o seu significado e como passou a ser utilizado na comunicação mudou consideravelmente ao longo do tempo (SHARMA et al., 2019), o que gera discordâncias quanto a sua definição exata.

Consultando dois dos principais dicionários da língua inglesa (dado que é um verbete originário deste idioma), Collins e Cambridge, o primeiro define o termo como "informação falsa, em geral sensacionalista, divulgada sob o disfarce de reportagem" (tradução nossa)[3], enquanto o segundo o denomina como "histórias falsas que parecem ser notícias, divulgadas na Internet ou através de outros meios de comunicação, geralmente criadas para influenciar opiniões políticas ou como uma piada" (tradução nossa)[4]. No entanto,

[1] https://trends.google.com/trends/explore?date=2013-12-06%202018-01-06&geo=US&q=fake%20news

[2] https://www.merriam-webster.com/wordplay/the-real-story-of-fake-news

[3] https://www.collinsdictionary.com/dictionary/english/fake-news

[4] https://dictionary.cambridge.org/dictionary/english/fake-news

nenhuma das definições abrange tudo o que este termo representa nos dias de hoje, que virou sinônimo para a propagação de informações falsas (COOKE, 2017), como também começou a ser utilizado por alguns grupos para desacreditar notícias que os desagradam, mesmo comprovada a verificação das fontes (CITS - CENTER FOR INFORMATION TECHNOLOGY & SOCIETY, 2024; NAKOV, 2020). Procurando este mesmo verbete em dicionários da língua portuguesa, apenas o encontramos no Houaiss[5], que o define como **desinformação**, que de acordo com esta fonte significa "ação ou efeito de desinformar", que é um conceito muito amplo.

A primeira definição existente na literatura foi cunhada em 2017, que o declara como "artigos de notícias que sejam intencional e verificavelmente falsas, que possam enganar os leitores" (ALLCOTT; GENTZKOW, 2017)(tradução nossa). Posteriormente surgiram outras contribuições de caracterização do termo, que embora concordem sobre a não **autenticidade** do conteúdo, divergem em relação à abrangência de conceitos como *sátira*, *rumores*, *teorias da conspiração*, *misinformation* e *boatos* na definição proposta (AIMEUR; AMRI; BRASSARD, 2023). Além disso, as definições existentes enfrentam limitações quanto ao tipo de informação propagada ou a **intenção** de enganar, não capturando seu sentido mais amplo de utilização (SHARMA et al., 2019).

Existem outros termos relacionados à disseminação de informações falsas muito presentes na literatura, como *misinformation* e *disinformation*. Embora em português ambos sejam traduzidos como "desinformação", o termo *misinformation* é empregado quando uma informação falsa é propagada sem intenção de enganar, e *disinformation*, quando o disparo de conteúdos deste tipo são efetuados com o claro objetivo de enganar ou confundir quem os consome (AIMEUR; AMRI; BRASSARD, 2023).

Como a finalidade deste trabalho é a classificação de informação falsa, vamos nos ater à definição de *misinformation*, uma vez que não temos como saber de forma automática a intenção de quem propagou as notícias falsas pertencentes às bases de dados utilizadas neste estudo. Portanto, neste trabalho nos guiamos pela definição de *misinformation* de (ALLCOTT; GENTZKOW, 2017) para adaptá-la ao nosso cenário de estudo, caracterizando *fake news* como "toda e qualquer informação falsa que foi propagada por algum meio para atingir um número maior de pessoas, seja pela mídia tradicional ou através das redes sociais".

[5] https://houaiss.uol.com.br/

2.1.2 Verificação de Fatos

Uma das formas mais efetivas de combate às *fake news* é por meio do processo de checagem de fatos (OLIVEIRA et al., 2024). A checagem de fatos (ou *fact-checking*, em inglês), pode ser compreendida como a verificação de autenticidade de narrativas através do confrontamento com dados, pesquisas e registros, desempenhada por jornalistas especializados (FONSECA, 2017; FATOS, 2024).

A apuração dos fatos sempre existiu no jornalismo como princípio, porém tal prática ocorria antes da publicação das notícias, chamada de checagem de fatos interna ou *ante hoc*. Contudo, a checagem de fatos ganhou um novo sentido nas últimas duas décadas, envolvendo verificar a precisão de declarações de figuras públicas ou de textos (e outros conteúdos) já em circulação, tendo como resultado a publicação da análise efetuada e a indicação das evidências consideradas (GRAVES; AMAZEEN, 2019).

Com o dinamismo da internet, a checagem *ante hoc* foi relegada a segundo plano, seja pela escassez de recursos nas redações tradicionais ou à necessidade de coberturas em tempo real (e publicações cada vez mais constantes) (GRAVES; AMAZEEN, 2019; FATOS, 2024). Além disso, é observada uma tendência dos veículos de comunicação ao jornalismo declaratório de autoridades e ao recurso *off the record*, que é quando as fontes não querem se identificar, o que muitas vezes é usado como um experimento da reação do público ao invés de informar (FIDALGO, 2017).

O jornalismo de verificação de fatos na forma que conhecemos hoje surgiu nos Estados Unidos, com iniciativas pontuais na década de 1990 para cobertura política. Na eleição de 1992, o jornalista Brooks Jackson se encarregou de verificar tudo o que era dito nos discursos dos candidatos à Casa Branca, sendo considerado pioneiro no formato (FACTCHECK.ORG, 2024; LUPA, 2015). Diante desse cenário de disseminação de informações falsas e imprecisas, o que começou como projetos ligados ao discurso político que logo acabavam após as eleições, originou a criação de instituições jornalísticas exclusivamente dedicadas a esta causa ao redor do mundo, com a primeira agência permanente de checagem de fatos, a *FactCheck.org*, sendo lançada em 2003 (FACTCHECK.ORG, 2024; LUPA, 2015).

O IFCN (International Fact-Checking Network), que conecta vários órgãos de checagem de fatos de maneira global, defende um código de princípios a ser seguido pelas agências, como forma de padronizar a operação e dar maior credibilidade ao seu papel na sociedade (GRAVES; AMAZEEN, 2019). Alguns destes princípios envolve analisar

declarações políticas das mais diferentes inclinações ideológicas, focar em fatos em vez de opiniões, usar fontes respeitadas e transparentes, como fontes oficiais ou acadêmicas, e garantir que todas as evidências e análises sejam transparentes para o público (IFCN, 2024).

Nesta dissertação, busca-se verificar e validar o uso de classificadores para auxiliar jornalistas na detecção de informação falsa, dado que a velocidade de propagação e o impacto de conteúdos deste tipo em quem os recebe pode causar grandes danos para a população (AIMEUR; AMRI; BRASSARD, 2023; SHARMA et al., 2019).

2.2 Modelos de Linguagem e Arquitetura Transformer - Visão Geral

Um modelo de linguagem é essencialmente uma simplificação voltada para a representação da linguagem humana de maneira computacional, embora atualmente já existam modelos capazes de desempenhar tarefas consideradas alheias à linguagem natural, como a geração de formalizações matemáticas e de código (PAES; VIANNA; RODRIGUES, 2024).

Atualmente, redes neurais são empregadas para aprender a função que determina a probabilidade das sequências, dando origem aos **modelos de linguagem neurais**. Uma contribuição importante foi a representação vetorial dos itens das sentenças (também chamados de *tokens*) como *embeddings*[6] (SENO et al., 2024). Enquanto a primeira geração de *embeddings* eram representações vetoriais para palavras isoladamente, os ***embeddings* contextualizados**, atual estado-da-arte, consideram o contexto da sentença no momento da utilização, permitindo assim representações distintas para um mesmo *token*, dependendo do contexto em que eles aparecem (JURAFSKY; MARTIN, 2023; PAES; VIANNA; RODRIGUES, 2024).

Para gerar *embeddings* contextualizados destacam-se dois métodos: as redes neurais recorrentes e os **Transformers** (VASWANI et al., 2017). Concentraremos nossa atenção nos modelos que seguem o segundo método, uma vez que foram os modelos usados nesta dissertação, por constituírem o estado-da-arte em diversas tarefas de Processamento de Linguagem Natural (PLN).

[6] Vetores densos de baixa dimensionalidade gerados a partir de textos.

2.2.1 Arquitetura básica dos Transformers

Transformers foi o nome dado à arquitetura de rede neural lançada em 2017 para modelagem de sequências de texto que revolucionou o campo do PLN, visto que, diferente dos modelos do estado-da-arte da época, galgou melhores resultados com um treinamento mais eficiente (VASWANI et al., 2017).

Em sua estrutura há dois componentes principais: o codificador e o decodificador. O codificador, ou *encoder*, processa a sequência de entrada, mapeando os símbolos recebidos para uma sequência de representações contínuas. O decodificador, ou *decoder*, recebe essa representação contínua e gera uma sequência de símbolos como saída, com um elemento gerado de cada vez e de forma **autorregressiva** (VASWANI et al., 2017), que é quando apenas os símbolos gerados anteriormente são considerados para a geração do próximo símbolo.

Modelagens com estes componentes já existiam antes, porém o diferencial dos Transformers foi utilizar mecanismos de atenção (VASWANI et al., 2017) para representar contexto, dispensando o uso de redes recorrentes. Assim, foi permitido treinar os pesos de um modelo em paralelo de forma mais eficiente (TUNSTALL; VON WERRA; WOLF, 2022). A Figura 1 mostra quais são os componentes formadores desta arquitetura.

O **codificador** é formado por uma pilha de camadas (originalmente seis) com estruturas idênticas e cada camada é constituída por dois elementos: um mecanismo de autoatenção de múltiplas versões (*multi-head self-attention*, em inglês), e uma rede neural completamente conectada de uma camada (*Feed-Forward*). Há uma conexão residual em torno de cada elemento das camadas (ou sub-codificadores), seguida por uma camada de normalização.

Já o **decodificador** também é composto por uma pilha de camadas idênticas (originalmente seis), onde cada camada (ou sub-decodificador), além de ter os mesmos elementos das camadas do codificador, possui um mecanismo de atenção de múltiplas versões (*multi-head attention*), que seria uma camada de atenção convencional, destinada para a comunicação com a saída do codificador. Outra diferença é que os mecanismos de atenção do decodificador não têm acesso aos *tokens* posteriores ao *token* gerado naquele momento (VASWANI et al., 2017).

Ao final da arquitetura Transformer, há duas camadas responsáveis por transformar a saída do decodificador, que é uma representação vetorial, em probabilidade. Na primeira, o vetor é processado por uma camada linear gerando um vetor contendo $\mid V \mid$ números

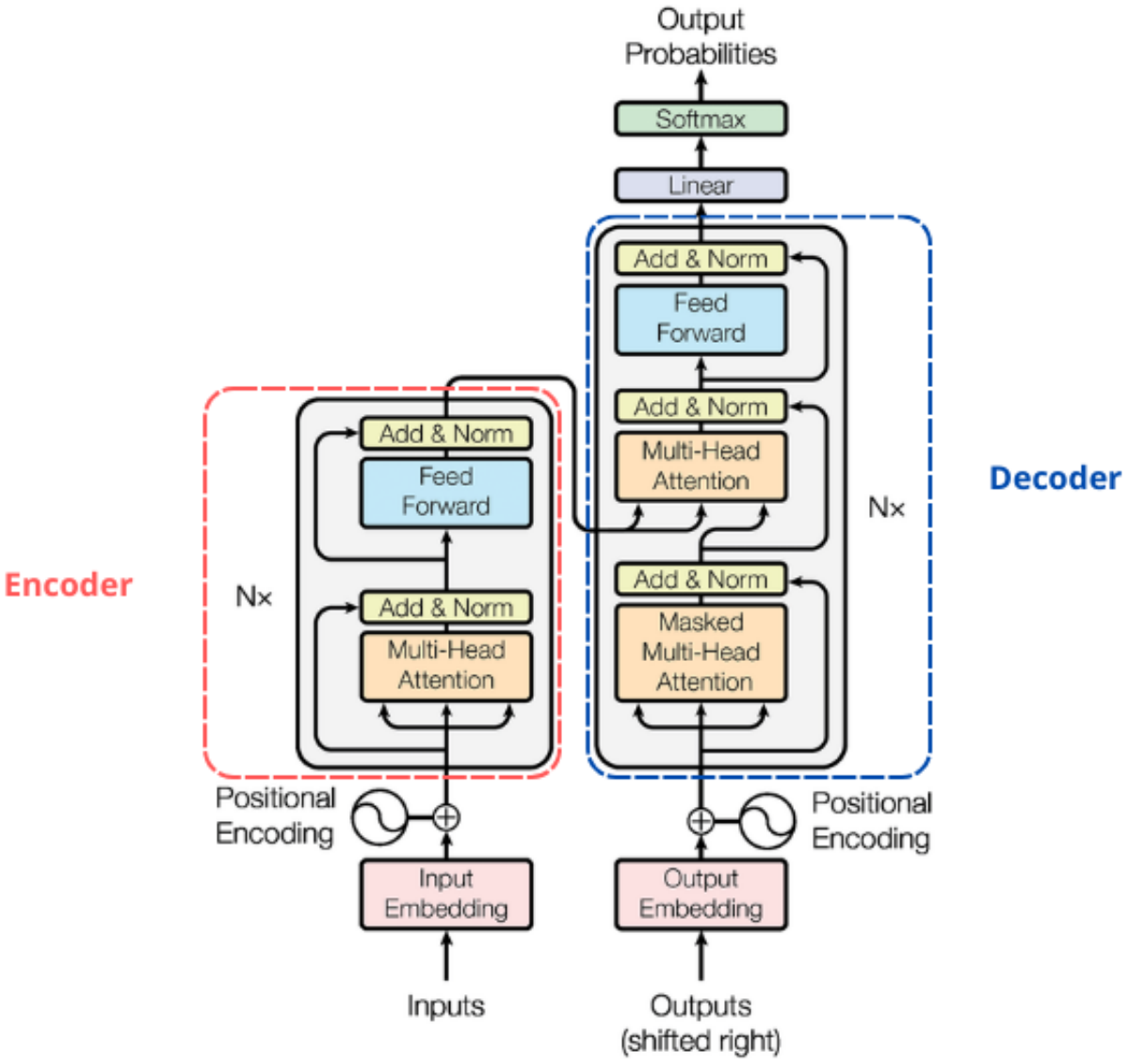

Figura 1: Arquitetura Transformer, adaptada de (VASWANI et al., 2017), com a sinalização dos componentes *encoder* e *decoder*.

reais, onde $| V |$ é o tamanho do vocabulário. A última camada submete este vetor a uma função de *softmax* para gerar a probabilidade de emissão de cada palavra do vocabulário (PAES; VIANNA; RODRIGUES, 2024).

De maneira resumida, os elementos desta arquitetura são os seguintes:

- Camadas de *embeddings*: o texto de entrada, já tokenizado, é transformado em uma matriz de *embeddings*, com um vetor para cada *token*, para que ele possa ser submetido às etapas seguintes.

- Codificador de posição ou *positional encoding*: guarda informações sobre a posição dos *tokens* no texto de entrada. É somado ao vetor de *embeddings* de cada item da entrada para representar tanto a posição absoluta quanto a distância relativa entre as palavras, preparando o dado a ser submetido aos componentes principais.

- Mecanismo de auto-atenção: identifica quais são as partes relevantes - ou que merecem mais "atenção" - de uma sequência de entrada. São atribuídos pesos a cada *token* em relação a sua importância para as demais partes da sequência, isto é, a sua relevância contextual. Desta forma, são geradas novas representações para cada *token*, os já citados ***embeddings* contextualizados**, que são capazes de capturar nuances semânticas e sintáticas. As novas representações são então combinadas para formar a saída final do mecanismo de auto-atenção, chamada de matriz Z.

- Mecanismo de *multi-head self-attention*: consiste no paralelismo das camadas de auto-atenção apresentadas anteriormente. Para a rede totalmente conectada lidar com as múltiplas matrizes Z geradas em paralelo, elas são concatenadas e multiplicadas por outra matriz de pesos adicional. Isso resulta em uma única matriz Z que representa o resultado do mecanismo de atenção em suas diferentes versões. Tal variabilidade possibilita a captura de diferentes representações dos termos de entrada, considerando que, a depender da sentença fornecida, podem surgir problemas de correfência ou polissemia entre seus termos (PAES; VIANNA; RODRIGUES, 2024). Na arquitetura original foram aplicadas oito camadas paralelas de atenção, também chamadas pelo autor de *heads* (VASWANI et al., 2017).

- Camada residual: tem como motivação evitar a perda de informações importantes durante o processamento em redes profundas, como o desaparecimento do gradiente. Na prática, algumas operações são ignoradas, fazendo com que representações originais destes *tokens* não passem pela camada de auto-atenção e pela rede completamente conectada.

2.2.2 Modelos de linguagem baseados em Transformers

Originalmente, a arquitetura Transformer foi projetada para atender a tarefas que transformam uma sequência em outra (*sequence-to-sequence*) (TUNSTALL; VON WERRA; WOLF, 2022), como, por exemplo, tarefas que geram textos a partir de uma entrada também textual, como tradução e sumarização. No entanto, seus principais componentes - a camada *encoder* e a camada *decoder* - de alto nível foram logo adaptados como modelos independentes.

Estes modelos, que adotam a arquitetura Transformer, deram origem a três categorias: **modelos *encoder***, que utilizam apenas o codificador, tendo como principal representante o BERT (*Bidirectional Encoder Representations for Transformers*) (DEVLIN et al., 2019); **modelos *decoder***, que utilizam apenas o decodificador, tendo como destaque os modelos da família GPT (RADFORD; NARASIMHAN et al., 2018); e os **modelos *encoder-decoder***, que fazem uso dos dois componentes, portanto mais próximos da estrutura original, como o BART (LEWIS et al., 2019) e o T5 (*Text-to-Text Transfer Transformer*) (RAFFEL et al., 2020).

O BERT (*Bidirectional Encoder Representations for Transformers*) (DEVLIN et al., 2019) é um modelo de linguagem pré-treinado que, devido a sua arquitetura de codificação bidirecional, consegue analisar contexto nas duas direções, gerando uma compreensão mais abrangente. Foi treinado em duas versões: *base* e *large*, onde a primeira possui 12 subcamadas de codificadores com 12 *heads* de atenção e a segunda com 24 subcamadas de codificadores e 16 *heads*. Em ambas as versões a sequência de entrada é limitada a 512 *tokens*, o que compromete seu uso com textos maiores. O processo de tokenização deste modelo envolve dois *tokens* especiais, que são o *token* [CLS], que indica o início do texto de entrada (no qual seu embedding de retorno representa todo o texto), e o *token* [SEP], que pode indicar tanto o final do texto de entrada como também a separação entre sentenças ou textos. Seu pré-treinamento se deu em dois tipos de tarefa: previsão de palavras mascaradas em uma sentença e previsão da próxima sentença com base na anterior.

O modelo T5 (*Text-to-Text Transfer Transformer*) (RAFFEL et al., 2020) segue a arquitetura Transformer com algumas adaptações, como o fato do codificador de posição passar a ser relativo. Seu pré-treinamento ocorreu em diversas tarefas, dentre elas a análise de sentimentos e a similaridade de sentenças. Um ponto importante foi a apresentação da possibilidade de enviar instruções curtas para indicar ao modelo qual das tarefas ele deveria executar, o que os autores chamaram de **prefixo específico de tarefa**, dando

início ao que depois se tornaria um novo universo de instruções com modelos de linguagem.

Neste trabalho de dissertação, foram utilizados o modelo **BERTimbau** (SOUZA; NOGUEIRA; LOTUFO, 2020), que é uma versão pré-treinada em português do modelo BERT, e o modelo **mT5** (XUE et al., 2020), que é a versão multilíngue do modelo T5, contemplando o português, visando explorar como lidar com modelos que seguem arquiteturas distintas na classificação de notícias falsas.

Também foi realizado o uso de modelos de linguagem de plataformas fechadas, com os modelos **embed-multilingual-v2.0** (que será referenciado como **cohere-*embeddings***) e **Command** disponibilizados pela empresa Cohere[7], além do modelo **Sabiá-3** (PIRES et al., 2023), do grupo de pesquisa Maritaca AI. O primeiro modelo é um *encoder* multilíngue com *embeddings* de 768 dimensões e adota como métrica de similaridade textual o produto escalar. Os dois últimos são modelos que seguem a arquitetura *decoder*, sendo baseados em instruções. O Command foi treinado também em português do Brasil, enquanto o Sabiá-3 é reconhecido como o maior modelo de linguagem treinado em português brasileiro atualmente e está na sua versão mais recente. Os modelos disponibilizados pela Cohere oferecem baixo custo quando comparados a outros modelos acessíveis por meio de APIs e, assim como o grupo Maritaca AI, disponibilizam créditos para a elaboração deste trabalho.

2.3 Pré-treinamento de modelos de linguagem

O **pré-treinamento** de modelos de linguagem é o primeiro treinamento ao qual eles são submetidos. É neste estágio que redes neurais profundas são treinadas com uma abundância de textos não rotulados, visando desenvolver um modelo capaz de processar linguagem de forma geral. Os modelos pré-treinados têm como propósito macro a geração ou preenchimento de texto, e podem ser refinados posteriormente para um domínio ou tarefa específica (TUNSTALL; VON WERRA; WOLF, 2022).

O processo de pré-treinamento de um modelo de linguagem envolve várias etapas cruciais: a seleção cuidadosa de corpora para treinamento, a limpeza e pré-processamento dos textos, o treinamento do tokenizador para dividir o texto em *tokens*, a definição da arquitetura do modelo, a especificação da função objetivo ou tarefa intermediária para orientar o aprendizado, a definição dos hiperparâmetros para controlar o treinamento, e a avaliação do modelo quanto à qualidade dos textos gerados e ao desempenho em tarefas de PLN específicas (PAES; VIANNA; RODRIGUES, 2024). É importante destacar que o

[7] https://cohere.com/

pré-treinamento demanda recursos computacionais consideráveis por um longo espaço de tempo, o que desencadeia um consumo excessivo de energia e acarreta impactos ambientais que precisam ser melhor discutidos na sociedade (PAES; VIANNA; RODRIGUES, 2024).

2.3.1 Tarefas intermediárias mais comuns

Durante o pré-treinamento é necessário estipular o que vai orientar o aprendizado do modelo, o que é feito através das chamadas **tarefas intermediárias**. As duas tarefas intermediárias mais comuns são a Modelagem de Linguagem Mascarada (MLM) e a Modelagem de Linguagem Causal ou Autorregressiva (AR).

A tarefa de MLM foi utilizada no pré-treinamento do modelo BERT, e sua inspiração vem do teste de linguagem Cloze (TAYLOR, 1953). Os textos de entrada são alterados de modo que alguns *tokens* são substituídos pelo *token* especial [MASK], onde o intuito é estimar quais *tokens* poderiam substituir quais foram mascarados considerando o contexto das sentenças (PAES; VIANNA; RODRIGUES, 2024; JURAFSKY; MARTIN, 2023). Já a tarefa de AR segue a mesma ideia de geração de *tokens* do decodificador nos Transformers, ou seja, a geração do próximo *token* de uma sentença considera apenas os *tokens* anteriores a ele.

2.3.2 Inferência com modelos pré-treinados

De maneira geral, o processo de inferência em IA consiste na submissão de dados para um determinado modelo sem que ele os conheça previamente. É por meio da inferência, popularmente chamada de teste, que a capacidade de generalização adquirida no treino é posta à prova. Com modelos de linguagem as inferências são realizadas com (i) modelos pré-treinados, ou (ii) com modelos pré-treinados que foram especializados em algum domínio ou tarefa específica.

Quando um modelo de linguagem pré-treinado é usado sem aperfeiçoamento posterior, trata-se de uma abordagem de **few-shot learning**, que pode ser compreendida como um método em que o modelo adquire informações de maneira rápida (por isso o *shot*) por meio de poucas amostras (daí o termo *few*) para efetuar inferências, sem que nenhuma atualização em seus pesos seja efetuada (SONG et al., 2023; BROWN et al., 2020).

Existem variações deste termo a depender da quantidade de informação transmitida ao modelo pré-treinado para que ele possa inferir algo, que pode ser apenas um exemplo, o que é chamado de **one-shot** ou ainda sem informar exemplo algum, configurando o que

é chamado de abordagem **zero-shot**. Em (RADFORD; WU et al., 2019), trabalho que precedeu (BROWN et al., 2020), já se apontava para a capacidade iminente de modelos de linguagem serem "aprendizes multitarefa não supervisionados", visto que, na fase de pré-treinamento, os dados utilizados não são rotulados e, com modelos mais robustos, o entendimento da linguagem tende a ser melhor internalizado. Nesta dissertação, alguns experimentos seguirão a abordagem *zero-shot*, com o intuito de verificar a capacidade de generalização dos modelos de linguagem trabalhados para o domínio de *fake news* com textos em português.

2.4 Ajuste fino de modelos de linguagem

Os modelos pré-treinados costumam ser aprimorados para algum propósito específico. Quando isso ocorre, é dito que houve um ajuste no modelo. A abordagem mais comum consiste em realizar um ajuste fino (*fine-tuning*) no modelo, incluindo uma ou mais camadas adicionais no modelo pré-treinado que terão como função objetivo uma tarefa-alvo específica (JURAFSKY; MARTIN, 2023).

A ideia por trás desta técnica é criar modelos especializados sem um treinamento exaustivo, dado que se espera que no pré-treinamento a linguagem já foi assimilada de maneira profunda. Outro aspecto importante é que no final trata-se de aprendizado por transferência, ou *transfer learning*, uma vez que o conhecimento prévio do modelo é usado como meio para uma aplicação fim diferente da original (JURAFSKY; MARTIN, 2023). Neste trabalho, foi implementado o *fine-tuning* dos modelos BERTimbau, mT5 e cohere-*embeddings*.

3 Trabalhos relacionados

A profusão de *fake news* é um problema que vai além do meio virtual, com alto poder de influência na tomada de decisões que impactam milhares de pessoas (MASON; KRUTKA; STODDARD, 2018), e a verificação de fatos não consegue competir na mesma velocidade. Também há poucos esforços em promover a alfabetização digital, que tem se mostrado um meio eficaz na formação de indivíduos mais críticos sobre o que consomem nas redes sociais (MILLER; MENARD; BOURRIE, 2024; DAME ADJIN-TETTEY, 2022; JONES-JANG; MORTENSEN; LIU, 2021), embora não seja uma medida imediata.

No campo da IA, existem várias iniciativas na literatura que buscam contribuir com a detecção automática de *fake news*, mediante a utilização de diferentes abordagens de ML (*Machine Learning*, ou aprendizado de máquina), seja comparando modelos tradicionais com redes profundas e modelos de linguagem (KHAN et al., 2021), avaliando quais características podem ser extraídas do texto para enriquecer as bases de treinamento (HASHMI et al., 2024; BALSHETWAR; RS, 2023; PARK; CHAI, 2023), ou discutindo o impacto de dados de contexto, conteúdo e de combinações entre eles (ALGHAMDI; LUO; LIN, 2024; CAPUANO et al., 2023).

Contudo, a maioria dos estudos sobre *fake news* trabalham apenas com dados em inglês (ALGHAMDI; LIN; LUO, 2024; FISCHER et al., 2022). Tal predominância também se dá em domínios de conhecimento correlatos: Trajano, Bordini e Vieira (2023) mostraram que a maioria dos grandes avanços de pesquisa em detecção de discurso ofensivo se concentram na língua inglesa. De forma geral, os demais idiomas são referenciados como *low-resource languages*, por terem poucos dados rotulados disponíveis. Na revisão sistemática sobre detecção de discurso de ódio conduzida por Poletto et al. (2021 apud TRAJANO; BORDINI; VIEIRA, 2023), dos 64 datasets encontrados, 37 estavam em inglês e apenas dois em português.

A pesquisa de Du et al. (2021) evidencia os prejuízos da escassez de iniciativas para combater a desinformação em outros idiomas. O estudo ilustra essa problemática ao

analisar o contexto da pandemia de Covid-19 nos Estados Unidos, onde a disseminação de notícias falsas em línguas minoritárias no país, faladas por imigrantes, expôs as limitações dos mecanismos de moderação de conteúdo das redes sociais e de alcance das agências de checagem, uma lacuna também observada nas eleições de 2020[1].

Durante as pesquisas por trabalhos sobre identificação de notícias falsas em português, conseguimos mapear diversos esforços nesta direção, culminando na seleção dos datasets trabalhados nesta dissertação e explorados na **RQ1**, como veremos mais adiante. Em meio a esta busca, notamos também trabalhos utilizando o dataset Fake.Br (MONTEIRO et al., 2018) para experimentação, que foi o primeiro criado para este fim em português, e teve uma construção cuidadosa. O pioneirismo e a metodologia utilizada são possíveis razões para explicar a sua predileção em estudos que não se propõem a angariar novos dados.

Para além dos trabalhos dos datasets selecionados, há também estudos recentes sobre detecção de *fake news* em português que alcançaram bons resultados: Renato M Silva et al. (2020), Fischer et al. (2022) e Moreira et al. (2023), todos utilizando o dataset Fake.Br e comparando diversos métodos tradicionais de classificação nesta tarefa, o que se assemelha à **RQ3**. Em Renato M Silva et al. (2020) modelos consagrados foram combinados à características baseadas na linguagem (como pausalidade e emotividade) e formas de representação de texto (Bag of Words (BoW), Word2Vec, e FastText), com os modelos Support Vector Machines (SVM), Random Forest (RF) e Regressão Logística (RL) obtendo os melhores resultados, este último atingindo 97,1% de F1-score com BoW, algo inesperado para os autores, dada a simplicidade desta forma de representação. Além disso, esta abordagem ultrapassa o resultado obtido pelo classificador SVM divulgado na apresentação do dataset, que era 89% (MONTEIRO et al., 2018).

Fischer et al. (2022) acrescentou modelos de *deep learning* e o modelo BERT em sua versão multilíngue (mBERT), que obteve o melhor resultado, com um F1-score de 98,4%, apresentando crescimento frente ao resultado anterior. Moreira et al. (2023) elaborou um estudo similar, com diferentes modelos de *machine learning* (ML) e também com modelos da família BERT, porém com a sua versão original, o que demandou a tradução dos textos para o inglês[2], e a distribuição treinada em português, o BERTimbau. Mais uma vez os modelos de linguagem atingiram as melhores marcas, com o BERTimbau alcançando F1-score de 98,95% para notícias falsas e 98,74% para notícias verdadeiras.

[1] https://www.vox.com/identities/21579752/asian-american-misinformation-after-2020
[2] A tradução se deu através da Google Cloud Translation API.

No entanto, nos dois trabalhos anteriores à Moreira et al. (2023), os dados de treino e teste foram divididos uma única vez (técnica conhecida como *holdout*), enquanto ele utilizou uma validação cruzada com dez divisões (comumente referenciada como *10-fold cross-validation*), que é um método amplamente utilizado em previsão para avaliar a capacidade de generalização de modelos preditivos (BERRAR, 2019), o que traz maior robustez ao resultado.

Há também revisões da literatura que mostram que os modelos de linguagem se adaptam bem a este tipo de tarefa, o que credencia a escolha dos modelos de linguagem aqui trabalhados. Khan et al. (2021) mostrou em um estudo com 19 modelos para detecção de notícias falsas que o modelo BERT e outros baseados na arquitetura Transformer obtiveram os melhores resultados. Capuano et al. (2023) em uma revisão sistemática com a análise de 40 artigos sobre detecção de *fake news* verificou que os modelos XLNet, ALBERT e LSTM com BERT obtiveram os resultados mais promissores nos últimos anos.

Embora haja a preocupação em descobrir quais modelos e técnicas são mais eficazes para classificar *fake news*, a avaliação do poder de generalização das soluções propostas na maior parte das vezes é negligenciada. Modelos são apresentados com bons resultados sobre conjuntos de teste, formados por dados não vistos durante o treinamento, porém não é verificada a desenvoltura destes mesmos modelos sobre dados de outras fontes, isto é, que não são parte do conjunto inicial de dados que deu origem aos subconjuntos utilizados para treinamento e inferência.

Este comportamento é o que definimos na Seção 1.2 como **validação *in-data***, tema de estudo da **RQ3**. Sem uma avaliação ampla neste sentido, os modelos continuam restritos a experimentos que não conversam com dados de notícias falsas no mundo real. A busca por modelos mais generalizáveis motivou a elaboração da **RQ4**, que aborda a **validação *cross-data***, também definida previamente. Buscamos na literatura trabalhos sobre generalização de modelos para o nosso domínio de estudo e correlatos, e embora eles não tenham como foco textos na língua portuguesa, os achados destes estudos podem servir como guia para experimentos futuros.

Hoy e Koulouri (2021), além de citarem que a ausência de uma definição comum sobre o tema pode gerar modelos enviesados, os autores abordam a necessidade da geração de modelos precisos, robustos e generalizáveis para a aplicação em cenários do mundo real, tendo como principais entraves o tamanho e a qualidade dos datasets disponíveis. Identificaram também que a capacidade de generalização poderia ser investigada em três grandes frentes, que são elas: (i) generalização entre datasets; (ii) generalização ao longo

do tempo; e (iii) generalização entre domínios. Nos experimentos *cross-data* é contemplada a avaliação dos pontos (i) e (iii), visto que nem todos os datasets disponibilizam informações temporais.

Em Hoy e Koulouri (2022), os mesmos autores exploraram a capacidade de generalização entre datasets sobre política. Realizaram uma série de experimentos com algoritmos clássicos como AdaBoost, Gradient Boosting, RL, Rede Neural, RF, SVM e investigaram o desempenho de representações ao nível de palavras (BoW, TF-IDF, Word2Vec, BERT) e de pistas linguísticas (34 ao todo, incluindo quantidade de palavras e de pontuações). Quando confrontados com dados de outros datasets, a taxa de acertos dos modelos sofreu uma redução de cerca de 50% em comparação com a validação *in-data*, com destaque para os classificadores treinados com datasets maiores.

Para detecção de discurso de ódio implícito, Kim, Park e Han (2022) efetuaram avaliações cruzadas entre três datasets com os modelos BERT e HateBERT, no qual o último foi pré-treinado em bases de textos com teor abusivo. Um diferencial do trabalho foi o uso de aprendizado contrastivo para o *fine-tuning* dos modelos selecionados, que "faz com que os pares positivos fiquem próximos e os pares negativos fiquem separados no espaço de representação" (RETHMEIER; AUGENSTEIN, 2023 apud KIM; PARK; HAN, 2022).

Nas avaliações cruzadas os modelos apresentaram um desempenho relativamente baixo, com uma queda de pelo menos 12,5% na pontuação F1-score em comparação com validações *in-data*, porém com a adoção da aprendizagem contrastiva houve uma melhora nos resultados das avaliações cruzadas em até 9%, mostrando-se como um caminho a ser explorado na construção de modelos mais generalizáveis.

Em Ng e Carley (2022), a capacidade de generalização foi avaliada sobre sete datasets contendo posturas associadas a *tweets* sobre temas variados. A detecção de postura (ou *stance detection*) busca classificar a inclinação do autor de um texto como favorável, contrária ou neutra. Nas redes sociais, ela é promissora para a detecção de *fake news*, por auxiliar na investigação do surgimento de rumores e em como postagens sobre supostos incidentes conseguem adeptos para a formação de um consenso (LILLIE; MIDDELBOE, 2019).

Foram criados classificadores a partir dos seguintes modelos: BERT, ALBERT, DistilBERT, RoBERTa, XLNet, onde destes BERT e DistilBERT obtiveram os melhores desempenhos. Na avaliação cruzada, assim como foi visto em outras pesquisas, a generalização foi baixa, com um F1-score médio de 0,33. Os autores também avaliaram a generalização com a agregação de todos os dados, o que elevou o resultado para 0,69,

após os rótulos das classes serem padronizados.

Também se observa uma tendência crescente na exploração das habilidades emergentes dos modelos de linguagem. Pan, Zhang e Kan (2023) investigaram a capacidade de generalização partindo de abordagens *few-shot* e *zero-shot* para o modelo RoBERTa sobre oito datasets. O trabalho também fez um comparativo destas habilidades com modelos pré-treinados em domínios específicos, utilizando os modelos BioBERT e SciBERT como referência, que obtiveram resultados melhores em comparação aos experimentos *zero-shot* do RoBERTa. A abordagem *few-shot* por sua vez, mostrou que com o *fine-tuning* de uma pequena parcela dos exemplos do dataset, os resultados melhoraram consideravelmente. Para a **RQ2**, trabalharemos com experimentos *zero-shot* sem o uso de modelos especializados, visando investigar tais habilidades sobre os datasets selecionados em português.

Existe a preocupação de que os LLMs, se mal utilizados, possam ser empregados para gerar notícias falsas mais convincentes (LOTH; KAPPES; PAHL, 2024; SANDRINI; SOMOGYI, 2023), o que precisa ser debatido e investigado em várias esferas da sociedade. Em um estudo preliminar, Hu et al. (2024) compararam os modelos BERT (com *fine-tuning*) e GPT3.5-turbo na detecção de notícias falsas, e verificaram que o LLM não conseguiu ultrapassar os resultados obtidos pelo modelo menor. No entanto, ele se mostrou eficaz na geração de informação complementar sobre a compreensão das notícias, o que alavancou os resultados do BERT, mostrando que talvez os LLMs não sejam os mais recomendáveis para esta tarefa, mas podem ser bons conselheiros.

Por fim, os trabalhos mencionados reforçam a necessidade de se construir modelos generalizáveis e mostra que este campo ainda é muito deficitário. Nesta dissertação, vamos explorar a capacidade de generalização de diferentes modelos de linguagem, embora sem a intenção de construir o melhor classificador de notícias falsas, mas sim de avaliar quais caminhos são mais promissores para que novos esforços sejam empregados no futuro.

4 Metodologia

Neste capítulo será apresentada a metodologia adotada neste trabalho para investigar a capacidade de generalização de classificadores de *fake news* com foco na língua portuguesa. A abordagem compreende três etapas principais: (i) a obtenção dos dados, com a seleção de datasets, a análise de seus exemplos e a aplicação de transformações necessárias para a sua exploração; (ii) a preparação dos dados, que recebe os dados coletados dos datasets como entrada e efetua os tratamentos necessários; e (iii) a execução dos experimentos, que prepara os dados processados na etapa anterior e os submete às estratégias adotadas para treinamento dos classificadores ou avaliação dos modelos de linguagem pré-treinados.

Durante a etapa (ii) os datasets foram analisados, cujos resultados são apresentados e discutidos no Capítulo 5. A Figura 2 exibe o panorama dos passos adotados na metodologia proposta. As seções que seguem trazem os detalhes de cada passo da metodologia, a começar pela seleção dos datasets. O código dos experimentos e análises está disponível publicamente[1].

4.1 Etapa 1: Levantamento de dados disponíveis

4.1.1 Seleção de datasets

Para responder à **RQ1**, foi realizada uma busca por trabalhos que tenham gerado datasets de notícias falsas em português, publicados no período de 2018 a 2022, levando em consideração os seguintes critérios: (i) aderência ao domínio (*fake news*), (ii) existência de dados em português, e (iii) disponibilidade dos dados. O Capítulo 5 focará na segunda parte desta questão de pesquisa, que busca compreender quais são as características principais dos conjuntos de dados desta natureza. A Tabela 1 apresenta os termos utilizados na busca por datasets.

Foi observado durante a pesquisa que alguns trabalhos tiveram a geração de datasets

[1] https://github.com/MeLLL-UFF/LMFactCheck

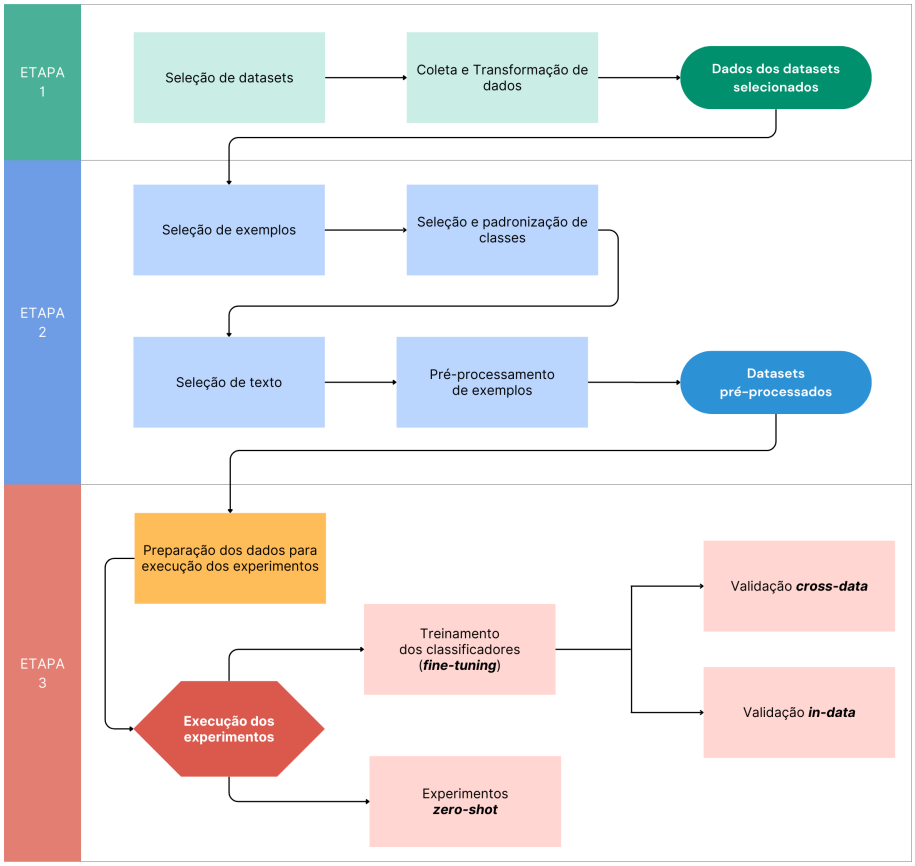

Figura 2: Metodologia geral adotada na pesquisa, separada por etapas. Após a seleção, coleta e transformação dos datasets, cada um deles é submetido aos passos de tratamento dos dados. Ao final da segunda etapa, é gerada uma versão pré-processada com os seus dados resultantes. Na terceira etapa, a versão pré-processada dos dados é ajustada para servir como entrada dos modelos de linguagem na execução dos experimentos.

Tabela 1: Termos utilizados durante a busca de artigos. Foram criadas combinações destes termos para alcançar o máximo de trabalhos possíveis, com pesquisas por título e conteúdo.

Critérios	**Termos de busca**
(i)	fake news, misinformation, disinformation, fact-checking, checagem de fatos, notícias falsas
(ii)	português, portuguese, língua portuguesa, multilingual, Brazil, Brazilian, Brasil
(iii)	dataset, corpus, repositório

de notícias falsas como objetivo principal. Em outros casos, a proposta foi classificar informações falsas ou investigar a sua propagação nas redes sociais, e a coleta de dados serviu como meio, o que no final também gerou novas bases como contribuição na área ao partilharem os dados trabalhados em seus estudos.

Ao final deste processo foram coletados 14 datasets de notícias ou informações falsas. A Tabela 2 apresenta os trabalhos selecionados e algumas de suas principais características, como fonte e temática dos dados (mais informações sobre os trabalhos selecionados estão disponíveis no Apêndice A). Dos datasets selecionados, cinco deles possuem temática única: três contém apenas notícias sobre Covid-19 e dois são dedicados às eleições presidenciais brasileiras de 2018.

Tal diferença pode levantar a reflexão acerca da dificuldade em construir conjuntos de dados de notícias falsas com assunto específico. Pode ser uma tarefa trabalhosa se o tópico não atingir uma quantidade considerável de pessoas para fomentar discussões e possivelmente o surgimento de desinformação, como também o impacto da temporalidade do tópico em si. As eleições, por exemplo, são eventos sazonais, então por mais que a política seja um elemento vivo na sociedade, fora do período eleitoral não há uma intencionalidade exacerbada para convencer um grupo a escolher determinado candidato.

Em relação às fontes dos dados, existem alguns padrões declarados: há datasets constituídos apenas por dados de redes sociais, outros apenas por notícias de agências de checagem de fatos. Alguns são formados por uma mescla do conteúdo publicado por estas agências com notícias de veículos de mídia conhecidos do grande público e, em menor evidência, o uso de APIs para coleta de notícias verificadas, que durante o seu processo acaba recorrendo ao conteúdo das agências de checagem ou a notícias de portais tradicionais.

Assim, é possível notar que os datasets são compostos por exemplos extraídos de formas distintas de se comunicar, ora com textos curtos e objetivos das redes sociais, ora com artigos de opinião de jornalistas que publicam semanalmente para seus leitores cativos. Nos datasets multilíngue, observa-se a preferência por utilizar ferramentas que façam uma busca mais ampla ou ainda a escolha por portais que, de alguma forma, colham informações de outras agências de checagem ao redor do mundo.

4.1.2 Coleta e Transformação de dados

Após a seleção dos datasets a partir dos trabalhos publicados, todos os repositórios indicados pelos autores foram acessados; contudo, alguns datasets precisaram ser transformados

Tabela 2: Relação de datasets sobre notícias falsas selecionados para este estudo, contendo a sua identificação, o ano em que seus trabalhos de origem foram publicados, de onde vieram os dados utilizados por eles e a temática abordada. O primeiro dataset da listagem não recebeu um nome específico por seus autores, por isso aqui está representado pelo ano e o nome do veículo de publicação.

Dataset	**Ano**	**Idioma**	**Tema**	**Origem**
BRACIS2019	2019	português	eleições 2018	WhatsApp (via Boatos.org), Twitter, *Mídia tradicional* para notícias verdadeiras (fontes não informadas)
Central de Fatos	2021	português	diversos	Aos Fatos, Estadão Verifica, Lupa, Boatos.org, Projeto Comprova, Fato ou Fake
COVID19BR	2021	português	Covid-19	WhatsApp
Factck.BR	2019	português	diversos	Ag. Publica (Truco), Lupa, Aos Fatos
Fake.Br	2018	português	diversos	Diário do Brasil, A Folha do Brasil, The Jornal Brasil, Top Five TV, G1, Folha de São Paulo, Estadão
FakeCovid	2020	multilíngue	Covid-19	Snopes, Poynter
FakeNewsSet	2020	português	diversos	Lupa, Aos Fatos, AFP, G1, R7
Fakepedia	2022	português	diversos	Boatos.org, Lupa, Aos Fatos, Globo, UOL, Extra, Folha de São Paulo
FakeRecogna	2022	português	diversos	G1, UOL, Extra, Ministério da Saúde, AFP Checamos, Boatos.org, E-farsas, Fato ou Fake, Projeto Comprova, UOL Confere
FakeTweet.Br	2019	português	diversos	Twitter
FakeWhatsApp.Br	2021	português	eleições 2018	WhatsApp
MM-COVID	2020	multilíngue	Covid-19	Snopes, Poynter, Sites de órgãos oficiais de saúde, Twitter
MuMiN	2022	multilíngue	diversos	Twitter, Google Fact Check Tools API
X-FACT	2021	multilíngue	diversos	Google Fact Check Explorer

antes de prosseguirem para a próxima etapa da metodologia. Isto foi necessário para padronizar o formato dos datasets e verificar se as informações principais - texto da notícia e classificação - estavam dispostas adequadamente, respeitando os seguintes critérios:

- Cada dataset deve conter um atributo com a indicação da classe dos exemplos,
- Para cada exemplo, deve existir exatamente uma classe associada, e
- Em cada dataset, os textos referentes a uma mesma seção da notícia devem estar contidos no mesmo atributo. Por exemplo, se há um atributo para títulos de notícia, ele deve ser único.

Inicialmente, os datasets foram padronizados para o formato *csv* e aqueles cujos dados estavam presentes em mais de um arquivo tiveram suas informações reunidas em um único. Padronizar o formato é importante para que todas as informações possam ser acessadas e manuseadas da mesma forma. Após isso, foi feita a verificação das informações principais, quando foi observado que em dois datasets a indicação da classe dos exemplos não estava explícita ou era dúbia: Central de Fatos (COUTO et al., 2021) e Fakepedia (CHARLES; RUBACK; OLIVEIRA, 2022).

No dataset Central de Fatos, a classe tem seus valores originalmente dispostos como listas, variando de 1 a 41 elementos cada. A Figura 3 traz alguns exemplos deste dataset, com a classificação representada pelo atributo *rating*. Os autores afirmam que isso acontece quando o dado coletado advém de análises sobre discursos políticos, nos quais a classificação pode variar a depender do trecho analisado. Na ausência de indicação do trecho do texto referente à cada classe da listagem, foram efetuadas algumas verificações até se chegar ao máximo número possível de elementos com apenas uma classe associada, havendo o descarte dos demais, como mostra a Figura 4.

title	text_news	rating
Palocci comprou Lotus de Senna e colocou carro...	Depois que foi preso na 35ª fase da Operação L...	['boato']
As checagens do debate da Band com os candidat...	Postulantes à Prefeitura do Rio de Janeiro par...	['verdadeiro', 'falso', 'falso', 'falso', 'fal...
Cunha renuncia e apaga uma série de tuítes neg...	O deputado federal Eduardo Cunha (PMDB-RJ) ren...	['CONTRADITÓRIO', 'EXAGERADO']
Bolsonaro erra ao afirmar que Centrão surgiu c...	O presidente Jair Bolsonaro (PSL) gravou entre...	['FALSO', 'SUBESTIMADO', 'VERDADEIRO']

Figura 3: Exemplos do dataset Central de Fatos antes da transformação de dados, evidenciando a classificação em formato de lista e podendo conter mais de um elemento.

Em Fakepedia, as notícias verdadeiras estavam como dois atributos adicionais dos exemplos de notícias falsas, contendo texto e url, e não como exemplos independentes. Eles foram transformados em novas entradas do dataset com a indicação da classe verdadeira,

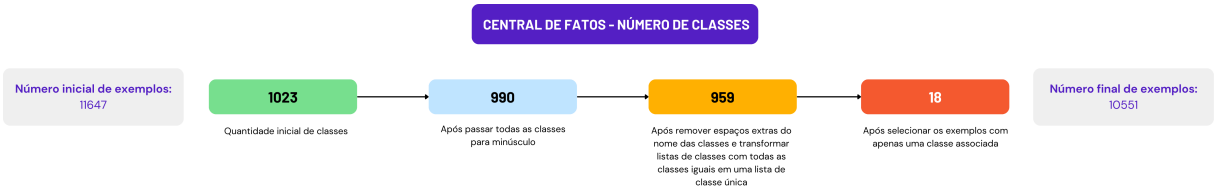

Figura 4: Quantidade de classes distintas do dataset Central de Fatos e sua variação durante o processo de transformação dos dados deste conjunto.

o que aumentou a quantidade de dados nulos, visto que a quantidade de informação disponível para notícias falsas não se restringe a apenas dois campos.

Para dois datasets, MuMiN (NIELSEN; MCCONVILLE, 2022) e BRACIS2019 (FAUSTINI; COVÕES, 2019), não foi possível coletar todos os exemplos, em virtude da atual política de acesso à *tweets* da rede social X (antigo Twitter). Como consequência, para o dataset BRACIS2019 apenas os exemplos com mensagens transmitidas pelo WhatsApp puderam ser coletados, formando um contingente consideravelmente menor. Para o MuMiN, foi coletada a menor versão disponível, a `MuMiN-small`.

4.2 Etapa 2: Tratamento dos dados

4.2.1 Seleção de exemplos

Após a etapa de Transformação, foi necessário efetuar algumas validações para eliminar exemplos não pertinentes aos objetivos propostos. Como o estudo se propõe a construir classificadores de notícias falsas, logo, fazendo uso de uma abordagem supervisionada, o primeiro crivo foi a existência de um rótulo de classe nos exemplos. Em seguida, a seleção também verificou qual versão dos datasets deveria ser considerada quando havia mais de uma disponível. A escolha levou em consideração a quantidade de registros a serem obtidos e a presença do texto em sua forma original, isto é, sem tratamentos aplicados. Por fim, foi realizada a seleção dos exemplos escritos em português.

Para o dataset FakeCovid (SHAHI; NANDINI, 2020), decidiu-se coletar os exemplos das duas versões encontradas como forma de aumentar o número de exemplos em português, visto que geralmente são escassos em datasets multilíngue. De maneira similar, em FakeNewsSet foi considerado o dataset utilizado na primeira etapa da metodologia *FakeNewsSetGen* (SILVA, F. R. M. da et al., 2020) pois, ao longo desta metodologia, estes dados são submetidos a diferentes etapas que buscam dados de divulgação das notícias no Twitter, o que originou um novo dataset com mais informação agregada (mediante novos

Titulo	Subtitulo	Noticia	Categoria	Data	Autor	URL	Classe
Qual será a espécie dominante na Terra se os seres humanos forem extintos?	Aconteceu com os dinossauros, e pode acontecer conosco. Mas não é fácil descobrir quem dominaria...	o ameaçar mudança climático e atual pandemia coronavírus e pessoa alertar existência perigar o a...	ciência	04/07/2020 16h46	Por BBC	https://g1.globo.com/natureza/noticia/2020/07/04/qual-sera-a-especie-dominante-na-terra-se-os-se...	1

Figura 5: Dataset FakeRecogna - Exemplo com pré-processamento aplicado ao texto da notícia. Nota-se que algumas palavras foram removidas e outras transformadas em verbos no infinitivo.

Tabela 3: Distribuição de registros em português nos datasets multilíngue. Os percentuais exibidos indicam a representação dos exemplos em português sobre cada base e no geral.

Dataset	**Número total de exemplos**	**Número de exemplos em português**
FakeCovid	12805	797 (6.22%)
MM-COVID	16254	640 (3.94%)
MuMiN	5075	503 (9.91%)
X-FACT	31189	7941 (25.46%)
Total	65323	9881 (15.12%)

campos), porém com menos exemplos em comparação com o dataset inicial.

Já o dataset FakeRecogna (GARCIA; AFONSO; PAPA, 2022) contém textos previamente tratados, o que, a depender do modelo a ser trabalhado, pode comprometer seu desempenho, visto que o emprego de tratamentos em excesso pode descaracterizar aspectos linguísticos importantes. Um exemplo disso são as *stopwords*, que podem não ter relevância para alguns modelos e serem removidas, enquanto para os modelos de linguagem podem ser importantes para referenciar elementos em sentenças. Neste caso, foi recuperada a base original adicionada no repositório, para mitigar a ocorrência de casos como o apresentado na Figura 5.

O próximo passo foi selecionar os exemplos por idioma. A indicação do idioma português nos datasets multilíngue se dá pelos valores *pt* e *Portuguese*, presentes em atributos com a indicação da linguagem adotada (nomeados como *lang* ou *language*). A Tabela 3 indica qual a porção remanescente dos dados das bases multilíngues após a seleção dos exemplos em português.

Em uma tentativa de ampliar a representatividade de informação em português nestas bases, foi verificado se, em meio aos exemplos sem idioma indicado e de países de língua portuguesa, outros exemplos poderiam ser considerados. Porém, nenhum novo exemplo

foi encontrado. Durante esta verificação, inclusive, foi descoberto que dentre os exemplos cadastrados com idioma português no dataset MM-COVID (LI et al., 2020), muitos não estavam de fato neste idioma.

Além disso, o texto deste dataset encontrava-se em mais de um campo: *claim* e *statement*, surgindo a necessidade de criação de um novo atributo para receber o texto válido de cada exemplo quando possível. Isso mostra que informações desencontradas comprometem a utilização dos datasets, dado que validações manuais são humanamente custosas. Para os demais datasets, assumiu-se que todos os textos, a julgar pela origem das informações, estão em português.

4.2.2 Seleção e padronização de classes

A classificação dos exemplos que compõem os datasets deste estudo vem de diferentes formas: **manual**, quando é concedida por pesquisadores mediante a análise do conteúdo disponível; **por checagem de fatos prévia**, quando se recupera checagens de fatos já realizadas e toma-se como classe o veredito atribuído pela agência responsável pela verificação; ou através da **confiabilidade da fonte**, onde se estipula um conjunto de meios de comunicação como *confiáveis*, também chamados por alguns de *mídia tradicional*, e assume-se que todas as notícias veiculadas por eles são verdadeiras.

Além da diversidade na forma de designar a classe de um exemplo, há uma falta de correspondência entre as classificações de notícia dadas pelas agências de checagem, visto que cada uma usa seus próprios termos - às vezes imagens - para indicar o veredito de determinado conteúdo em circulação. Isso faz com que modelos de classificação de texto voltados para *fake news* fiquem limitados apenas aos extremos, isto é, selecionando somente exemplos que são claramente verdadeiros ou falsos, apesar do fenômeno de disseminação de informações falsas ser multifacetado: pode ser um conteúdo fabricado, como pode ser um conteúdo verdadeiro movido para outro contexto, ou ainda conter porções de informações verdadeiras e falsas, e essas nuances são dificilmente capturadas olhando apenas para os extremos.

Devido à dificuldade em correlacionar as classificações de fontes distintas, foi verificado, para cada dataset, quais classes poderiam indicar se o conteúdo da notícia é verdadeiro ou falso, como forma de igualar as classificações entre os datasets, cuja maioria possui apenas duas classes. As classes selecionadas foram então convertidas nas classes "verdadeiro" ou "falso", havendo o descarte dos exemplos que não se encaixaram nestes casos.

Ao longo deste mapeamento, foram encontrados datasets contendo diferentes grafias para a mesma classe, o que precisou ser padronizado, como, por exemplo, no dataset FakeCovid, que possui classes como *Partially false, PARTIALLY FALSE, Partly false, PARTLY FALSE*. A Tabela 4 ilustra como a seleção de classes dos datasets foi construída.

A maioria dos conjuntos de classe apresentados são binários, com um termo remetendo à ideia de conteúdo falso e outro de modo antagônico, indicados por termos em português, em inglês, ou pelos números 0 (zero) e 1 (um). Embora, em geral, o zero seja visto como indicador de classificação negativa ou falsa, nos datasets COVID19BR (MARTINS et al., 2021) e FakeWhatsApp.Br (CABRAL et al., 2021) ele representa o oposto, visto que seus autores consideraram a existência ou não de *misinformation* na hora de classificar, logo o valor zero nestes dois conjuntos indica a ausência de informação falsa.

4.2.3 Seleção de texto

Antes do pré-processamento de texto dos exemplos, foi preciso selecionar qual atributo textual deveria ser considerado em cada dataset, pois em seis deles os textos das notícias estão distribuídos em mais de um campo, que podem guardar título, subtítulo, texto do corpo da publicação e alegação verificada. Os datasets que se encaixam nesta situação são: Central de Fatos, Factck.Br (MORENO; BRESSAN, 2019), FakeCovid, FakeNewsSet, Fakepedia e FakeRecogna.

Quando o dataset possuía a alegação que teve a sua veracidade checada, o seu campo correspondente era escolhido como texto principal, caso dos datasets Factck.Br e FakeNewsSet. Para os demais, após explorar o conteúdo dos campos candidatos, decidiu-se considerar como texto principal o conteúdo do corpo das publicações, pelo menor grau de dados ausentes.

4.2.4 Pré-processamento de exemplos

As técnicas de pré-processamento de texto empregadas visam a mínima interferência possível sobre o conteúdo original dos datasets, uma vez que não se vislumbra, neste momento, gerar versões aprimoradas dos datasets em uso, mas sim avaliá-los para o objetivo buscado. Portanto, a limpeza de dados adotada somente remove espaços em excesso, links, símbolos especiais, menções a nomes de usuários em redes sociais (como *@usuario* por exemplo), e-mails, *emojis* e marcadores de estilo, isto é, o símbolo * (negrito), e o símbolo _ (itálico).

Tabela 4: Detalhes sobre a distribuição de classes das notícias dos datasets selecionados. O traço - indica que não há classe predominante porque se tratam de conjuntos totalmente balanceados. Já *N/A* indica que não foi possível obter valor para este campo. Quando o dataset passou por padronização de classe, a quantidade original de classes é exibida entre parênteses ao lado da quantidade atual, na coluna "No.de Classes".

				Classes selecionadas para representar	
Dataset	**No. de Classes**	**Valores adotados**	**Classe principal (%)**	**"verdadeiro"**	**"falso"**
BRACIS2019	2	[0, 1]	0 (93.71)	1	0
Central de Fatos	18	[enganoso, falso, comprovado, contexto errado, evidência comprovada, fora de contexto, fake, fato, boato, distorcido, contraditório, exagerado, impreciso, verdadeiro, insustentável, "verdadeiro, mas", de olho, ainda é cedo para dizer]	boato (52.34)	verdadeiro, fato, comprovado	boato, falso, fake
COVID19BR	2	[0, 1]	0 (68.54)	0	1
Factck.BR	14 (18)	[falso, distorcido, impreciso, exagerado, insustentável, verdadeiro, outros, Subestimado, Impossível provar, Discutível, Sem contexto, De olho, "Verdadeiro, mas", Ainda é cedo para dizer]	falso (72.04)	verdadeiro	falso
Fake.Br	2	[0, 1]	-	1	0
FakeCovid	4 (13)	[False, partially false, misleading, Mostly false]	False (93.47)	N/A	False
FakeNewsSet	10 (14)	[falso, Verdadeiro, Enganoso, distorcido, Exagerado, "Verdadeiro, mas", Checamos, contraditório, Ainda é cedo para dizer, Insustentável]	falso (49.30)	Verdadeiro	falso, Enganoso
Fakepedia	2	[fake, true]	fake (58.52)	true	fake
FakeRecogna	2	[0, 1]	-	1	0
FakeTweet.Br	2 (3)	[fake, true]	fake (66.89)	true	fake
FakeWhatsApp.Br	2	[0, 1]	1 (53.67)	0	1
MM-COVID	2	[fake, real]	real (99.69)	real	fake
MuMiN	2	[misinformation, factual]	misinformation (99.00)	factual	misinformation
X-FACT	6	[mostly true, partly true/misleading, false, true, complicated/hard to categorise, other]	false (49.00)	true	false

Por meio destas técnicas, o uso dos *tokens* de entrada nos modelos de linguagem é otimizado, evitando que parte deles sejam designados para representar espaçamentos indevidos, links e outros termos que estão muito mais relacionados à forma como estes dados foram disponibilizados. Após a limpeza, para cada dataset foram removidos os exemplos com texto nulo, com conteúdo inteiramente repetido como também aqueles com texto duplicado, mantendo apenas um exemplo de cada um destes casos no dataset analisado. É importante remover dados duplicados para que eles não gerem modelos com viés e assim tenham a sua capacidade de generalização comprometida.

Para a duplicação de texto especificamente, foram observados dois cenários que desencadeiam este problema: por publicações em redes sociais, onde usuários podem replicar um mesmo conteúdo em novas postagens, ou quando agências de checagem de fatos diferentes averiguam uma mesma alegação, como mostra a Figura 6.

Author	claimReviewed	title
https:www.aosfatos.org	Você sabia que vereadores, deputados, governadores, senadores não pagam imposto de renda! Nem contribuem com a previdência!	É falso que políticos não pagam IR nem contribuem para a Previdência | Aos Fatos
https:piaui.folha.uol.com.brlupa	Você sabia que vereadores, deputados, governadores, senadores não pagam imposto de renda! Nem contribuem com a previdência!	#Verificamos: falso que políticos não pagam IR nem contribuem para previdência

Figura 6: Exemplo de duplicação de texto quando a fonte da informação diverge, aqui representada pelo campo “Author”.

Exemplos com sentenças contendo apenas uma palavra também foram desconsiderados, uma vez que é improvável gerar uma alegação passível de ser verificada com tamanha concisão. A Tabela 5 ilustra como cada dataset foi reduzindo em tamanho com o avançar das validações de preparação dos dados.

4.3 Etapa 3: Plano de experimentação

Para responder às questões de pesquisa **RQ2**, **RQ3** e **RQ4**, foram selecionados modelos de linguagem com arquiteturas distintas e combinados a diferentes estratégias de execução e validação para a composição dos experimentos. Para responder à RQ2, foram selecionados os modelos de linguagem pré-treinados sem ajuste, na modalidade *zero-shot*. Foram adotadas as tarefas intermediárias MLM e AR para os modelos BERTimbau e mT5 respectivamente, e o uso de *prompts de instruções* para os modelos de arquitetura *decoder*. Os detalhes acerca dos experimentos deste tipo podem ser conferidos na Seção 4.3.2.

Para responder às questões RQ3 e RQ4, foi efetuado o *fine-tuning* dos modelos de linguagem dos tipos *encoder* e *encoder-decoder*, gerando classificadores de notícias. Es-

Tabela 5: Distribuição de exemplos ao longo das primeiras verificações efetuadas sobre cada dataset. Partindo dos exemplos rotulados, foi verificado o número total de registros duplicados (coluna Sem duplicação Total), quais possuíam texto duplicado (coluna Sem duplicação Textual) e quais deles faziam parte de classes de interesse para o treinamento dos classificadores.

		Sem duplicação			**Distribuição final após Etapa 2**		
Dataset	**Exemplos rotulados e em português**	**Total**	**Textual**	**Com classes válidas**	**Classe "verdadeiro"**	**Classe "falso"**	**Total**
BRACIS2019	177	175	175	175	11	163	174
Central de Fatos	10551	10551	10548	9753	66	9686	9752
COVID19BR	2899	2899	2899	2899	1454	877	2331
Factck.BR	1309	1309	1289	1043	119	924	1043
Fake.Br	7200	7199	7199	7199	3599	3600	7199
FakeCovid	797	797	623	580	0	580	580
FakeNewsSet	2651	2651	2651	2617	1281	1335	2616
Fakepedia	15428	8887	7777	7777	3568	4174	7742
FakeRecogna	10898	10898	10898	10898	5449	5339	10788
FakeTweet.Br	299	299	298	298	97	199	296
FakeWhatsApp.Br	21289	21205	6926	6926	3298	3014	6312
MM-COVID	640	640	487	487	484	2	486
MuMiN	503	503	400	400	5	364	369
X-FACT	7941	7937	7922	5660	1781	3876	5657

tes classificadores são construídos no escopo da RQ3, quando para cada dataset ocorre a divisão de seus dados, com parte deles sendo destinados ao aperfeiçoamento dos modelos, e os demais, não utilizados no passo anterior, sendo reservados para a avaliação do classificador criado. Estas avaliações, com dados de mesma origem daqueles utilizados durante o treinamento, é o que chamamos nesta dissertação de *validação in-data*, que é a forma padrão de avaliação de desempenho de classificadores, e aqui será importante na comparação dos resultados obtidos com outras abordagens.

Para responder à RQ4, cada classificador avalia os dados dos outros datasets, desconhecidos pelo modelo até então, realizando o que chamamos de *validação cross-data*. Os detalhes sobre os experimentos envolvidos com as questões de pesquisa citadas se encontram na Seção 4.3.3. A Tabela 6 apresenta uma versão sintetizada do plano de execução dos experimentos contemplados pela metodologia proposta.

O intuito é verificar se os classificadores conseguem um bom desempenho sobre dados desconhecidos, seja porque foram treinados em determinada base, ou porque possuem conhecimento acumulado suficiente para isso. Nas próximas seções veremos quais são as métricas de avaliação escolhidas para os experimentos, detalhes sobre a construção de cada um deles, e a configuração adotada para execução.

Tabela 6: Plano de execução dos experimentos.

Questão de Pesquisa	**Tipo de Experimento**	**Modelo**	**Arquitetura**	**Estratégia**
RQ2	*zero-shot*	BERTimbau	*encoder*	Tarefa intermediária MLM
		mT5	*encoder-decoder*	Tarefa intermediária AR
		Command Sabiá	*decoder*	uso de *prompt* de instruções
RQ3	*in-data*	BERTimbau cohere-embeddings	*encoder*	*fine-tuning* e teste sobre dados de mesma origem dos dados do treino
		mT5	*encoder-decoder*	
RQ4	*cross-data*	BERTimbau cohere-embeddings	*encoder*	inferência com os classificadores gerados para RQ3 sobre dados de origem desconhecida
		mT5	*encoder-decoder*	

4.3.1 Formas de avaliação dos experimentos

Como os modelos seguem arquiteturas diferentes, as formas de avaliação de seus resultados também variam. Vamos apresentar a seguir como foi feita esta avaliação considerando,

além da arquitetura, a proposta de cada experimento e as métricas escolhidas para cada caso, cujas fórmulas são exibidas na Tabela 7. Consideramos como parâmetro de bons resultados F1 macro e medidas de similaridade textual iguais ou superiores a 80% nos experimentos *in-data* e iguais ou superiores a 70% nos experimentos *cross-data* e *zero-shot*, neste último caso quando aplicáveis.

Tabela 7: Métricas de avaliação utilizadas no estudo, onde a primeira é voltada para classificação e as demais para similaridade de texto. Para a Distância de Levenshtein, o método *head* seleciona apenas o primeiro caracter da sequência, enquanto o método *tail* seleciona todos os caracteres com exceção do primeiro.

Métrica	**Formulação**
F1	$\frac{2 \cdot precision \cdot recall}{(precision + recall)}$, onde $precision = \frac{TP}{TP+FP}$, e $recall = \frac{TP}{TP+FN}$
Similaridade cosseno	$\frac{x \cdot y}{\|x\| \, \|y\|}$, onde x e y são vetores que representam sequências de texto
Distância de Levenshtein	$lev(a,b) = \begin{cases} \|a\| & \text{se } \|b\| = 0, \\ \|b\| & \text{se } \|a\| = 0, \\ lev(tail(a), tail(b)) & \text{se } head(a) = head(b), \\ 1 + \min \begin{cases} lev(tail(a), b) \\ lev(a, tail(b)) \\ lev(tail(a), tail(b)) \end{cases} & \text{caso contrário} \end{cases}$ onde a e b são sequências de texto

4.3.1.1 Modelos *encoder*

Para os modelos da arquitetura BERT, por conterem uma camada de classificação, podemos utilizar métricas tradicionais de avaliação nos experimentos do tipo *in-data* e *cross-data*, como **acurácia** e **F1**. A classificação utilizando o modelo cohere-embeddings também fará uso destas métricas, dado que a API retorna a classe dos exemplos enviados na requisição. Na tarefa MLM, usada nos experimentos *zero-shot* com BERTimbau, tais formas de avaliação podem ser empregadas se o preenchimento da máscara das sentenças não for feito de maneira livre, ou seja, se os valores utilizados para o preenchimento das máscaras forem as classes. As peculiaridades deste tipo de experimento serão discutidas mais adiante.

A acurácia representa a quantidade de acertos de elementos da classe positiva (TP, do inglês *True Positive*) e da classe negativa (TN, do inglês *True Negative*) diante de todos os exemplos, que incluem além de TP e TN os erros de classificação para a classe positiva (FP, do inglês *False Positive*) e para a classe negativa (FN, do inglês *False*

Negative) (HARRISON, 2019). Contudo, o desbalanceamento da maioria dos datasets aqui trabalhados não pode ser ignorado.

Considerar apenas a quantidade de acertos não seria o mais indicado, isto porque a métrica F1 traz mais informação sobre como se dão estes acertos, com o cálculo da média harmônica entre a precisão ou *precision* (capacidade de encontrar somente os exemplos relevantes) e a sensibilidade ou *recall* (capacidade de encontrar todos os exemplos positivos, da classe de interesse) (HARRISON, 2019). Foi escolhido como representante de F1 o **F1 macro**, dado que ele avalia o desempenho independentemente da frequência de cada classe. Tanto para acurácia, quanto para F1, quanto mais próximos de 1 (um) são os resultados, melhores eles são. A acurácia será exibida nos experimentos *in-data* e *zero-shot*, mas apenas de maneira informativa.

4.3.1.2 Modelos *decoder*

Os modelos mT5, Command e Sabiá-3 são modelos de geração de texto, porém não operam da mesma forma. Command e Sabiá-3, que são do tipo *decoder*, recebem instruções explícitas sobre os valores de retorno esperados (com os valores "TRUE" ou "FALSE" para o modelo Command e "VERDADEIRO" ou "FALSO" para o modelo Sabiá-3), os quais podem ser pós-processados para refletir a classe predita por estes modelos, sendo possível utilizar as métricas **acurácia** e **F1 macro**.

Os dois modelos *decoder* foram acessados via API, o que os deixa suscetíveis a falhas, como também retornos diferentes do esperado e descrito nas instruções. Nestes casos, houve o descarte destes exemplos e eles não foram considerados no cálculo das métricas citadas.

4.3.1.3 Modelos *encoder-decoder*

Os modelos baseados no mT5, não recebem instruções; apenas um breve prefixo para apoiar o modelo no entendimento da tarefa a ser desempenhada. Por conta disso, a avaliação de seus resultados se deu por meio da similaridade entre o texto gerado pelo modelo e a classificação real de cada exemplo, uma vez que uma verificação de igualdade entre estes elementos não consideraria proximidade (seja do ponto de vista semântico ou mesmo gráfico), havendo uma perda de informação neste sentido. Para isso foram selecionadas duas medidas de similaridade de texto: a **similaridade cosseno** e a **distância de Levenshtein normalizada**. Estas métricas foram calculadas através da biblioteca

`textdistance`[2].

A similaridade cosseno calcula a distância entre dois vetores quaisquer x e y, cada um deles representando uma entrada de texto. A distância de Levenshtein calcula a distância de edição entre dois termos a e b, onde a forma normalizada garante que o retorno da métrica esteja sempre entre zero e um. Nos dois casos, valores mais próximos de 1 (um) indicam maior similaridade entre os textos.

4.3.2 Experimentos *zero-shot*

Os experimentos *zero-shot* deste estudo visam explorar as habilidades emergentes de modelos de linguagem pré-treinados na classificação de textos de notícias na esfera das *fake news* em português. As capacidades emergentes já vêm sendo observadas em *Large Language Models* (LLMs), como foi exposto em (BROWN et al., 2020), o que motivou a criação da **RQ2**.

O intuito da validação da capacidade de generalização de modelos sem nenhum tipo de ajuste vem do seguinte questionamento: *(i) A habilidade emergente de classificar textos observada nos LLMs também poderia ser observada em modelos menores?* Mais do que isso: *(ii) ela poderia ser observada em modelos que não fossem decoders?* Para a validação *zero-shot*, foram utilizados os mesmos modelos representantes de cada arquitetura: BERTimbau, mT5, Command e Sabiá-3, com a submissão de todos os exemplos pré-processados de cada base de dados.

4.3.2.1 Modelos menores (BERTimbau e mT5)

Para os modelos BERTimbau e mT5, a capacidade *zero-shot* se deu pelo preenchimento de máscaras adicionadas ao texto de cada notícia. A representação das máscaras varia um pouco para cada modelo, com [MASK] para o BERTimbau e <extra_id_0> para o mT5.

Para auxiliar neste processo, foram criados de maneira empírica **templates** com pequenos trechos de prefixo e sufixo, que adicionados aos textos poderiam ajudar os modelos a preencherem estas máscaras indicando uma classificação para as sentenças. Foi testada a adição da máscara tanto no início como ao final das sentenças. Nos Quadros 4.1 e 4.2 são exibidos os templates criados para os modelos.

[2] https://pypi.org/project/textdistance/

A escolha da posição das máscaras é interessante para verificar a capacidade de entendimento do contexto geral por parte dos modelos antes de efetuar a completação. O BERTimbau, por ser bidirecional, teoricamente não é afetado pela posição da máscara. O mT5 também não deveria sofrer grande impacto, uma vez que também possui *encoders* em sua estrutura.

Template 1-BERTimbau:
Verdadeiro ou falso: É [MASK] que **a Terra gira em torno do Sol**.

Template 2-BERTimbau:
Afirmar que "**a Terra gira em torno do Sol**"é [MASK].

Quadro 4.1: Templates usados no experimento *zero-shot* com o modelo BERTimbau.

Template 1-mT5:
Completar: É <extra_id_0> que **a Terra gira em torno do Sol**.

Template 2-mT5:
"Completar: Afirmar que " **a Terra gira em torno do Sol**" é <extra_id_0>."

Quadro 4.2: Templates usados no experimento *zero-shot* com o modelo mT5.

Configuração adotada A execução dos experimentos *zero-shot* para os modelos BERTimbau e mT5 se deu através do método `pipeline` da biblioteca HuggingFace[3]. Para utilizá-lo, é necessário informar os seguintes dados:

- o texto contendo a máscara a ser preenchida, de acordo com o template escolhido,
- a versão de implementação do modelo (exibidas na Tabela 8),
- a tarefa a ser executada, cujo tipo é representado por termos reservados. Para o BERTimbau, foi executada a tarefa intermediária MLM, representada pelo valor "fill-mask", e para o modelo mT5 foi configurada a execução da tarefa intermediária AR, representada pelo valor "text2text-generation".

Uma peculiaridade dos modelos da família BERT é que é possível passar como parâmetro deste mesmo método uma lista de termos a serem considerados para o preenchimento

[3] https://huggingface.co/

das sentenças, através do campo *targets*. Para explorar esta possibilidade, os dois *templates* criados para o BERTimbau foram executados de duas maneiras: (i) sem restrição de preenchimento, e (ii) informando como termos possíveis para preenchimento as palavras **verdadeiro** e **falso**.

Avaliação dos experimentos - BERTimbau Para cada sentença X enviada ao `pipeline` do BERTimbau, a tarefa MLM retorna os n primeiros valores na forma $V = \{(v_1,p_1), \ldots, (v_n,p_n)\}$, onde $v_i \in V$, V é o conjunto de *tokens* conhecidos pelo modelo (seu vocabulário), e p_i é o valor da pontuação obtida pela função de custo para cada *token* v_i substituir a máscara presente em X.

Quando nenhuma restrição de preenchimento da máscara é imposta, é escolhido o *token* com o maior valor de p. Caso contrário, é efetuada uma busca dentre os termos permitidos por aquele de maior pontuação. Quando há esta delimitação, é possível calcular as mesmas métricas de avaliação dos classificadores *cross-data* deste mesmo modelo.

Vale ressaltar que o *token* com maior probabilidade não necessariamente tem alguma relação com as classes presentes nas bases de dados, o que faz com que a avaliação se dê de forma qualitativa, onde vamos verificar quais foram os termos mais comuns para o preenchimento das máscaras dos exemplos conforme a classe original e o quão variado é este conjunto.

Avaliação dos experimentos - mT5 A avaliação dos resultados com mT5 também será qualitativa pelas mesmas razões, porém, por ser um modelo de geração de texto, o retorno para cada sentença não necessariamente terá um termo apenas, o que pode comprometer a análise desta abordagem.

4.3.2.2 LLMs (Command e Sabiá-3)

A execução deste experimento para os modelos Command e Sabiá-3 foi por meio de *prompt engineering*, com o envio de instruções aos modelos para a classificação dos textos de cada exemplo. O Quadro 4.3 mostra o *prompt* submetido à API de cada modelo. Foram desenvolvidas variações das instruções até que elas fossem claras o suficiente para que os modelos respondessem da forma solicitada.

O *prompt* final do modelo Command foi submetido em inglês, com a intuição de que como a linguagem da maioria dos dados utilizados nos treinamentos dos LLMs é o inglês, é

preferível que ele absorva corretamente a instrução e depois traduza o texto internamente para efetuar a classificação solicitada (CHANG et al., 2024).

Seguindo o mesmo critério, o *prompt* final do modelo Sabiá-3 foi submetido em português, dado que o modelo foi treinado em português de forma expressiva e todos os textos aqui trabalhados também estão em português. O seu conjunto de instruções consiste na versão traduzida do *prompt* adotado para o modelo Command, com a inclusão de uma sentença ao final, que se mostrou necessária durante as primeiras explorações, como forma de evitar que os retornos fossem muito extensos. Por se tratar de um experimento com custo envolvido, não exploramos o impacto do idioma nos resultados encontrados.

Prompt do modelo Command:
Please respond with TRUE if the following statement is likely to be accurate or reliable or with FALSE if the statement is likely to be wrong or an attempt at misinformation.

Prompt do modelo Sabiá-3:
Responda com VERDADEIRO se a afirmação a seguir for provavelmente precisa ou confiável ou com FALSO se a afirmação for provavelmente errada ou uma tentativa de desinformação.
Responda apenas VERDADEIRO ou FALSO.

Quadro 4.3: Prompts usados no experimento *zero-shot*, com o modelo Command consumindo a versão em inglês e o modelo Sabiá-3 a versão em português, que é uma tradução da versão em inglês, exceto pela última frase, destacada em itálico.

Configuração adotada A execução deste experimento para os modelos do tipo *decoder* se deu a partir da criação de requisições para cada exemplo de cada dataset pré-processado, sendo realizado o envio de um exemplo por vez. Cada requisição carregou como instrução os prompts apresentados no Quadro 4.3 ao *endpoint* da plataforma de cada modelo, sendo todas do tipo POST.

Para a utilização do modelo Command, foram criadas requisições para o método `chat`, que direcionam para o endpoint `https://api.cohere.com/v1/chat`. Já para a utilização do modelo Sabiá-3, o processo ocorreu de maneira análoga, enviando além do *prompt* (por meio do *role system*), o texto de cada exemplo (por meio do *role user*) para o endpoint `https://chat.maritaca.ai/api/chat/inference`.

4.3.3 Validação *in-data* e *cross-data*

De maneira formal, dizemos que: Considerando um conjunto $\mathbf{D} = \{D_1, D_2, \ldots, D_z\}$ de datasets e um conjunto de modelos de linguagem $\mathbf{M} = \{M_1, M_2, M_w\}$, cada dataset $D_i \in \mathbf{D}$ é dividido em conjunto de treinamento T_i, conjunto de validação V_i e conjunto de teste Te_i. A seguir, é executado o *fine-tuning* a partir de um modelo de linguagem $M_k \in \mathbf{M}$, usando T_i para ajuste dos pesos e V_i para a parada antecipada, gerando um classificador C_{ki}. Para a validação *in-data*, C_{ki} é usado para testar Te_i, enquanto para a validação *cross-data*, C_{ki} é usado para classificar Te_j de uma base $D_{j,j\neq i} \in \mathbf{D}$.

Embora esta seja a ideia central das duas validações, o modelo cohere-embeddings, diferentemente do BERTimbau e mT5, que são disponíveis gratuitamente para uso local, é fechado e com custo de acesso envolvido. Para implementar estas ideias, foi necessário preparar os dados e o treinamento de modos diferentes. Nas próximas seções abordaremos quais foram estas diferenças, bem como os impactos envolvidos.

4.3.3.1 Configuração de *fine-tuning* para a validação *in-data* dos modelos BERTimbau e mT5

Para que o *fine-tuning* dos modelos BERTimbau e mT5 pudessem gerar classificadores mais generalizáveis, foi efetuada uma validação cruzada com 5 (cinco) *folds*, estratificados por classe. A cada iteração, novos exemplos compuseram a parcela de dados designada para testes, fazendo com que ao final todos os exemplos fossem testados. O resultado final de cada classificador foi a média dos resultados destas iterações. Desta forma, para cada modelo (BERTimbau e mT5), foi possível gerar um classificador especializado em cada dataset. A Tabela 8 mostra a parametrização de treinamento destes modelos, bem como as versões de implementação escolhidas e outros dados adicionais. Dentre eles, está o valor fixado como *semente* para permitir a reprodutibilidade dos experimentos, através do campo *random_state*. Estes classificadores foram treinados utilizando uma máquina virtual com acesso cedido pela universidade, a NVIDIA DGX-1 Tesla P100-SXM2-16GB.

Como o modelo mT5 faz uso de instruções curtas, todos os exemplos receberam como prefixo o texto *"Responda verdadeiro ou falso:"* antes de serem submetidos ao treinamento. Tanto o BERTimbau quanto o mT5 suportam como tamanho máximo de entrada 512 *tokens*, havendo o truncamento de seu conteúdo quando este limite é atingido.

Para a geração de cada um destes classificadores, foram utilizados os dados dos *folds* de treino, onde deste total foram separados 10% dos exemplos para formarem o conjunto

de validação. Este conjunto tem como função calibrar o desempenho considerando os valores da função de perda (ou *loss function*) para a parada antecipada, caso os valores desta métrica não evoluam por cinco épocas consecutivas, evitando assim a execução desnecessária de épocas.

Tabela 8: Parametrização básica adotada e versões de implementação escolhidas para criação dos classificadores baseados nos modelos de linguagem BERTimbau e mT5.

	Valor adotado
Parâmetro configurado	
random_state	42
tamanho do batch	16
número de épocas	dez
taxa de aprendizagem	2e-5
otimizador	adamw_torch
weight_decay	0.01
Modelo	**Versão de implementação**
BERTimbau	neuralmind/bert-base-portuguese-cased
mT5	google/mt5-small

4.3.3.2 Configuração de *fine-tuning* para a validação *in-data* do modelo cohere-embeddings

Para o modelo cohere-embeddings, não foi possível efetuar a validação cruzada, dado que isto exigiria mais recursos computacionais frente a um número muito maior de experimentos, considerando o número de iterações e a quantidade de datasets. A alternativa encontrada foi a execução do *fine-tuning* através do *holdout* dos dados pré-processados, isto é, a divisão dos exemplos dos datasets em conjuntos, sem alternância de exemplos entre eles.

Com isso, conseguimos aplicar as validações propostas, porém, como a divisão dos dados é efetuada uma única vez, não há como saber se a composição dos conjuntos criados é suficientemente representativa. A seguir, será apresentada a forma de separação dos dados adotada e os detalhes envolvendo a criação dos classificadores deste modelo via API.

Separação dos dados em conjuntos Os dados pré-processados de cada dataset foram divididos em conjuntos de treino, validação e teste, com respectivamente 70%, 10% e 20% dos dados cada. Cada porção gerada manteve a distribuição original de elementos por classe, para que elas fossem minimamente representativas do conjunto de origem, além de ser realizado um embaralhamento dos dados de cada conjunto antes desta divisão.

Ao final do pré-processamento, a base de dados MM-COVID apresentou apenas dois exemplos para a classe "falso", e por conta desta baixa representatividade para uma das classes não pôde ser dividida nestes três conjuntos, estando inapta a treinar os classificadores do modelo cohere-embeddings, ficando restrita aos testes.

Montagem das requisições do modelo cohere-embeddings A utilização do modelo cohere-embeddings se dá através da chamada ao *endpoint https://api.cohere.com/v1/classify*, porém ele não pode ser acionado diretamente, porque isso implicaria em uma classificação por busca semântica sem nenhuma especialização do modelo com os dados dos datasets aqui trabalhados. Para efetuar o *fine-tuning* de um modelo na plataforma Cohere, primeiro é preciso criar os datasets, através do *endpoint https://api.cohere.com/v1/datasets*. Os dados pré-processados dos datasets, antes dispostos no formato `.csv` precisaram ser convertidos em arquivos no formato `.jsonl`.

Para que um dataset na plataforma possa ser usado para *fine-tuning*, é exigido que cada classe tenha pela menos cinco exemplos associados, o que impossibilitou a criação de classificadores treinados com os datasets FakeCovid e MuMiN, sendo enviados os conjuntos de treino e validação de 11 dos 14 datasets trabalhados. Uma limitação encontrada é que a API não permite a criação de mais de dez datasets por dia, sendo preciso distribuir o trabalho em alguns dias, dada a criação prévia de pequenos datasets para o aprendizado de manuseio das capacidades da plataforma e seu funcionamento.

O próximo passo foi o aperfeiçoamento do modelo em si, que se dá por um outro *endpoint*, o *https://api.cohere.com/v1/finetuning/finetuned-models*, que recebe o identificador do dataset enviado no estágio anterior e o nome do modelo que deve ser aperfeiçoado. Como já foi dito no final da Seção 2.2.2, cohere-embeddings foi o termo adotado por nós para facilitar a referenciação ao longo do texto, porém o nome real da implementação do modelo é *embed-multilingual-v2.0*, sendo até o presente momento a única versão multilíngue disponível para uso, com os demais *encoders* restritos a textos em inglês.

Após isso, foram gerados 11 classificadores, cada um especializado em um dataset informado. Para efetuar as predições, foram montadas requisições ao método `classify`,

informando o identificador do classificador e os exemplos dos conjuntos de teste. Ocorre que só podem ser submetidos 96 exemplos de teste por requisição, enviados no campo *inputs* como uma lista. Isso fez com que fosse preciso enviar os dados em blocos, totalizando 1470 requisições, incluindo validações *in-data* e *cross-data*.

Nas requisições para classificação dos exemplos de teste, também foram informados exemplos para fornecer informação de contexto, uma vez que este modelo efetua a classificação por meio de busca semântica. Para fornecer o contexto, foram informados dados do conjunto de treino do dataset que estava submetendo os exemplos de teste. A plataforma permite o envio de 2500 itens para contextualização, que devem ser enviados como uma lista por meio do campo *examples*. Na construção das requisições, sempre enviamos a quantidade máxima possível, com registros escolhidos de forma aleatória. Quando se fazia necessário enviar mais de uma requisição para submissão dos exemplos de teste, o mesmo conjunto de exemplos selecionados para contexto era enviado, para que todos os exemplos de teste fossem expostos às mesmas condições.

4.3.3.3 Configuração de validação *cross-data*

Conforme dito na seção anterior, para o modelo cohere-embeddings, após a montagem da estrutura necessária, para efetuar as validações, bastava enviar as requisições para o *endpoint* `classify` informando modelo e os exemplos dos conjuntos de teste. Logo, o que diferenciava se a validação era *in-data* ou não era basicamente se aquele conjunto de teste era da mesma base utilizada para treinar o modelo chamado ou não. As validações *cross-data* se deram apenas com os dados dos conjuntos de teste, uma vez que enviar todos os exemplos das demais bases implicaria em um número muito maior de requisições.

Em relação aos modelos BERTimbau e mT5, para a validação *cross-data* ser efetuada, foi verificado qual havia sido o melhor classificador gerado por cada modelo em determinada base de dados de treinamento, visto que cada iteração gerava um modelo diferente, totalizando cinco ao todo. Como critério, foram selecionados os modelos com melhor desempenho para F1 macro e similaridade cosseno.

A validação *cross-data* realizada por um classificador treinado por determinada base de dados, considera todos os exemplos pré-processados das demais bases, assim como ocorre nos experimentos *zero-shot*, possibilitando a comparação dos resultados das abordagens. Para exemplificar, na busca pelo melhor classificador BERTimbau treinado com o dataset FakeNewsSet, é verificado o resultado obtido para F1 macro para cada um dos cinco *folds* de teste. A versão do classificador que alcançou o maior valor é selecio-

nada para classificar os exemplos de todas as bases de dados diferentes da sua base de treinamento, logo, diferentes de FakeNewsSet.

5 Análise dos conjuntos de dados

A Seção 4.1.1 apresentou as bases de dados sobre *fake news* disponíveis na língua portuguesa, selecionadas para responder a **RQ1**, que diz o seguinte:

> Que bases de dados existem para a classificação de notícias falsas em português e quais são as suas características principais que podem influenciar no desempenho dos classificadores?

Este capítulo estende a resposta para a **RQ1**, explorando e analisando as principais características das bases encontradas.

5.1 Distribuição e classificação de notícias

Durante a análise dos dados disponíveis publicamente nos repositórios, observou-se que muitos exemplos estão duplicados ou contêm dados faltantes em atributos-chave, sendo necessário descartá-los. É importante frisar que a dissertação se ateve ao que foi observado nos dados disponibilizados, que, por estarem sujeitos a mudanças e atualizações por seus autores, podem conter informações que divergem de seus artigos originais.

Como apresentado na Tabela 5, a duplicidade de exemplos foi observada em cinco datasets, com o dataset Fakepedia sofrendo a maior redução por conta deste problema, de 42,40%. Foram detectadas outras duplicações, tanto de texto, que levou à eliminação de 67,47% dos registros do dataset FakeWhatsApp.Br, quanto novas que surgiram após o pré-processamento do texto dos exemplos, principalmente após a remoção de espaços.

Após a coleta e transformação dos datasets, já se desenhava um cenário de muita discrepância em relação ao tamanho destes conjuntos de dados, e, ao final da Etapa 2, a quantidade de elementos entre eles oscilou ainda mais, o que pode influenciar na desenvoltura dos classificadores gerados a partir de cada dataset. Para ilustrar esta distribuição entre as Etapas 1 e 2, os datasets foram agrupados em cinco intervalos que representam a

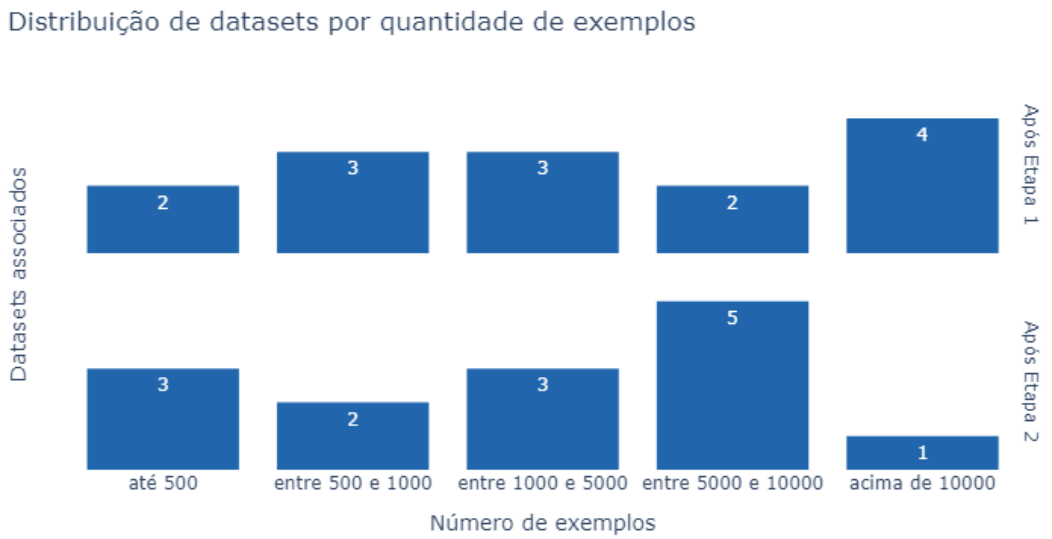

Figura 7: Comparativo entre as Etapas 1 e 2 da quantidade de datasets por intervalo de exemplos disponíveis. Após a Etapa 1, temos os dados originais dos datasets com pequenas transformações quando necessário. Após a Etapa 2, os dados passaram por alguns tratamentos, o que pode resultar na eliminação de exemplos.

quantidade de exemplos disponíveis de cada, exibida pela Figura 7. Estes intervalos têm o seu início inclusivo e o final exclusivo, começando com o número de datasets com até 500 exemplos (não inclusivo) e fechando com o intervalo contendo o número de datasets com mais de 10.000 exemplos associados.

Na distribuição final, dos quatro datasets que tinham mais de dez mil exemplos, apenas um se manteve nesta faixa e outros três datasets terminaram com menos de 500 exemplos, A Figura 8 mostra a distribuição dos datasets selecionados no conjunto final de dados. Nela é evidenciado que, apesar de termos várias bases distintas, poucas são representativas em termos de volume de dados, uma vez que quase 50% dos datasets respondem a menos de 2% do número total de exemplos cada. Tal fato dá margem para se pensar que talvez combinar datasets pequenos poderia equilibrar este cenário, porém isso geraria novas implicações a serem consideradas, o que foge do escopo determinado neste momento.

Analisando a distribuição de classes, a maioria dos datasets exibe desbalanceamento entre as classes, como mostrou a Tabela 4, chegando ao final da Etapa 2 (vide Tabela 5) com 61,5% dos exemplos para notícias falsas e 38,5% dos exemplos formado por notícias verdadeiras. Este cenário pode gerar aos modelos uma tendência em predizer a classe "falso", visto que foi o que eles mais tiveram informação durante o treinamento. Apenas quatro datasets têm maioria de exemplos da classe "verdadeiro". Em outros dois, as classes

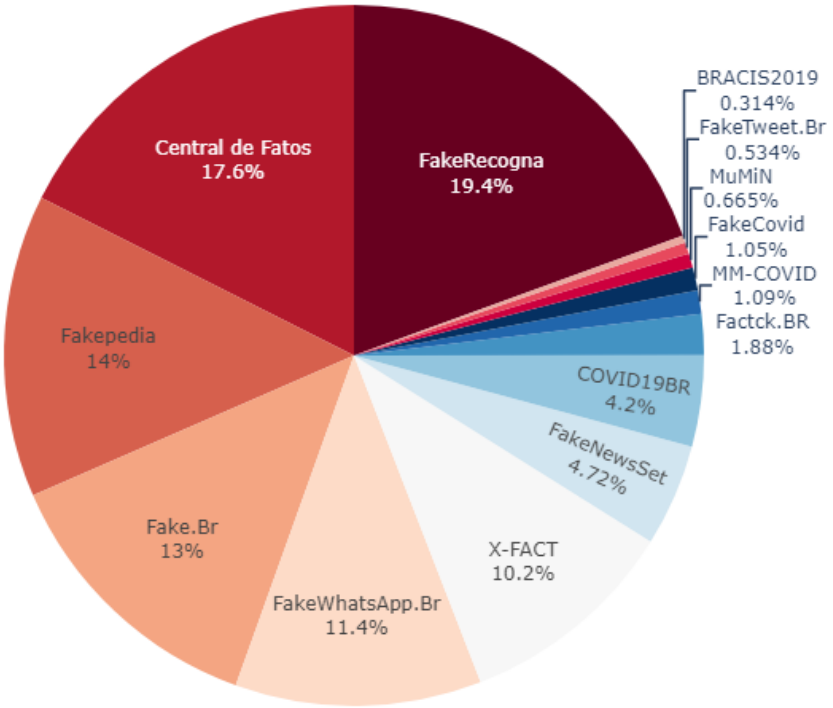

Figura 8: Distribuição final de notícias por dataset. Os três conjuntos com mais dados associados respondem por 51% de todos os registros disponíveis, mostrando a distribuição desigual de informação entre os conjuntos de dados.

são totalmente balanceadas.

Apesar de datasets balanceados serem mais propícios para o aprendizado de classificadores, este balanceamento não reflete o mundo real, pois notícias falsas e verdadeiras não são observadas na mesma proporção (PEI et al., 2023). Por outro lado, não é trivial observar a proporção do mundo real: se por um lado, o senso comum dita que existem mais notícias verdadeiras do que falsas, com a produção de desinformação em larga escala, pode ser que tal senso comum esteja deixando de ser verdadeiro.

Ademais, a discrepância entre as classes é compatível com os portais de verificação de fatos, a principal fonte de dados para os datasets: a maioria das checagens de fatos são concluídas com a indicação de *fake news*, porque é o tipo de conteúdo de maior interesse para verificação. A Figura 9 mostra a distribuição geral dos exemplos dos datasets por fontes de dados, onde diversos portais foram organizados em cinco categorias representativas.

A origem dos dados é um fator que pode influenciar nos resultados dos experimentos, principalmente aqueles com validação *cross-data*, dado que textos de fontes semelhantes poderiam se aproximar dos resultados obtidos pelos classificadores com dados próprios. Destaca-se a mescla de fontes em alguns datasets, o que pode causar ruído nos dados, e

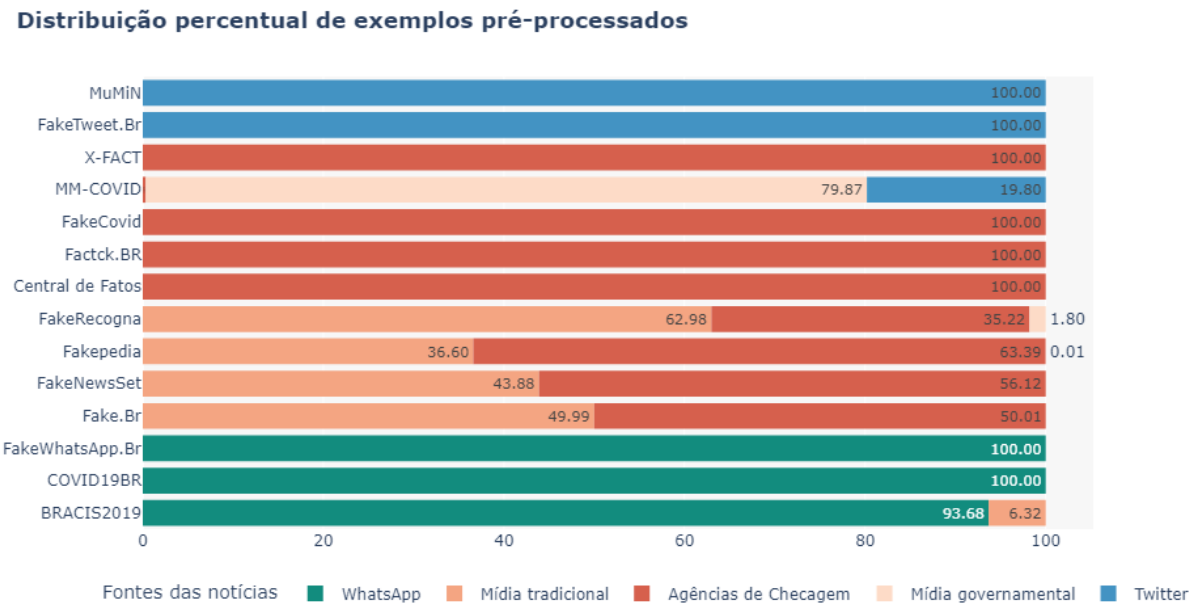

Figura 9: Distribuição dos datasets por fonte de notícia após a Etapa 2 ser concluída.

também a forte presença de textos de órgãos governamentais na base MM-COVID. Esta base é formada por textos com pequenos informes sobre atualizações da pandemia de Covid-19.

5.2 Análise dos textos das notícias

As diferentes fontes utilizadas para coletar os datasets acabam por fazer com que eles apresentem características distintas. Por exemplo, bases de *tweets* naturalmente contêm textos muito curtos, acarretando em informação descontextualizada. Textos obtidos de redes sociais, para que não caiam neste problema, dependem muito da forma como são selecionados.

Os exemplos deste tipo do dataset BRACIS2019, diferentemente de outros que também têm textos originários do WhatsApp, foram coletados a partir de verificações de fatos, por meio das quais se extraiu apenas o teor verificado. Isso faz com que seus textos naturalmente tenham passado por uma espécie de curadoria sobre a existência de alegação passível de ser verificada, representando assim uma parcela da desinformação propagada nesta rede. No exemplo a seguir desta mesma base, é nítida a natureza da informação transmitida:

> *CRIMINOSOS INTERNACIONAIS VIERAM DEIXAR DINHEIRO PARA CAMPANHA DE HADDAD. Na tarde de 14 de setembro de 2018, equipe*

> *da receita federal de Viracopos (Campinas), apreende 50 milhoes de dolares em 19 maletas Louis Vitton, vindas como bagagem acompanhada de "Teodorín Obiang" filho do ditador de Guiné Equatorial, acredita-se que a quantia apreendida seria para campanha de Hadad"...*

Por outro lado, algumas bases demonstram escassez do conteúdo original verificado. Para os textos de notícias verificadas, os datasets Factck.BR e FakeNewsSet são os únicos que trazem além das informações da publicação da checagem, como título e corpo da notícia, a alegação original, presente no campo *claimReviewed*, que segue uma arquitetura voltada para reconhecimento por meios de busca de portais voltados para verificação de fatos[1]. Um dos exemplos de FakeNewsSet contém o seguinte texto: *Papa Francisco cancela a Bíblia e propõe criar um novo livro.* Embora esta alegação esteja em terceira pessoa, o que pode causar algum estranhamento em relação aos textos de outras informações falsas divulgadas, ele é coeso e passível de verificação, além de não possuir um juízo de valor envolvido.

Já outros datasets, se fazem valer dos textos coletados para indicar *fake news*, o que nem sempre funciona conforme esperado. No dataset Fakepedia, que possui notícias extraídas da agência Boatos.org, temos tanto textos similares aos existentes em BRACIS2019, como outros que não configuram um conteúdo verificável, como no texto *e WhatsApp no telefone (61) 991779164.*

Entretanto, mesmo aquelas que contêm os fatos verificados, podem incluir conteúdo tendencioso, com o discurso sendo de alguém informando sobre o que já aconteceu e dizendo reiteradas vezes que aquela história não é verdadeira. Ou seja, o exemplo pode falar sobre um conteúdo falso, mas não deixa explícito qual é. No dataset FakeRecogna, em um de seus exemplos, o texto já traz o veredito sobre a notícia, sem informar qual mensagem estava sendo propagada originalmente, como pode ser visto a seguir:

> *É enganoso tuíte que associa a morte de um ator indiano à imunização pela vacina Covaxin, ocorrida dois dias antes. O artista sofreu um infarto e a equipe médica que o atendeu descartou ligação entre os fatos, conforme divulgado amplamente pela imprensa indiana. Cardiologistas ouvidos pelo Comprova afirmam que a vacina não forma trombos que levam à doença cardíaca.*

Para a criação dos modelos, este pode ser considerado como um ponto de atenção, porque notícias falsas e boatos buscam parecer reais para alcançar mais pessoas, enquanto

[1] https://developers.google.com/search/docs/appearance/structured-data/factcheck

textos jornalísticos sobre um tema falso podem não trazer de maneira tão expositiva tais características em seu conteúdo. Além destes fatores, há problemas na forma como o conteúdo destes datasets é obtido, onde encontramos casos como textos de notícias contendo nomes de opções dos menus de navegação destes sites, mensagens de erro no momento da coleta dos dados, ou ainda todas as mensagens da caixa de comentários dos leitores para determinada publicação, dentre outros casos.

Algumas bases de dados incluem textos que tentam mascarar o que seria o texto original por meio de numerais substituindo vogais e a utilização de outros termos para fazer referência a um tema conhecido. Por exemplo, na base de dados COVID19BR, há referências ao Corona vírus como "coronga vairuz", algo que poderia não ser detectado por algum filtro de busca por conteúdo suspeito nas redes sociais, por exemplo. Estes casos podem indicar que novas formas de se falar sobre um tema são criadas e que os mecanismos de regulação ou mesmo modelos criados para detecção precisam se atualizar para conseguir detectar essas variações na comunicação.

5.2.1 Análise estatística dos textos

Para cada dataset foi calculada a média de tamanho dos textos, a quantidade média de palavras por sentença, o tamanho médio por palavra, assim como as quantidades mínima e máxima de palavras por sentença encontradas em cada dataset. A Tabela 9 traz estas informações, comparando com os valores antes e após o tratamento dos dados.

Sobre o tamanho dos textos, há problemas nos dois sentidos: textos curtos demais que não expressam uma ideia candidata a alegação, ou textos muito longos que não parecem traduzir a essência das *fake news* habituais. A base Fake.Br detém os maiores textos, com uma média de 3899 caracteres cada. O dataset X-FACT (GUPTA; SRIKUMAR, 2021) se encontra no campo oposto, com textos com uma média de 101 caracteres. Esta base não informa o endereço da alegação, o que impossibilita acessar o site de onde ela foi obtida, principalmente quando são muito curtas, não sendo possível saber se havia mais informação a ser coletada.

Para os datasets com médias mais baixas de palavras por sentença, vimos que muitos textos estão cortados, seja com a inclusão de reticências ao final, como na sentença classificada como verdadeira *Num post que acumula milhares de partilhas afirma-se que o conhecido .....*, ou entre parênteses no meio do texto, como na sentença classificada como falsa *"Eu queria lembrar que quando eu fui governador de São Paulo (...), a Universidade Virtual do Estado de São Paulo tinha 3 mil alunos. Nós passamos para 50 mil alunos em*

Tabela 9: Valores de média sobre os textos selecionados de cada dataset considerando o tamanho dos textos, a quantidade de palavras e o tamanho médio de palavra.

	Com dados originais					**Após pré-processamento**				
Dataset	**Tamanho médio do texto**	**Qtde média de palavras**	**Tamanho médio de palavra**	**Menor qtde de palavras encontrada**	**Maior qtde de palavras encontrada**	**Tamanho médio do texto**	**Qtde média de palavras**	**Tamanho médio de palavra**	**Menor qtde de palavras encontrada**	**Maior qtde de palavras encontrada**
BRACIS2019	558	93	5	5	643	563	93	5	5	637
Central de Fatos	3776	620	5	8	8039	3540	585	5	8	7979
COVID19BR	753	109	26	1	5938	781	128	5	2	5868
Factck.BR	137	23	5	3	1599	120	20	5	3	776
Fake.Br	3899	643	5	9	7517	3890	641	5	9	7516
FakeCovid	4468	684	6	170	4574	4378	683	5	168	3282
FakeNewsSet	114	20	5	1	777	114	20	5	2	776
Fakepedia	5585	901	5	0	80484	4819	781	5	2	80484
FakeRecogna	848	139	5	0	2598	846	138	5	2	2597
FakeTweet.Br	207	30	6	3	55	160	27	5	2	53
FakeWhatsApp.Br	98	15	8	1	5486	690	115	5	6	2726
MM-COVID	71	11	6	3	40	81	12	6	3	46
MuMiN	173	24	6	4	52	144	23	5	4	49
X-FACT	100	17	5	1	265	101	17	5	2	265
Média	1485	238	7	15	8433	1445	234	5	15	8075

Tabela 10: Média de incidência de símbolos e palavras maiúsculas nas textos dos exemplos. Só foram consideradas em maiúsculo as palavras com mais de dois caracteres.

Dataset	**Hash-tag**	**Reti-cências**	**Excla-mação**	**R$**	**Percen-tual**	**Interro-gação**	**Palavras em maiúsculo**
BRACIS2019	0,06	0,22	0,35	0,1	0,08	0,16	0,7
Central de Fatos	0,19	0,26	0,42	0,13	0,14	0,64	0,93
COVID19BR	0,06	0,17	0,25	0,06	0,07	0,18	0,57
Factck.BR	0,02	0,07	0,13	0,05	0,06	0,0	0,28
Fake.Br	0,02	0,28	0,25	0,27	0,18	0,31	0,9
FakeCovid	0,06	0,15	0,22	0,3	0,31	0,44	0,99
FakeNewsSet	0,02	0,04	0,07	0,06	0,06	0,0	0,58
Fakepedia	0,04	0,26	0,33	0,13	0,14	0,28	0,8
FakeRecogna	0,12	0,12	0,15	0,07	0,11	0,12	0,7
FakeTweet.Br	0,17	0,33	0,28	0,04	0,01	0,19	0,48
FakeWhatsApp.Br	0,09	0,22	0,43	0,04	0,07	0,19	0,62
MM-COVID	0,15	0,14	0,0	0,03	0,02	0,02	0,22
MuMiN	0,18	0,06	0,12	0,03	0,04	0,08	0,37
X-FACT	0,01	0,44	0,06	0,03	0,05	0,04	0,23

oito meses", o que indica a subtração de trechos do texto original, neste caso um provável discurso político. Um ponto a ser observado é que, embora alguns datasets compartilhem fontes de notícias, as estatísticas de seus textos mostram que a definição do que deve ser considerado como texto varia entre eles.

Para exemplificar, os datasets Fake.Br e FakeNewsSet têm as fontes gerais de notícia e são balanceados, contudo a média de palavras é muito distante. Isso indica que provavelmente textos curtos de mídia tradicional, o que também se aplica a instituições governamentais, são frutos de coletas apenas do primeiro parágrafo das notícias, o que no jornalismo é conhecido como *lead* e costuma condensar as informações essenciais que devem ser transmitidas, embora não seja uma regra seguida por todos os veículos. A Tabela 10 apresenta a verificação da ocorrência de algumas marcas textuais no conteúdo das notícias, com o intuito de descobrir algum comportamento relacionado, por meio da incidência de reticências, *hashtags*, sinais de pontuação e palavras escritas em caixa alta.

O uso de *hashtags* traz palavras significativas para a mensagem, o que pode contribuir com a semântica, porém na tokenização para uso pelos classificadores, seus termos associados acabariam avulsos ao longo do texto, o que pode comprometer a conexão entre as palavras nos textos. Nos experimentos de preenchimento de máscaras, este símbolo gráfico também pode comprometer a distinção do modelo entre os *tokens*, dado que os que contêm sub-palavras começam com o mesmo símbolo. Já era esperado que datasets com dados do Twitter liderassem neste quesito, porém as bases FakeRecogna e Central de Fatos aparecem com níveis altos, o que pode trazer ruído aos modelos ao receberem seus

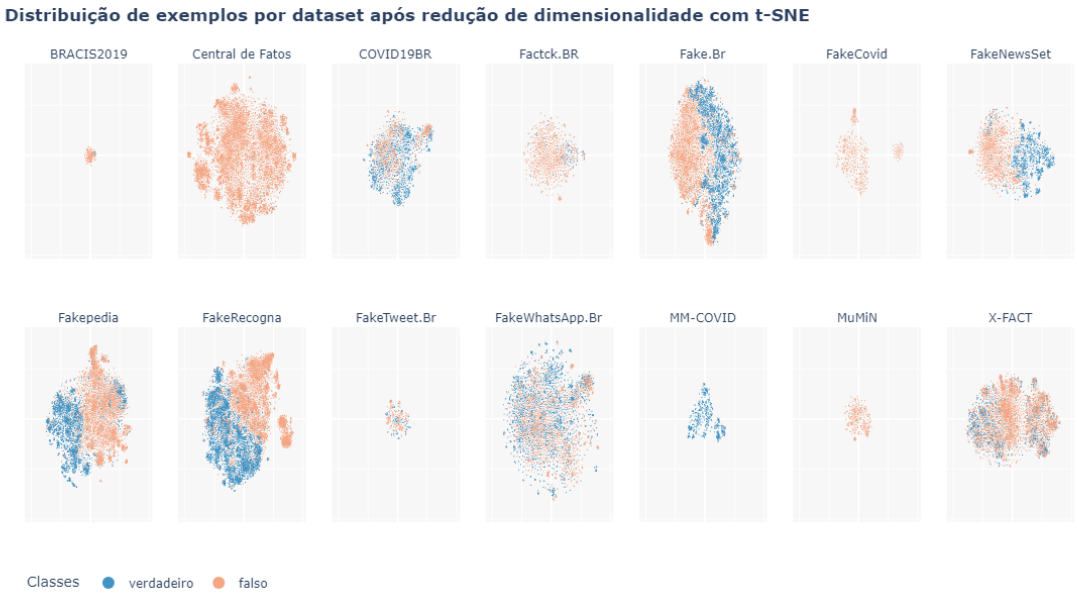

Figura 10: Distribuição de notícias por dataset e classificação após utilização do algoritmo t-SNE.

dados.

Como vimos antes, as reticências em vários casos indicam subtração de conteúdo, o que compromete o entendimento dos textos. A base X-FACT conta com mais de 5000 exemplos, porém quase metade deles possui reticências, o que deve comprometer a performance dos modelos. As ocorrências de "R$" e "%" ocorrem quase que de maneira regular, o que indica que é comum nos textos citações a valores monetários e percentuais.

5.2.2 Disposição semântica dos datasets

Para verificar a distribuição dos datasets sob o ponto de vista semântico, foram gerados *embeddings* contextualizados com o modelo BERTimbau e em seguida o algoritmo t-SNE (*t-distributed Stochastic Neighbor Embedding*) foi aplicado para redução de dimensionalidade (no nosso caso, para duas dimensões, informada por meio do hiper parâmetro *n_components*) (VAN DER MAATEN; HINTON, 2008).

O modelo t-SNE é uma técnica não linear e não supervisionada para redução de dados com muitas dimensões, que consiste essencialmente em preservar pequenas distâncias entre pares de instâncias em espaço dimensional superior e inferior a partir de determinado número de vizinhos (VAN DER MAATEN; HINTON, 2008), definido na criação do modelo. Neste trabalho, foi considerada a quantidade padrão de dez vizinhos, representada pelo hiper parâmetro *perplexity*.

A Figura 10 exibe a dimensão de cada dataset pela quantidade de pontos, mas também de que forma notícias falsas e verdadeiras se distribuem no espaço. As bases Fakepedia e FakeRecogna têm notícias verdadeiras mais à esquerda, enquanto em Fake.Br e FakeNewsSet os lados se invertem. Já os datasets COVID19BR e FakeWhatsApp.Br têm seus pontos muito misturados, e por se tratarem de bases de mensagens da mesma rede social, a partir desta abordagem não é perceptível observar se os exemplos de diferentes classes são separáveis.

A partir da análise da representação semântica presente na Figura 10, verifica-se que não se formaram grupos bem definidos para cada classe, cuja observação pode indicar a necessidade de formas melhores de representação de textos deste domínio, assim como evidencia a dificuldade em detectar *fake news* apenas pelo seu conteúdo.

5.2.3 Comparação léxica dos datasets

Para alinhar o desempenho dos classificadores a serem treinados com as bases, são reportadas nesta seção a frequência de palavras no texto, exibindo os termos mais frequentes e os mais raros para cada classe. A seguir, os datasets são comparados quanto à proximidade léxica por meio do índice de similaridade Jaccard. Para estas análises, as palavras foram passadas para a sua forma de escrita em letra minúscula e aquelas sem relevância semântica, as chamadas *stopwords*, foram descartadas. Os termos mais e menos frequentes estão representados nas Figuras 11 e 12, e a similaridade entre os datasets na Figura 13.

De maneira geral, com base nos termos mais frequentes, os textos dos datasets são voltados para política e redes sociais. Embora tenhamos algumas bases de dados com notícias apenas sobre Covid-19, causa estranheza que termos relacionados à pandemia não estejam entre as dez palavras mais frequentes das classes, nos levando à reflexão de que mesmo quando o assunto é saúde, o meio é político. Outras evidências nesta direção são a presença de nomes próprios, como os de ex-presidentes da república brasileira e o termo “paulo”, que nos textos está relacionado tanto a localidade São Paulo, quanto ao ex-ministro da fazenda.

Durante o levantamento de *stopwords* adicionais, removemos menções a números dos textos, que se referiam em sua maioria a quantias de dinheiro. O termo “r$600,00” decidimos manter, por ser o mais expressivo em aparição, e por ter um contexto próprio, visto que foi o valor liberado pelo governo federal para o pagamento de auxílio emergencial à

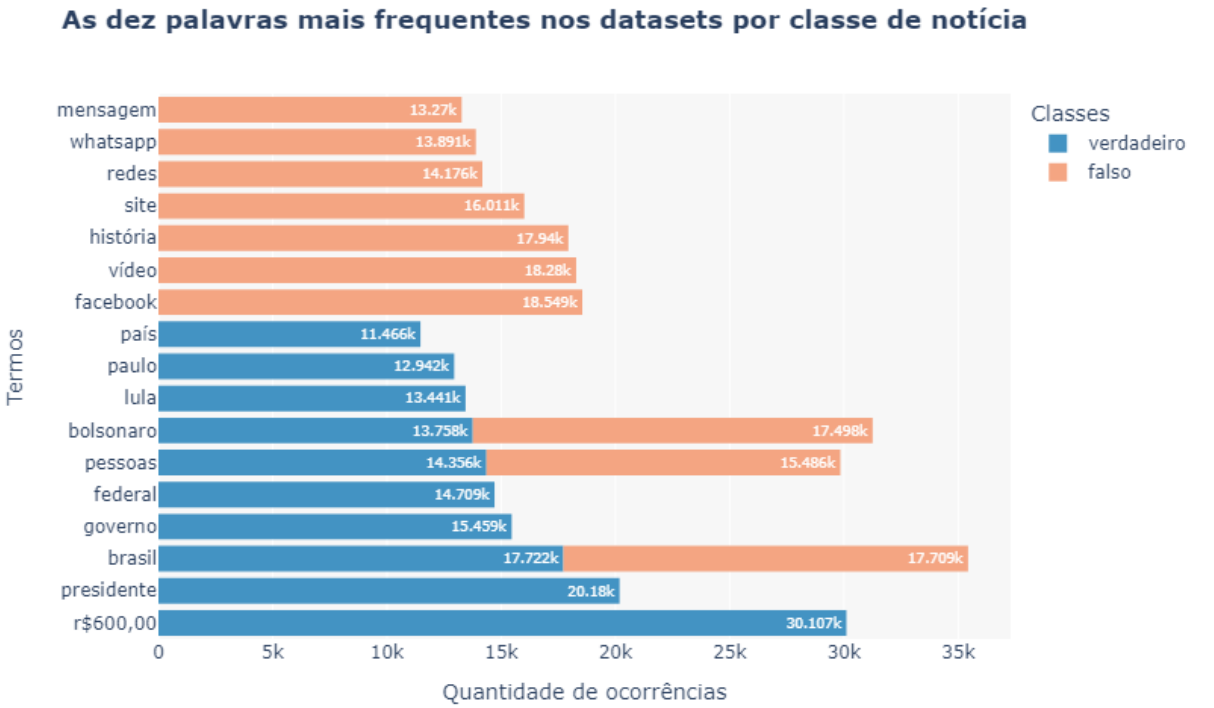

Figura 11: As dez palavras mais frequentes nas notícias verdadeiras e falsas de todo o conjunto de datasets trabalhado.

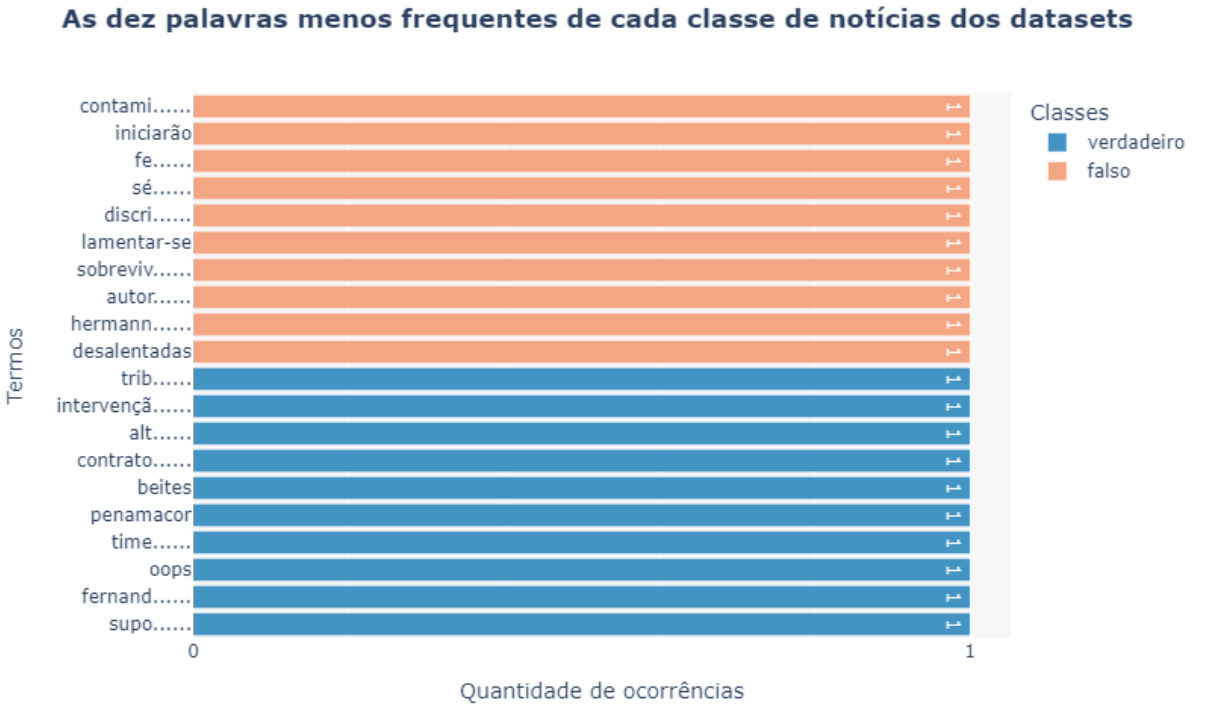

Figura 12: As dez palavras menos frequentes nas notícias verdadeiras e falsas de todo o conjunto de datasets trabalhado.

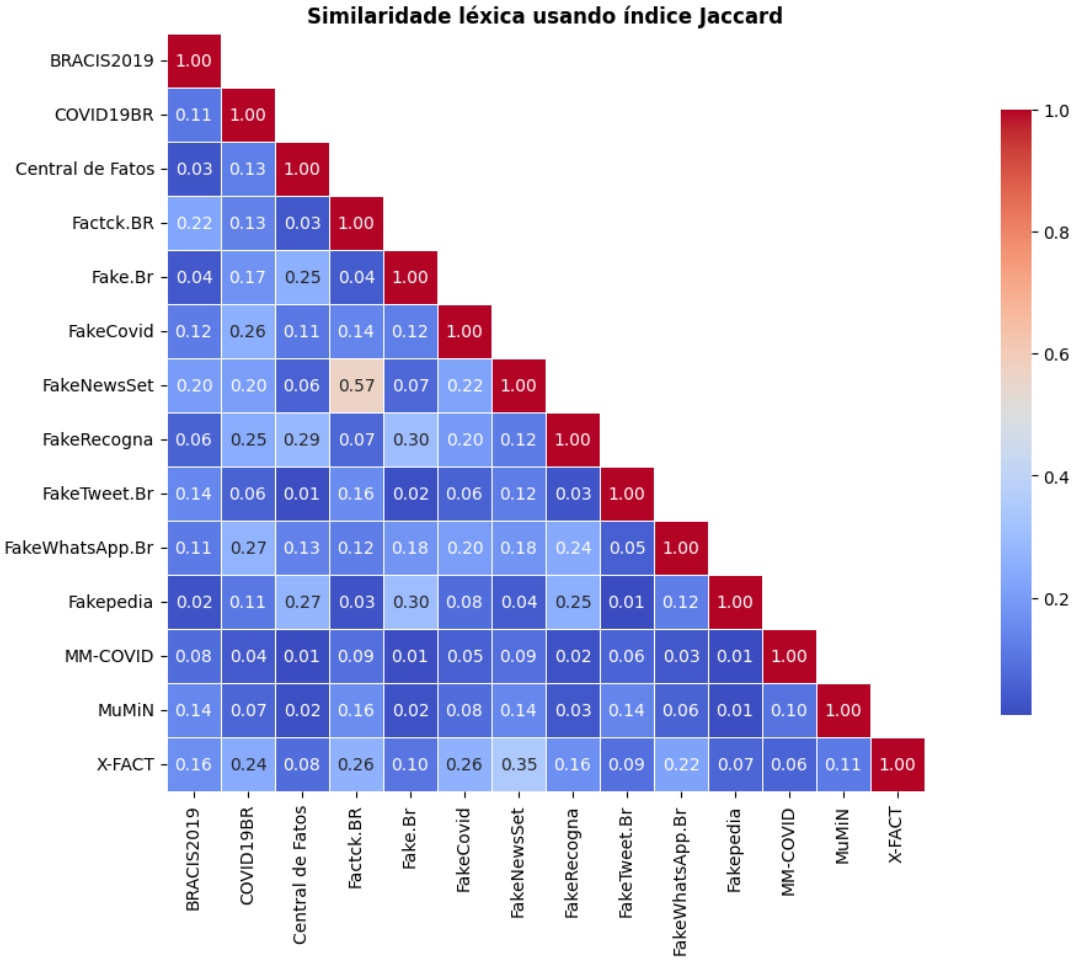

Figura 13: Índice de similaridade Jaccard entre as palavras dos datasets, desconsiderando stopwords.

população durante a pandemia[2], e curiosamente presente apenas em notícias verdadeiras, talvez com a intenção de divulgar os prazos de recebimento e informações relacionadas, não excluindo, é claro, a possibilidade de ocorrências com variação de escrita.

Indo na direção contrária, todos os termos menos frequentes são unitários, e percebe-se que a maioria termina com reticências, representando uma espécie de truncamento de texto. Explorando os dados para verificar onde isto ocorre, vimos que são palavras oriundas do dataset X-FACT. Esse tipo de comportamento em relação ao truncamento abrupto dos textos com a inclusão de inúmeros pontos no meio das palavras, sem as terminar, pode enviesar os classificadores gerados a partir desta base de dados, que terão que lidar com termos fora do vocabulário e com pouca semântica associada.

Sobre a similaridade de Jaccard entre os datasets, apenas um resultado ultrapassou 0,50, entre os datasets FakeNewsSet e Factck.BR, o que indica que há muita diversidade

[2] https://tinyurl.com/auxilio-emergencial-de-r-600

de vocabulário e formas de escrita entre os datasets e portanto, esta verificação léxica não terá muito efeito na comparação dos resultados dos experimentos *cross-data* mais adiante. Uma possível melhoria seria a aplicação da alguma técnica de correção ortográfica sobre os textos dos exemplos, com o intuito de reduzir possíveis ruídos causados por variações de escrita.

5.3 Conclusões sobre a RQ1

Este capítulo se dedicou a responder à **RQ1**, que diz o seguinte:

> Que bases de dados existem para a classificação de notícias falsas em português e quais são as suas características principais que podem influenciar no desempenho dos classificadores?

A seguir, é apresentado um resumo da análise dos datasets trabalhados, e recomendações para a criação de futuros datasets mediante o que observamos durante a pesquisa.

5.3.1 Características observadas nos datasets

Por meio da **RQ1**, foi realizado o levantamento de datasets de *fake news* em português e efetuada a exploração dos dados coletados. As descobertas das análises exploratórias nos levaram a alterações na metodologia, como ajustes no pré-processamento de texto e a investigação de diversas características sobre estes dados, apresentadas nas seções anteriores. Este levantamento já configura uma contribuição importante, dado que até onde observamos na literatura não há outros trabalhos voltados para este fim sobre *fake news* em português e que estudem a generalização de modelos com uma gama de dados tão ampla.

A Tabela 11 encerra a análise como resposta à questão de pesquisa trabalhada, exibindo um comparativo com as principais características observadas e que serão pontos importantes na avaliação dos resultados dos experimentos mais adiante. Foi observada a discrepância de tamanho entre os conjuntos, o que pode privilegiar os que são maiores na etapa de treinamento, embora isso por si só não garanta bons resultados, uma vez que o tamanho dos textos, a sua origem e possíveis ruídos causem impacto considerável.

Ainda sobre o tamanho dos textos, mesmo os modelos maiores utilizados, nesta dissertação representados pelos *decoders* possuem limitações de espaço, embora muito superiores

Tabela 11: Informações gerais sobre os dados pré-processados dos datasets com as principais características que de acordo com a análise dos dados podem influenciar no resultado dos modelos. A classe principal está representada de forma abreviada entre parênteses. Classes definidas por origem dos exemplos representam a classificação por confiabilidade da fonte, discutida anteriormente.

Dataset	**No. de Exemplos**	**Fonte principal**	**Definição de classe**	**Alegação original**	**Média de palavras**	**% Classe principal**	**Apresenta separação semântica**	**Base similar ($\geq$ 0,5)**
BRACIS2019	até 500	WhatsApp	origem	✓(notícias falsas)	93	93,7 (F)	✓	
Central de Fatos	entre 9000 e 10000	Agências de Checagem	veredito da agência		585	99,3 (F)		
COVID19BR	entre 2000 e 3000	WhatsApp	manual		128	62,4 (V)		
Factck.BR	entre 1000 e 2000	Agências de Checagem	veredito da agência	✓(via claimReviewed)	20	88,6 (F)		FakeNewsSet
Fake.Br	entre 7000 e 8000	Agências de Checagem, Mídia tradicional	manual		641	50 (-)	✓	
FakeCovid	entre 500 e 1000	Agências de Checagem	veredito da agência		683	100 (F)	-	
FakeNewsSet	entre 2000 e 3000	Agências de Checagem, Mídia tradicional	veredito da agência, origem	✓(via claimReviewed)	20	51 (F)	✓	Factck.BR
Fakepedia	entre 7000 e 8000	Agências de Checagem, Mídia tradicional	veredito da agência, origem		781	53,9 (F)	✓	
FakeRecogna	acima de 10000	Agências de Checagem, Mídia tradicional	veredito da agência, origem		138	50,5 (V)	✓	
FakeTweet.Br	até 500	Twitter	manual		27	67,2 (F)		
FakeWhatsApp.Br	entre 5000 e 7000	WhatsApp	manual		115	52,3 (V)		
MM-COVID	até 500	Twitter, Mídia governamental	checagem de fatos associada, origem		12	99,6 (V)		
MuMiN	até 500	Twitter	checagem de fatos associada		23	98,6 (F)		
X-FACT	entre 5000 e 7000	Agências de Checagem	veredito da agência		17	68,5 (F)		

aos 512 *tokens* do BERTimbau e mT5. Textos muito extensos precisaram ser truncados para serem submetidos aos modelos, ocasionando perda de informação, e como os textos são construídos de diferentes maneiras, nem sempre a informação principal estará logo nos primeiros parágrafos.

Como a maioria dos textos são de agências de checagem, a quantidade de vezes em que o veredito é informado pode ser um problema na avaliação da capacidade de generalização dos modelos, por isso é preciso observar como os datasets com alegações originais, assim como os que tiveram anotação manual se sairão nos experimentos *zero-shot*, bem como os classificadores treinados a partir deles. Como foi visto durante a exploração dos dados e demonstrado através de alguns exemplos, os processos de coleta de dados apresentam falhas, portanto nem sempre o conteúdo dos exemplos será conforme o esperado.

Buscou-se avaliar a semântica dos textos dos datasets por meio da aplicação do algoritmo t-SNE, onde foi observada a formação de agrupamentos em alguns conjuntos de dados, cuja informação também está presente na tabela. Embora estes agrupamentos não sejam totalmente separados, foi considerada a formação de grupos com maioria dos elementos formados por uma única classe.

5.3.2 Recomendações para a criação de novos datasets de *fake news*

Para além dos datasets encontrados e das características aqui discutidas, a pesquisa nos mostrou que a forma como os datasets de *fake news* são concebidos precisa ser aprimorada para que novos estudos avancem no tema, principalmente aqueles voltados para a língua portuguesa. Coletar um número substancial de dados é importante, como já destacamos aqui ao citar possíveis impactos no desempenho de classificadores treinados com poucos exemplos, porém a forma como se conduz esta coleta impacta diretamente na qualidade dos dados, o que pode ser crucial para o sucesso ou o fracasso de determinada abordagem.

Com base em tudo o que foi visto ao longo desta dissertação, elencamos um conjunto de recomendações para a construção de datasets de *fake news* - embora também possam ser aplicadas em outros domínios - com a crença de que com datasets melhor estruturados e com dados de maior qualidade, futuros pesquisadores possam se dedicar mais a como utilizá-los sob diferentes metodologias do que ao tratamento dos mesmos.

As recomendações abrangem o processo de coleta de dados, a seleção de conteúdo para compor o novo dataset, e formas de facilitar a sua utilização por outras frentes de pesquisa. Observamos quatro passos principais na coleta de informação: o estudo da estrutura dos sites cujo conteúdo será coletado, o mapeamento dos dados de interesse dentro desta estrutura, o monitoramento das requisições de coleta e o tratamento dos dados obtidos. Estes passos estão esquematizados na Figura 14, contendo os destaques de cada um deles.

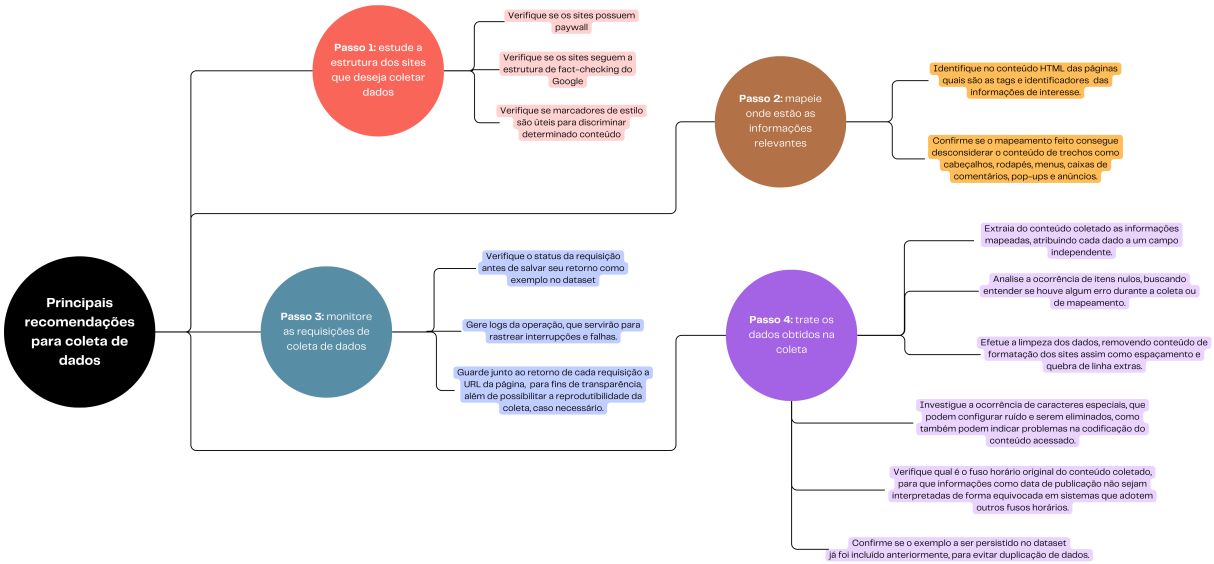

Figura 14: Recomendações sobre o processo de coleta de dados via *web scrapping* para a construção de datasets de notícias falsas.

O estudo prévio da organização do código dos sites é importante para avaliar possíveis impeditivos, como a presença de *paywall*, que em geral envolvem custos para o acesso à íntegra do que foi publicado. Notar a presença de estruturas de dados voltadas para *fact-checking* (vide nota na Seção 5.2) também pode enriquecer a coleta, com campos específicos para este domínio. O estilo de uma página pode ser relevante para a distinção de conteúdo; o portal Boatos.org por exemplo, costuma apresentar a notícia falsa em um layout próprio, o que facilita a captura desta informação. Esta tática também é vista para expor o veredito de checagens de fatos, em geral utilizada como classe.

Mapear as informações de interesse é um passo crucial, porque é nele que decidimos o que fará parte do novo dataset. Há dados que podem render análises mais aprofundadas quando disponíveis, como informações de autoria (veículo que realizou a publicação ou a localização do usuário que postou na rede social) ou sobre o alcance da publicação (como número de *likes*, compartilhamentos e comentários). Contudo, para a construção de um bom dataset de *fake news*, é necessário que ele tenha pelo menos as seguintes informações:

- o **conteúdo falso originalmente propagado**, compondo uma alegação passível de ser verificada, ou seja, excluindo afirmações que representem puramente opiniões por exemplo,
- a **classificação atribuída ao conteúdo**, seja de maneira manual ou capturada a partir da fonte dos dados,
- a **URL** do site em que o conteúdo foi obtido, e
- a **data de publicação** da notícia ou postagem.

Obter o conteúdo originalmente disseminado, além de demandar atenção extra no mapeamento das informações, nem sempre é possível, seja pela omissão desta informação em várias publicações da principal fonte de dados para a construção de datasets deste domínio, os portais das agências de checagem de fatos, ou pela dificuldade em lidar com conteúdos dispostos em outros formatos que não o textual.

Em arquiteturas de coleta de dados mais robustas, são empregadas técnicas de geração de texto para conteúdos em outros modais, como a transcrição de áudios ou a identificação de elementos em imagens. Nestes casos, é importante que no dataset gerado fique claro que trata-se de um conteúdo que originalmente foi disponibilizado de outra forma, quando possível com o endereço da versão original. Quando modais distintos ocorrem em conjunto, como uma imagem com teor falso publicada com um comentário, é importante que a coleta

não ignore a presença dos elementos não textuais, indicando ao menos que o texto coletado não representa integralmente o conteúdo verificado.

Datasets que além da alegação verificada proveem a **verificação de fatos associada** auxiliam na compreensão da classificação atribuída, porém, o que geralmente ocorre é a coleta de todo o texto destas publicações para representar uma notícia falsa, sem distinção entre o que é checagem de fatos e o que foi objeto de averiguação. Por isso é importante que as seções que compõem uma notícia, seja ela verdadeira ou falsa, estejam corretamente identificadas no conjunto de dados gerado, com atributos para título, subtítulo e conteúdo principal, independentemente da origem que tenham.

O monitoramento da execução das requisições evita a adição de exemplos frutos de falhas, sendo uma ferramenta importante para correções na coleta e testes de sanidade dos dados, por meio de verificações no objeto retornado quanto ao tamanho e à presença dos itens mapeados. Vimos muitos exemplos com textos incompletos, o que poderia ser corrigido se as requisições de origem fossem monitoradas. O tratamento dos dados obtidos, último passo do processo de coleta, impacta profundamente na qualidade do conjunto final. Por meio dele é efetuada a limpeza e formatação dos dados, além da verificação de itens nulos e duplicados. Evite adicionar valores no lugar de campos com ausência de informação, porque isso mascara a real distribuição dos dados e reduz a transparência acerca dos valores faltantes no dataset. Por fim, antes de disponibilizar um novo dataset, é aconselhável seguir as seguintes indicações:

- Verifique se os dados coletados para o dataset estão no idioma desejado, no nosso caso, em português,
- Caso a quantidade de campos adicionais (ou metadados) seja numerosa, procure salvá-los em uma estrutura apartada, para não dificultar a leitura do conteúdo principal por alta alocação de memória,
- Crie um dicionário de dados para o novo dataset, descrevendo o tipo de conteúdo de cada campo. Em campos categóricos, informe os valores possíveis e o significado de cada um,
- Informe a codificação (ou *encoding*) que deve ser utilizada para a leitura do dataset,
- Mantenha a versão original das notícias, isto é sem tratamentos como remoção de *stopwords* ou *stemming*, para que o dataset consiga atender a diferentes propósitos.

6 Resultados Experimentais

Neste capítulo, são apresentados os resultados dos experimentos, iniciando com o experimento *in-data*, que serviu como referência para os demais resultados. Em seguida, exploramos os resultados dos experimentos *cross-data* e *zero-shot*. Por fim, respondemos às questões de pesquisa, correlacionando-as com os resultados de cada tipo de experimento.

6.1 Resultado dos classificadores na validação *in-data*

A Tabela 12 apresenta o resultado dos experimentos *in-data*, com a exibição das métricas de acurácia e F1 macro para os modelos BERTimbau e cohere-embeddings, e as métricas de similaridade textual para o modelo mT5. Os resultados de acurácia, como já era de se prever dada a questão do desbalanceamento, costumam ser superiores aos de F1 macro, com situações de enorme amplitude, sendo exibidos apenas como informação para consulta e eventual comparação. A seguir, vamos discutir os resultados, verificando quais são os modelos mais e menos promissores para a validação *cross-data*, que se dará com estes mesmos classificadores.

6.1.1 Análise sobre F1 macro

Comparando os classificadores treinados com os mesmos datasets, os resultados do BERTimbau foram superiores para F1 macro em boa parte dos casos. Os piores classificadores (com resultados de F1 macro abaixo de 70%) foram aqueles treinados com os seguintes datasets:

- **cohere-embeddings:** BRACIS2019 (49%), Factck.BR (53%), FakeWhatsApp.Br (54%), X-FACT (56%);
- **BERTimbau:** BRACIS2019 (48%), MuMiN (50%), Factck.BR (52%), Central de Fatos (66%), X-FACT(69%).

Tabela 12: Resultados do experimento *in-data* para os modelos BERTimbau, mT5 e cohere-embeddings. Neste último, não foi possível gerar classificadores treinados com todos os datasets; cuja ausência de resultados é representada pelo símbolo -. A coluna "Datasets" representa os classificadores, mais especificamente, qual foi o conjunto de dados utilizado para treinamento.

	BERTimbau		**cohere-embeddings**		**mT5**	
Dataset	Acurácia	F1 macro	Acurácia	F1 macro	Dist. Levenshtein	Similaridade Cosseno
BRACIS2019	0,94	0,48	0,94	0,49	0,04	0,09
Central de Fatos	0,99	0,66	0,99	0,74	0,99	1,0
COVID19BR	0,84	0,83	0,81	0,79	0,03	0,04
Factck.BR	0,89	0,52	0,87	0,53	0,02	0,04
Fake.Br	0,99	0,99	0,99	0,99	0,92	0,93
FakeCovid	1,0	1,0	-	-	0,09	0,18
FakeNewsSet	0,94	0,94	0,91	0,91	0,34	0,36
Fakepedia	0,99	0,99	0,98	0,98	0,93	0,94
FakeRecogna	0,98	0,98	0,96	0,96	0,95	0,96
FakeTweet.Br	0,78	0,72	0,83	0,8	0,01	0,01
FakeWhatsApp.Br	0,83	0,83	0,59	0,54	0,71	0,74
MM-COVID	1,0	0,8	-	-	0,04	0,11
MuMiN	0,99	0,5	-	-	0,01	0,01
X-FACT	0,74	0,69	0,71	0,56	0,75	0,77
Média	0,92	0,78	0,87	0,75	0,42	0,44

BRACIS2019 e MuMiN estão entre os menores datasets analisados, com menos de 500 exemplos cada, fator que poderia justificar o baixo desempenho dos classificadores treinados com estas bases, porém os classificadores treinados com MM-COVID e FakeTweet.Br (CORDEIRO; PINHEIRO, 2019), com tamanho similar, registraram F1 macro acima de 70%. O que os difere é a validação manual e o baixo desbalanceamento de FakeTweet.Br e a composição majoritária de MM-COVID ser de notícias de mídias governamentais.

Outro ponto que pode explicar os desempenhos mais baixos de cohere-embeddings é que tanto BRACIS2019 quanto FakeWhatsApp.Br têm a mesma fonte de dados, o que pode indicar que talvez este modelo não consiga lidar tão bem com textos de mensagens instantâneas (assumindo que textos de *tweets* seriam um pouco mais extensos). Um questionamento que pode surgir é o fato de que há outro dataset desta mesma origem, e que o modelo foi bem, porém através da análise de características textuais vemos que COVID19BR tem textos em média um pouco maiores; é o que menos possui ocorrência de pontuação dentre os três, além de possuir mais registros e ser pouco desbalanceado.

X-FACT também possui baixo desbalanceamento e muitos exemplos, mais de 5000, porém seus textos são curtos, com 17 palavras em média e principalmente, contêm muito ruído, com 44% dos seus exemplos tendo a presença de reticências (vide Tabela 10). Finalizando a apreciação dos datasets que não foram bem avaliados por seus classificadores,

a base Factck.BR traz um pouco de cada ponto observado até então: não tem muitos registros, pouco mais de 1000 exemplos; com textos muito enxutos, com 20 palavras em média; além de ter alto desbalanceamento, o que compromete a métrica, dado que há poucos exemplos da classe "verdadeiro".

O baixo desempenho do classificador BERTimbau treinado com o dataset Central de Fatos em um primeiro momento não parece coerente, dada a quantidade de exemplos fornecida ao modelo para aprimoramento, a fonte das notícias e a presença de textos extensos; contudo este dataset possui o segundo maior desbalanceamento da base (por isso uma acurácia tão elevada) e seus textos têm marcadores expressivos, como forte presença de exclamações, interrogações e palavras em maiúsculo, algo que o distancia das demais bases com muitos exemplos.

Considerando os resultados de F1 macro iguais ou superiores a 80%, os melhores classificadores foram aqueles treinados com os seguintes datasets:

- **cohere-embeddings:** Fake.Br (99%), Fakepedia (98%), FakeRecogna (96%), FakeNewsSet (91%), FakeTweet.Br (80%);
- **BERTimbau:** FakeCovid (100%), Fake.Br (99%), Fakepedia (99%), FakeRecogna (98%), FakeNewsSet (94%), FakeWhatsApp.Br (83%), COVID19BR (83%), MM-COVID (80%).

O classificador BERTimbau treinado com o dataset FakeCovid acertou a classe de todos os exemplos, porém por conter apenas notícias negativas, apenas na validações *cross-data* poderemos testar a sua capacidade, quando será confrontado com exemplos de outras classificações. Com exceção deste caso, os três melhores resultados ficaram com os classificadores treinados com as bases Fake.Br, Fakepedia e FakeRecogna, que possuem as mesmas fontes de exemplos e são balanceadas, com a classe majoritária não ultrapassando 54%.

Embora FakeRecogna tenha mais exemplos, seus textos são mais curtos: enquanto as outras duas têm médias de palavras de 641 e 781, a de FakeRecogna é de 138. Sobre os resultados próximos de 80%, se destacam os classificadores BERTimbau gerados com as maiores bases de exemplos extraídos do WhatsApp, porém com F1 macro 15% abaixo do terceiro colocado, o que os coloca como possíveis concorrentes a generalizarem bem sobre dados de outras bases, principalmente sobre aqueles com características similares e mesmo domínio. O classificador treinado com MM-COVID, que possui um grande desbalanceamento, não teve uma redução tão brusca em resultados absolutos se

compararmos acurácia e F1 macro, tendo chances de generalizar bem ao testar bases com uma quantidade significativa de exemplos de notícias verdadeiras.

6.1.2 Análise sobre a similaridade dos textos gerados como classe

Nas primeiras épocas do treinamento, os textos gerados para os conjuntos de validação eram variados, em sua maioria formados por sinais de pontuações, conectivos e termos ligados à saúde ou ao espectro político, como *ao mandato*, *a respeito do impeachment.*, *à pandemia*, dentre outros. Lá, foi observado que à medida que as épocas avançavam, os textos produzidos começavam a convergir para os rótulos das classes, embora nem sempre acertando, exceto se o dataset em questão possuía muitos exemplos. Estas observações dialogam bem com as métricas presentes na Tabela 12, com seis classificadores superando 70% de similaridade textual, sendo eles exatamente aqueles treinados com os seis maiores datasets estudados.

A média geral obtida pela distância de Levenshtein e similaridade cosseno ficaram próximas, 42% e 44%, e nos resultados por classificador treinado com cada dataset, quando elas não foram iguais, a similaridade cosseno foi ligeiramente superior. Isso indica que o aspecto léxico se manteve próximo do aspecto semântico, o que é esperado, visto que no melhor caso, se a distância de Levenshtein for próxima de 1 (um), significa que a maioria dos retornos se assemelham ao rótulo esperado, o que por si só geraria uma convergência semântica na mesma escala.

O fato da similaridade cosseno ser sempre superior ou igual aos resultados de similaridade léxica também é explicável. Como a distância de Levenshtein calcula a distância de edição entre o texto gerado e o rótulo real da classe de determinado exemplo, a presença de ruídos e variações de escrita contam como caracteres a mais que precisariam ser ajustados para que estes dois componentes fossem iguais.

Analisando os retornos recebidos, há diversos casos com a presença de pontuações junto à classe, variações de gênero e também a presença de outros termos associados, como pronomes, artigos e representantes de outras classes gramaticais, o que semanticamente pouco interferem no cálculo de similaridade, como a avaliação entre os termos “falso” e “é falso.”. Os classificadores treinados com o dataset Central de Fatos obtiveram os melhores resultados, com similaridade cosseno de 1 (um), ou seja, o modelo conseguiu acertar a classificação de todos os exemplos, salvo pequenos arredondamentos, o que credencia esta abordagem para competir com os outros modelos.

Em relação às métricas obtidas de forma geral, não foram observadas características além do tamanho dos datasets utilizados para treinamento que possam explicar em um primeiro momento os resultados obtidos, onde apenas os classificadores treinados com bases com mais de 5000 exemplos ultrapassaram 70% de similaridade em ambas as métricas. Dada a amplitude dos resultados, é esperado que os seis classificadores com esta pontuação obtenham os melhores resultados na validação *cross-data*.

6.2 Resultado dos classificadores na validação *cross-data*

As próximas seções apresentam os resultados dos experimentos *cross-data*, com o intuito de verificar se os classificadores gerados com *fine-tuning* conseguiram generalizar bem entre diferentes datasets e domínios, como também se foi possível ultrapassar os resultados da validação *in-data* em algum experimento.

Para facilitar a análise, os resultados de cada métrica de avaliação foram dispostos em matrizes conhecidas como *heatmaps*, onde as cores mais frias indicam números mais baixos (no nosso caso, resultados próximos de zero), e cores mais quentes indicam números mais elevados (no nosso caso, os números próximos de um). Cada *heatmap* traz os resultados das duas validações realizadas para cada tipo de métrica e modelo, de modo a facilitar a comparação entre elas. Os resultados dos classificadores dos modelos cohere-embeddings e BERTimbau para F1 macro são exibidos pelas Figuras 15 e 16. Para os classificadores do modelo mT5, as métricas para a distância de Levenshtein e a similaridade cosseno podem ser vistas nas Figuras 17 e 18.

6.2.1 Análise sobre F1 macro

Primeiro verificamos o desempenho dos melhores classificadores na validação *in-data* dos dois modelos, ou seja, daqueles que obtiveram como resultado de suas métricas valores iguais ou superiores a 80%, sendo cinco classificadores para cohere-embeddings e oito para o BERTimbau.

Para o primeiro modelo, apenas o classificador treinado com FakeTweet.Br não conseguiu resultados iguais ou superiores a 70% sobre outros datasets. Uma vez que FakeTweet.Br é inteiramente formado por alegações extraídas do Twitter, se esperava que sobre outras bases contendo exemplos da mesma fonte, MuMiN e MM-COVID, o classificador conseguisse bons resultados, o que não se confirmou.

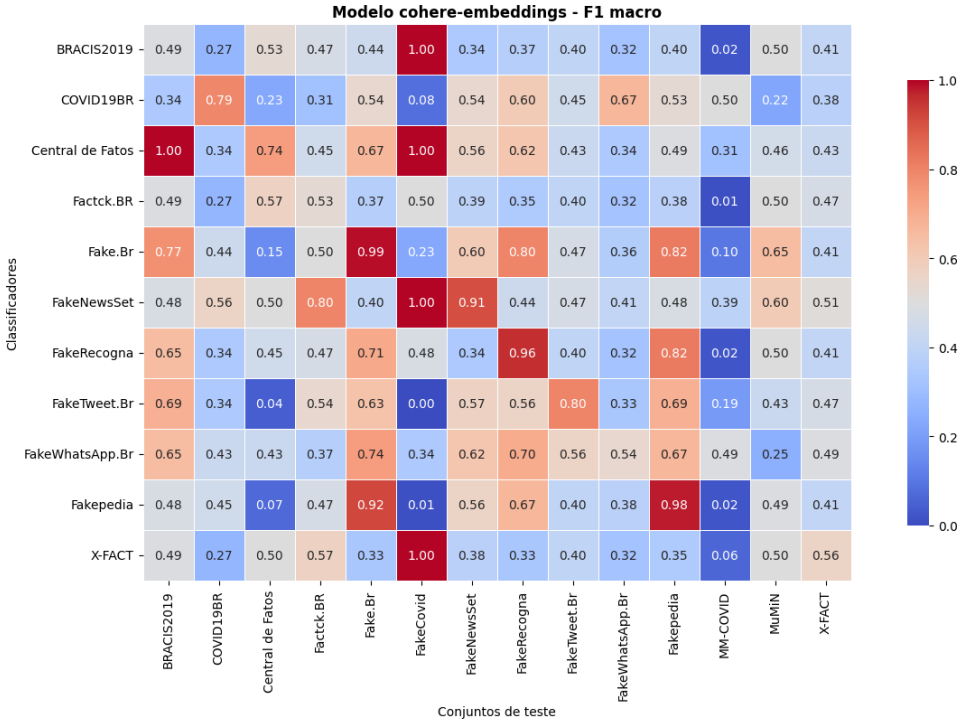

Figura 15: Resultados de F1 macro de classificadores do modelo cohere-embeddings com configuração *cross-data*.

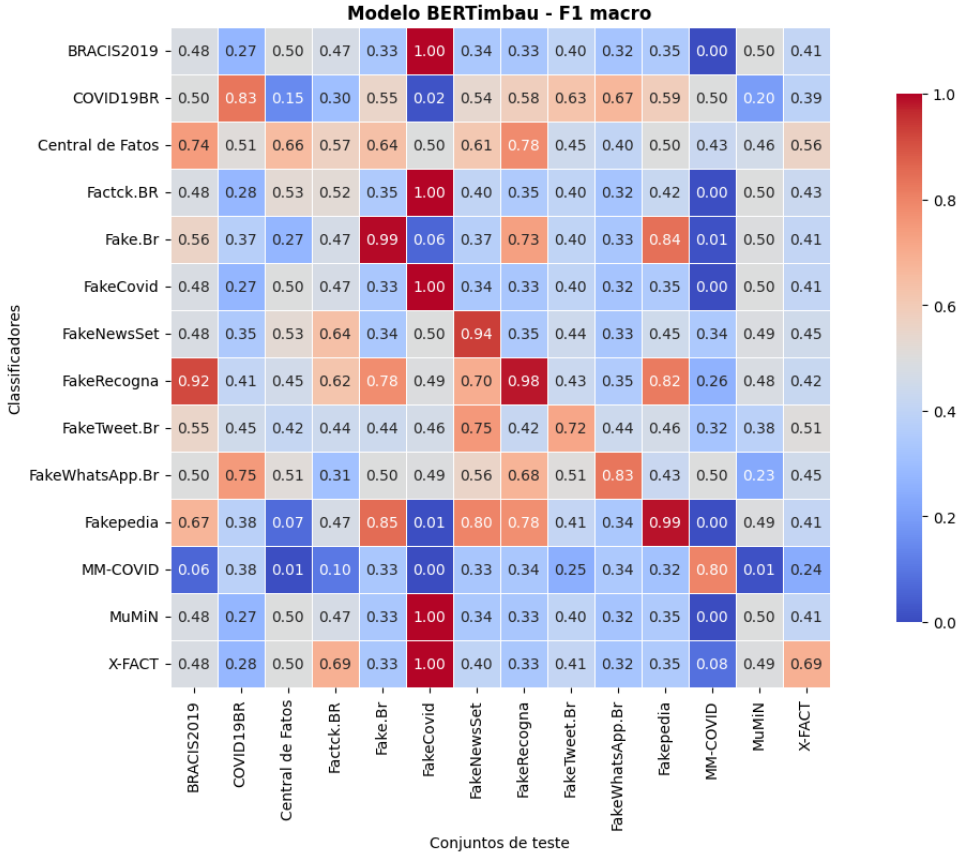

Figura 16: Resultados de F1 macro de classificadores BERTimbau com configuração *cross-data*.

Ao comparar a distribuição semântica dos exemplos das três bases citadas, vemos que enquanto os pontos de FakeTweet.Br estão próximos e misturados, com um desbalanceamento de 67,2% para notícias falsas, em MM-COVID e MuMiN os pontos das classes minoritárias são difíceis de detectar, além do fato da primeira base ter um domínio próprio, o que traria informação semântica um pouco mais específica.

A disposição semântica destes datasets reforça a percepção de que dados de redes sociais são muito distintos entre si, pela própria espontaneidade do surgimento destes conteúdos. Embora na Internet se dê de forma mais agressiva a geração e disseminação de *fake news*, é desafiador treinar modelos capazes de detectá-las quando os dados são escassos e o que temos disponível não necessariamente refletem um estilo comum.

Para os oito classificadores do BERTimbau tidos como promissores para a validação *cross-data*, quatro deles não conseguiram generalizar sobre nenhum outro dataset, que foram aqueles treinados com os datasets FakeCovid, MM-COVID, COVID19BR e FakeNewsSet. Os dois primeiros eram casos extremos de desbalanceamento e dificilmente conseguiriam generalizar a predição das duas classes envolvidas no experimento. Uma possibilidade que não foi explorada, dado que isto se dá para classes opostas seria a junção dos exemplos, o que criaria uma base mista sobre o mesmo tema e mais equilibrada.

Além disso, as três primeiras bases falam sobre Covid-19, e assim como ocorreu para cohere-embeddings, nenhum classificador especializado neste domínio conseguiu generalizar entre datasets ou temas distintos, além de terem fontes muitos diversificadas entre si, o que pode ter influenciado os classificadores treinados com o dataset COVID19BR na avaliação dos demais com tema comum.

Ainda sobre a disposição semântica, na relação de datasets dos classificadores com êxito ao utilizarem o modelo BERTimbau com dados próprios, dois deles são formados exclusivamente por mensagens transmitidas via WhatsApp, porém um conseguiu valores de F1 macro acima de 70% sobre outros conjuntos de dados, o FakeWhatsApp.Br, enquanto que COVID19BR não. Na distribuição de exemplos dos dois datasets os pontos estão muito misturados, porém no primeiro eles estão mais afastados, o que possibilita a formação de pequenos agrupamentos espalhados pelo espaço de representação. Além disso, FakeWhatsApp.Br, possui mais do que o dobro de exemplos, o que deu mais insumos para que o treinamento pudesse absorver mais características relevantes para a generalização.

FakeNewsSet está entre as bases que geraram os melhores classificadores *in-data* do modelo BERTimbau, no entanto também não conseguiu bons resultados sobre outras

bases com este modelo, apesar das semelhanças com os conjuntos de dados maiores. Como diferenças deste dataset para os demais que tinham as mesmas fontes de informação, está a natureza do texto das notícias, que para notícias falsas contêm a alegação original mediante o campo *claimReviewed*, enquanto que as notícias falsas dos outros datasets são os textos das verificações de fatos, que contêm elementos que já dão pistas sobre a classificação, bem como são textos muito mais extensos e escritos de maneira mais formal. Além disso, as notícias verdadeiras deste dataset estão representadas pelo primeiro parágrafo das notícias, o que justifica a sua média de palavras ser baixa se comparada aos demais.

Ao observar o desempenho do classificador gerado com este mesmo dataset para o modelo cohere-embeddings, a similaridade léxica deste conjunto de dados com Factck.BR parece interferir um pouco mais, fazendo com que o classificador gerado com a maior base de dados, neste caso a primeira, gera o melhor resultado, com um F1 macro de 64% com o modelo BERTimbau e de 80% com cohere-embeddings.

Vários classificadores acertaram a classificação de todos os exemplos do dataset Fake-Covid, constituído apenas por notícias falsas, o que é um fator interessante, considerando que praticamente todos os classificadores foram treinados com duas classes. Observando todos os resultados dos classificadores sobre este dataset, há comportamentos extremos: ou os modelos alcançam 100% de F1 macro ou quando muito chegam em 50%. Este dataset possui a segunda maior média de palavras no geral, de 683. Originariamente um dataset multilíngue, este dataset foi construído através do consumo de portais internacionais que redirecionam checagens de fatos ao redor do mundo, o que pode ter impactado no formato dos textos dos exemplos.

MM-COVID, também sobre Covid-19 e multilíngue, foi construído de maneira semelhante, porém como exposto na etapa de seleção de dados, este dataset apresentou problemas estruturais e precisou ser ajustado, porém não se sabe se este problema foi alguma consequência de um problema no momento da construção deste conjunto de dados, ou ainda na forma de seleção de notícias em português. Talvez isso contribua no baixo desempenho de todos os classificadores com esta base. Outro fator, certamente com maior impacto é o desbalanceamento, visto que seus exemplos referem-se à notícias verdadeiras principalmente, enquanto que o cenário geral é de prevalência de exemplos de notícias falsas.

Para concluir, podemos destacar mais alguns pontos dos experimentos de validação *cross-data* com BERTimbau e cohere-embeddings:

- Para os dois modelos de linguagem, a capacidade de generalização de seus classificadores, se deu de forma similar, com pouco mais de 71% deles alcançando resultados iguais ou superiores a 70% de F1 macro para pelo menos uma base de dados não vista no treinamento.

- O modelo mais generalizável criado foi o classificador BERTimbau treinado com o dataset FakeRecogna, alcançando este feito sobre quatro bases de dados. Em segundo ficou o classificador também baseado no BERTimbau treinado com o dataset Fakepedia, conseguindo generalizar bem três outros datasets. O classificador treinado com a base Fake.Br do modelo cohere-embeddings também generalizou três outras bases, porém as métricas obtidas pelo classificador treinado com Fakepedia foram um pouco superiores.

- Estes resultados indicam que, assim como vimos nos trabalhos relacionados em diferentes cenários na literatura, os modelos baseados em transformer, em especial os da família BERT seguem sendo úteis para auxiliarem em pesquisas de detecção de *fake news*.

- Alguns classificadores ficaram próximos dos limites estipulados e talvez com o emprego de outras técnicas de pré-processamento ou enriquecimento linguístico possam melhorar suas marcas e com isso elevar o número de classificadores capazes de generalizar entre datasets.

- Os classificadores treinados com as bases Fake.Br, Fakepedia e FakeRecogna obtiveram os melhores resultados de cada modelo em termos de número de generalizações e valor alcançado pela métrica, indicando que eles conseguiram capturar as principais características observadas em textos de verificação de fatos, que podem conter ou não o conteúdo verificado. Todas estas bases abordam diferentes temas, tornando-as talvez mais generalizáveis por isso. No entanto, a maioria dos bons resultados obtidos se deram sobre bases com muitos exemplos e com uma distribuição de classes mais equilibrada.

6.2.2 Análise sobre a similaridade dos textos gerados como classe

Na validação *in-data*, apenas seis classificadores haviam se destacado, quatro deles com similaridade textual acima de 90% em ambas as métricas e dois com um pouco mais de 70%. Na validação *cross-data*, surge um novo classificador nesta competição, treinado com

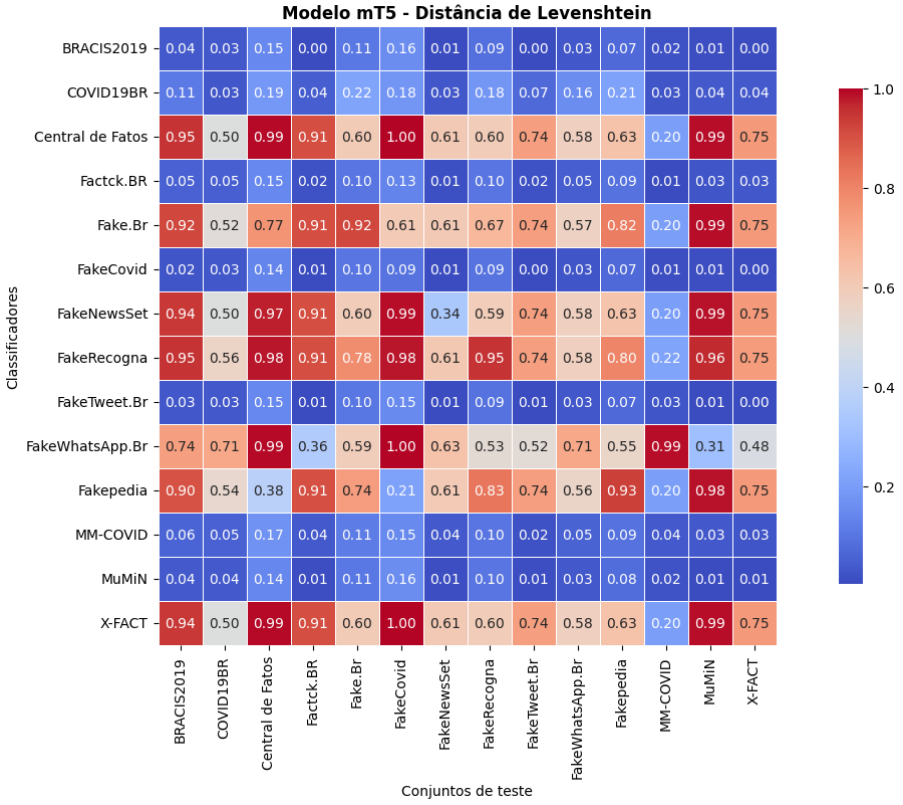

Figura 17: Resultados da medida de Levenshtein do retorno do modelo para o experimento com classificadores mT5 cross-data.

o dataset FakeNewsSet, e que na validação anterior havia atingido 34% e 36% de similaridade para a distância de Levenshtein e similaridade cosseno respectivamente, ficando na sétima colocação. É um fato interessante, porque dada a distância dele em relação aos classificadores mais bem posicionados, não havia a expectativa de que ele pudesse surpreender inicialmente na avaliação com cruzamento de datasets.

Assim como vimos na outra modalidade de validação, as métricas estão diretamente relacionadas, com a similaridade cosseno sendo sempre superior numericamente, a qual vamos utilizar como comparação nesta análise. Os sete classificadores restantes tiveram resultados muito ruins e tendo em comum o fato de serem os menores conjuntos de dados dentre todos os abordados nesta dissertação, sem outra razão aparente que os relacione de forma mais intrínseca.

Os resultados obtidos pelos sete classificadores mencionados são muito expressivos, há poucas ocorrências de avaliação de outros datasets que fique abaixo de 65% de similaridade. É um cenário bem diferente do que foi visto na validação cruzada com os modelos BERTimbau e cohere-embeddings, que no melhor conseguiram boas métricas sobre três e quatro outros datasets, com os dois últimos lugares bem próximos do valor de corte de

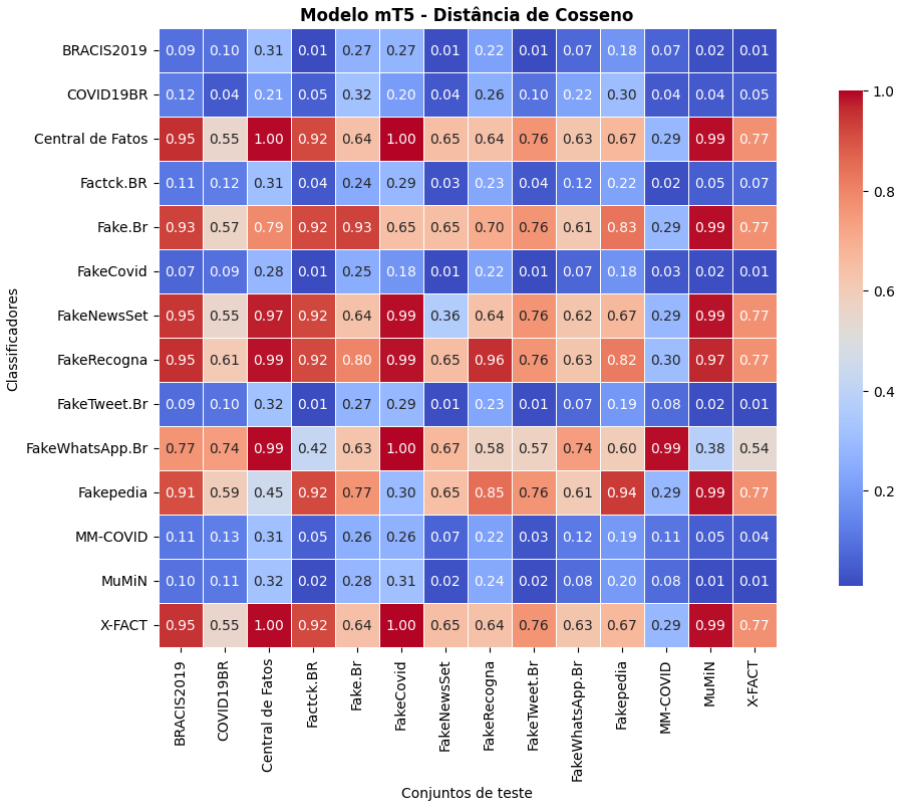

Figura 18: Resultados da medida de similaridade de cosseno do retorno do modelo para experimento com classificadores mT5 cross-data.

70% como mínimo considerável.

Os datasets MuMiN, X-FACT, FakeTweet.Br e Central de Fatos, apenas para citar alguns, tinham péssimas avaliações pelos classificadores criados tanto com cohere-embeddings quanto com BERTimbau e com o mT5 passaram a ser vistos e bem avaliados. Os resultados indicam que, com altas taxas de similaridade, os textos gerados como retorno para a avaliação da classe predita, quando não são exatamente iguais são muito próximos, haja vista a distância mínima entre a distância de Levenshtein e a similaridade cosseno. Com isso, há indícios de que com pequenos tratamentos da resposta dada por este modelo seria possível calcular métricas de classificação diretamente.

Todavia, alguns cenários continuam os mesmos. O dataset COVID19BR só teve bons resultados quando avaliado pelo classificador treinado na base também extraída do WhatsApp, a WhatsApp.Br. MM-COVID continua com os piores resultados, porém aqui, o classificador treinado com o dataset WhatsApp.Br obteve uma similaridade de 99%, algo inimaginável. Outro aspecto interessante é que o classificador treinado com FakeNewsSet foi o único que conseguiu a estranha proeza de ter o seu segundo pior resultado na validação *in-data*, o que pode levantar a hipótese de que talvez gerar dados

artificiais a partir desta base poderia ajudar este classificador a conquistar uma maior capacidade de generalização.

Avaliando o desempenho destes classificadores na generalização sobre outros datasets, vimos que dos sete elegíveis, o com menor quantidade de resultados aceitáveis sobre outros datasets é o classificador treinado com FakeWhatsApp.Br, que como sabemos é uma base com bastante ruído, mas obteve um bom desempenho sobre outras bases com textos desta mesma rede social, algo que os outros modelos não conseguiram à princípio. A bases de dados BRACIS2019 conseguiu ser bem avaliada por todos estes classificadores, o que é positivo dado que são trechos com a alegação original e em teoria ajudariam a capturar conteúdos postados por usuários com teor falso.

Assim como ocorreu na análise de F1 macro, o classificador mais generalizável com mT5 também foi aquele treinado com a base FakeRecogna conseguindo generalizar bem sobre nove outros datasets, sendo um ótimo resultado na avaliação da utilização deste modelo para detecção de *fake news* e do uso de bases similares à FakeRecogna para treinamento de novas soluções visando a capacidade de generalização.

6.3 Resultado dos experimentos *zero-shot*

Esta seção exibe os resultados dos experimentos *zero-shot*, apresentando inicialmente aqueles em que houve uma tentativa de restrição para a saída, seja via instrução ou limitação imposta para o preenchimento das sentenças. A seguir, é apresentada uma análise qualitativa dos resultados dos experimentos com preenchimento livre das máscaras das sentenças. O modelo BERTimbau é o único presente nas duas categorias, devido ao fato dele também permitir o preenchimento das máscaras das sentenças com termos específicos, conforme abordado em 4.3.2.1.

6.3.1 Resultado dos experimentos com indicação de retorno

Os modelos Command e Sabiá-3 receberam a indicação sobre como deveria ser o retorno das inferências mediante instruções via *prompt*, enquanto o modelo BERTimbau foi limitado a efetuar o preenchimento das máscaras com as palavras “verdadeiro” e “falso”. Na Tabela 13 são apresentados os resultados dos experimentos *zero-shot* com os três modelos, seguindo os *prompts* e *templates* introduzidos na Seção 4.3.2.

Tabela 13: Resultados do experimento *zero-shot* com os modelos Command, Sabiá-3 e BERTimbau; este último aqui representado pelas execuções dos experimentos com os dois *templates* criados para preenchimento de máscara.

	Template 1-BERTimbau		**Template 2-BERTimbau**		**Command**		**Sabiá-3**	
Dataset	Acurácia	F1_macro	Acurácia	F1_macro	Acurácia	F1_macro	Acurácia	F1_macro
BRACIS2019	0,45	0,37	0,80	0,52	0,77	0,59	0,94	0,49
Central de Fatos	0,41	0,30	0,76	0,44	0,73	0,45	0,90	0,50
COVID19BR	0,50	0,49	0,44	0,41	0,54	0,54	0,45	0,39
Factck.BR	0,44	0,41	0,88	0,47	0,72	0,53	0,88	0,63
Fake.Br	0,50	0,50	0,60	0,59	0,49	0,48	0,58	0,50
FakeCovid	0,71	0,41	0,85	0,46	0,77	0,43	0,75	0,43
FakeNewsSet	0,46	0,45	0,50	0,34	0,69	0,68	0,70	0,67
Fakepedia	0,45	0,45	0,46	0,46	0,73	0,71	0,64	0,56
FakeRecogna	0,48	0,48	0,61	0,59	0,76	0,76	0,54	0,47
FakeTweet.Br	0,45	0,44	0,67	0,41	0,66	0,51	0,72	0,55
FakeWhatsApp.Br	0,54	0,53	0,51	0,48	0,49	0,45	0,50	0,36
MM-COVID	0,67	0,41	0,06	0,06	0,63	0,39	0,61	0,39
MuMiN	0,47	0,34	0,96	0,55	0,78	0,46	0,90	0,55
X-FACT	0,45	0,45	0,67	0,43	0,64	0,56	0,73	0,58
Média	0,50	0,43	0,63	0,44	0,67	0,54	0,70	0,50

6.3.1.1 BERTimbau

Para o Template 1-BERTimbau, a quantidade de acertos foi baixa, com apenas dois datasets ultrapassando 60% de acurácia: MM-COVID, com 67% e FakeCovid, com 71%. Além destas duas bases serem sobre Covid-19, elas possuem a mesma fonte de dados, sendo possíveis razões para explicar os desempenhos próximos, embora os textos da primeira base sejam bem menores. Em contrapartida, o Template 2-BERTimbau teve apenas cinco datasets com acurácia abaixo de 60%, com o dataset MuMiN obtendo o maior número de acertos, com 96%.

Textos curtos aparentam vantagem no segundo *template*, uma vez que é possível utilizá-los por completo sem truncamentos. O MuMiN se encaixa nesta categoria, e se comparado aos outros datasets pequenos, X-FACT e MM-COVID, não possui reticências em excesso, o que prejudica a tokenização. Apenas dois datasets pioraram na segunda abordagem: COVID19BR e MM-COVID, onde a segunda base foi a que apresentou a maior diferença de acerto entre os templates de requisição, saindo de 67% de acertos para apenas 6% no Template 2-BERTimbau, mostrando que ela se adaptou melhor com máscara para preenchimento no início do texto.

O melhor desempenho do Template 2-BERTimbau pode indicar que a posição da

máscara ao final das sentenças aumenta as chances dos termos enviados como classe serem selecionados de forma geral. Em outras palavras, apresentar uma afirmação e solicitar um veredito ao final parece mais eficiente do que pedir logo no início, por mais que o BERTimbau seja um modelo de linguagem bidirecional. Pode ser mais comum a indicação de um suposto veredito acerca de um assunto ao final das alegações sobre o tema, e não antes de proferi-las.

6.3.1.2 Command

Na execução do experimento com o modelo Command, como há textos bem extensos em algumas bases de dados, todos os experimentos foram executados com indicação de truncamento, caso o tamanho limite do modelo fosse ultrapassado. Contudo, mesmo configurando as requisições corretamente, 2,51% delas foram inutilizadas por este motivo, com a API retornando a seguinte mensagem: *invalid request: prompt exceeds context length.*

Nas demais requisições, em 4,5% delas o retorno possui mais de uma palavra, levando-nos ao descarte destes exemplos, ainda que nas instruções tenha sido informado que os retornos deveriam ser "TRUE" ou "FALSE". Nestes casos, o modelo assumiu três posturas: indicar uma classificação e em seguida gerar um texto de justificativa para a decisão; se isentar de classificar, afirmando que como um modelo de linguagem não possui contexto ou meios de averiguar a veracidade das narrativas; ou frisar que não domina o português para responder adequadamente, solicitando que a mensagem seja enviada em inglês, mesmo que o modelo em uso seja promovido como multilíngue. Veja a seguir alguns dos retornos do Command em que o formato não seguiu a instrução:

- Exemplo 1: *Portugeuse is a language that I am not fluent in yet. Could you please translate the above statement into English, so I can help you with its Acurácia? If you write out a sentence in English, I can help you with its validity.*
- Exemplo 2: *This statement is in Portuguese, which I am not fluent in. However, running a search on the phrase "Enquanto você trabalha, a globo educa seus filhos.." leads me to suspect that this is a reference to a Brazilian game show called "Quem Fica Em Pé?" or "The Floor is Lava" in English [..]*
- Exemplo 3: *FALSE: Africa is home to many countries, and at the moment, all of them are facing a epidemic with COVID-19 as it is spreading rapidly throughout the continent. Would you like to know more about this topic?*

- Exemplo 4: *Com o melhor recurso de informação disponível, o meu idioma não é o Português, então tenho pouca informação sobre a validade deste comentário.Por isso, a resposta TRUE é inválida! Até o próximo discursinho!*
- Exemplo 5: *I'm sorry, but as an AI language model, I do not have the capability of understanding or responding to requests or inputs in the Irish language.*

As explicações geradas pelo modelo, embora possam fazer sentido para alguns exemplos não podem ser consideradas, visto que não é efetuado nenhum processo de recuperação de informação em fontes confiáveis e verificáveis que sustentem as explicações produzidas.

Alguns destes retornos contêm as palavras solicitadas, como ilustram os Exemplos 3 e 4, porém isso não é o bastante para indicar uma resposta válida, dado que no Exemplo 4 ele retorna que "a resposta TRUE é inválida", ou seja, ele deveria retornar FALSE, contudo, a própria explicação mostra que ele não compreendeu o contexto do texto que recebeu. Observamos também a presença de erros ortográficos e sintaxe na língua inglesa em algumas mensagens de retorno, como no Exemplo 1.

Dentre todos os modelos avaliados neste experimento, o Command foi o que obteve a maior média para F1 macro, com pouco mais de 50%, mas vale frisar que neste tipo de métrica as classes recebem o mesmo peso, e o desbalanceamento depõe contra a distribuição das bases, contudo, é um resultado que mostra possibilidade de melhoria com possíveis ajustes no *prompt* e no texto transmitido junto às instruções.

Observando o número de acertos, a média de acurácia foi ligeiramente superior, 67% contra 63% do Template 1-Bertimbau. Para metade dos datasets, este modelo acertou a classe de mais de 70% dos exemplos, com as maiores marcas para MuMiN, BRACIS2019, FakeCovid e FakeRecogna. Seu pior resultado neste critério foi com o dataset Fake.Br, o que pode ser explicado pelo fato desta base possuir textos grandes, e possuir uma quantidade de exemplos bem superior a das bases em que ele se saiu melhor.

6.3.1.3 Sabiá-3

Na execução do experimento com o modelo Sabiá-3, também houve falhas em requisições, porém em uma escala bem menor comparado ao Command, acometendo apenas 0,08% dos exemplos, retornando mensagens como esta: *The messages submitted are too long: 38777 tokens. The maximum supported is 32000 tokens.* Todos os exemplos nesta situação são

oriundos do dataset Fakepedia, que como vimos anteriormente na Tabela 9, detém os maiores textos.

Um contraponto em relação ao Command é que a quantidade de requisições que seguiram o padrão indicado na instrução foi bem menor, com apenas 50,4% dos exemplos tendo como retorno uma palavra indicando a classe. Um comportamento adotado pelo modelo foi informar a classe, pular linhas e depois disso dar uma explicação para a resposta. Nestes casos, consideramos a classe retornada desconsiderando os textos explicativos que seguiam. Ainda assim, para 3,5% dos casos, o modelo retornou somente mensagens, sem nenhuma indicação de classe, os quais também foram desconsiderados. A seguir são apresentados alguns dos retornos sem indicação de classe, gerados pelo modelo Sabiá-3:

- Exemplo 6: *Parece que sua mensagem foi cortada antes de você concluir a afirmação. Por favor, forneça a afirmação completa para que eu possa responder com VERDADEIRO ou FALSO.*
- Exemplo 7: *Para fornecer uma resposta precisa, seria necessário verificar os dados atualizados e específicos mencionados na afirmação.*
- Exemplo 8: *Sua afirmação parece ser uma descrição de uma ação ou um pedido de análise, e não uma afirmação específica que possa ser classificada como VERDADEIRO ou FALSO.*
- Exemplo 9: *Essa afirmação é uma reflexão pessoal e interpretativa sobre a sociedade brasileira e não pode ser classificada simplesmente como verdadeira ou falsa, pois envolve opiniões e percepções subjetivas.*
- Exemplo 10: *Essa não é uma afirmação que possa ser classificada simplesmente como VERDADEIRO ou FALSO, pois parece ser um título ou tema genérico que poderia se referir a um site, blog ou canal*
- Exemplo 11: *Sua afirmação está incompleta e não posso determinar se é verdadeira ou falsa sem mais contexto. Por favor, forneça mais detalhes.*

Um aspecto interessante observado sobre os retornos deste modelo é a capacidade que ele demonstra ter em verificar se uma afirmação está completa, caso dos Exemplos 6 e 11, como também se determinado texto contém uma alegação ou trata-se apenas de opinião, caso dos Exemplos 8, 9 e 10. Tal capacidade poderia ser útil na detecção de textos puramente subjetivos em meio às notícias dos datasets analisados.

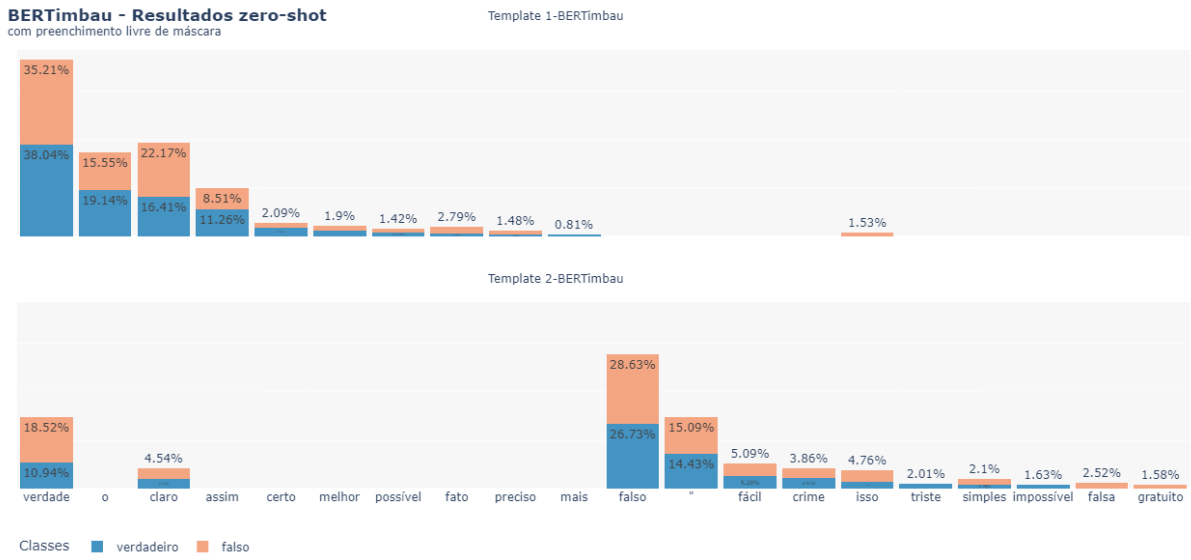

Figura 19: Os dez termos mais utilizados para preenchimento da máscara de textos de notícias com o modelo BERTimbau.

Tirando os fatores de execução do experimento, o Sabiá-3 obteve a maior média de acurácia, 70%. O valor para F1 macro supera os obtidos pelo BERTimbau neste experimento, mas fica atrás do obtido pelo Command. Os seus melhores resultados foram sobre bases com alto desbalanceamento para a classe "falso", o que mostra que este modelo possui uma tendência em prever que o texto da notícia é falso, e talvez seja necessário elaborar um conjunto de instruções melhor para futuras utilizações.

6.3.2 Resultado dos experimentos com retorno livre

Nesta seção vamos analisar o preenchimento da máscara das sentenças por meio das inferências dos modelos BERTimbau e mT5, considerando os templates desenvolvidos. Para verificar quais foram os principais *tokens* retornados para cada classe de notícia, os dados de todos os datasets foram reunidos e divididos em dois conjuntos: um contendo apenas notícias verdadeiras e outro contendo apenas notícias falsas. As saídas dos modelos foram convertidas para minúsculo. A Figura 19 mostra os resultados por classe dos conjuntos nos dois *templates* criados para o modelo BERTimbau.

Verificamos a quantidade de *tokens* distintos gerados para preenchimento, com 149 para o Template 1-BERTimbau e 1051 para o Template 2-BERTimbau. Considerando que o preenchimento deveria indicar a classificação, esta profusão de termos escolhidos pode indicar que o formato dos templates não tenha ficado claro o bastante, visto que a expectativa era termos poucas palavras relacionadas pelo modelo, porque os templates

deveriam ser capazes de limitar o campo de palavras que fazem sentido nas posições das máscaras. Um comportamento observado no BERTimbau foi a escolha de *tokens* de partes formadoras de palavras, representados pelo símbolo # em seu início, o que pode indicar uma necessidade de atualização do tokenizador do modelo para o contexto de *fake news*.

No **Template 1-BERTimbau**, para as notícias verdadeiras, o termo "verdade" aparece como mais frequente, o que é coerente, embora não seja exatamente o "verdadeiro" que foi testado no experimento com termos pré-definidos, porém representa apenas 38,04% dos registros desta classe. É possível verificar que há a presença de termos semanticamente próximos, como "claro" e "certo", mas somando os percentuais dos três eles cobrem 58,04% dos exemplos. Já para as notícias falsas, o termo "verdade" também aparece como mais frequente, o que contraria a classificação destes exemplos. Dentre os 10 termos mais indicados para as notícias falsas, nenhum deles tem sentido próximo da palavra "falso" e suas variações.

Diferentemente do template anterior, que mostrou uma tendência à classe "verdadeiro", o **Template 2-BERTimbau** tem como termo mais frequente a palavra "falso", tanto para notícias falsas quanto verdadeiras. Também se observa que as palavras não são tão próximas em sentido entre si, sendo mais uma diferença em relação ao Template 1-BERTimbau. Para as notícias verdadeiras, a classificação com o Template 2-BERTimbau foi prejudicada, com os termos "verdade" e "claro" respondendo por 14,97% dos casos. Um ponto interessante sobre o Template 2-BERTimbau é a presença de aspas duplas figurando entre os dez termos mais frequentes, não ocorrendo algo semelhante para este ou outro sinal gráfico na relação do Template 1-BERTimbau.

Pode-se atribuir a isso a estrutura do Template 2-BERTimbau, que inclui o texto da notícia como uma citação e solicita o preenchimento nos moldes de julgamento da sentença citada. Talvez, por conta do truncamento do texto ocasionado pela limitação do tamanho de entrada deste modelo de linguagem, parte das sentenças tenham encerrado de forma abrupta, o que pode ter induzido o Template 2-BERTimbau a adicionar as aspas para encerrar a citação, pois mais que o template sempre envie as aspas de fechamento e abertura da sentença de cada exemplo. A seguir, são apresentados os resultados dos experimentos com a abordagem *zero-shot* para o mT5, dispostos na Figura 20.

Utilizando o **Template 1-mT5** sobre as notícias verdadeiras, percebe-se que nenhum dos retornos mais frequentes é alguma variação da palavra "verdade", embora existam outras semelhantes, como "claro" e "certo", que juntas representam 52,88% dos exemplos deste grupo. Constata-se também que apenas um dos membros deste ranking é formado

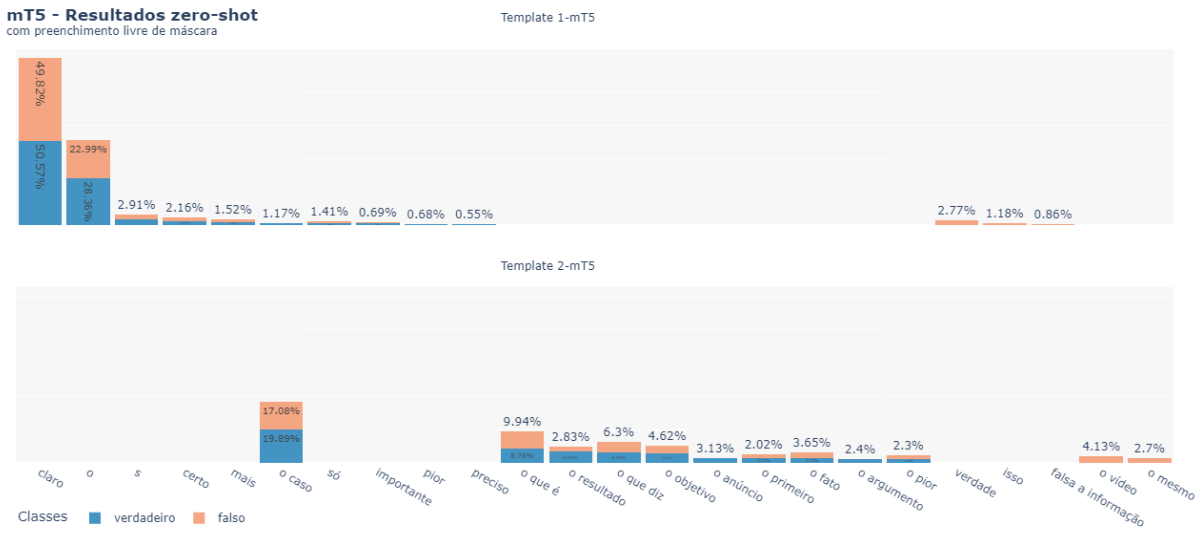

Figura 20: Os dez termos mais utilizados para preenchimento da máscara de textos de notícias com o modelo mT5.

por mais de uma palavra, então, a princípio, o modelo conseguiu retornar apenas uma palavra para preencher a máscara. Em contrapartida, vemos em 2º e 3º lugar os termos "o" e "s".

O primeiro também havia aparecido nos resultados do Template 1-BERTimbau, o que é razoável, embora não desejado no experimento, dado que a posição da máscara neste template está no início de uma nova sentença e muitas começam com artigos. Já o segundo não encontramos uma justificativa clara para a escolha pensando no âmbito da língua portuguesa, onde esse termo não tem sentido próprio. Poderíamos pensar que ele representa o sufixo do plural, dado que palavras que terminam com esta letra geralmente estão no plural, porém dada a estrutura do template, essa teoria foi descartada. Uma hipótese que podemos levantar, sabendo que este modelo de linguagem é multilíngue, é que de tal termo poderia ser o mesmo usado na língua inglesa para indicar posse, considerando que a maioria do treinamento deste modelo foi efetuado sobre textos em inglês.

Para o **Template 2-mT5**, a distribuição dos textos mais frequentes se deu de forma mais equilibrada em relação a todos os cenários apresentados anteriormente, incluindo os que utilizaram o modelo BERTimbau. Contudo, nenhum deles pode ser visto como uma resposta que ao preencher as máscaras das sentenças consiga indicar a classificação dos textos das notícias.

6.4 Respondendo às questões de pesquisa RQ2, RQ3 e RQ4

6.4.1 Conclusões sobre a RQ2

Foram executados diversos experimentos *zero-shot* como meio para responder à **RQ2**, que traz o seguinte questionamento:

> Modelos de linguagem pré-treinados em uma tarefa intermediária incluem como habilidade emergente a identificação da veracidade das notícias?

O principal desafio para utilizar estes modelos sem nenhum tipo de aperfeiçoamento, foi buscar formas de transmitir o objetivo da tarefa, que é classificar notícias falsas, sempre respeitando as características de cada arquitetura e as limitações de cada modelo de linguagem.

Na construção dos *prompts* e dos *templates*, infinitas possibilidades se apresentam para a combinação de palavras, de posicionamento de máscaras e de formatação no geral, o que faz com que os resultados obtidos não sejam absolutos e possam ser superados com a adesão de novas técnicas e explorações.

Dos modelos trabalhados, o *encoder-decoder* não conseguiu gerar resultados que pudessem ser equiparados às classes das notícias, mostrando que modelos desta arquitetura em suas versões menores encontram grande dificuldade em gerar textos que se traduzam em rótulos, dada a sua própria natureza geradora de texto.

Para a maioria dos registros, a saída do modelo gerou *tokens* especiais junto ao texto, que foram devidamente tratadas, porém isto gera dois possíveis alertas: (ii) A construção da máscara para este modelo precisa ser aprimorada de forma mais específica; (i) A versão multilíngue do modelo T5 em sua versão *small* não possui esta habilidade bem desenvolvida, o que tornaria este modelo inapto a identificar a veracidade de notícias apenas com o seu treinamento inicial.

O modelo BERTimbau, através do Template 2-BERTimbau, alcançou boas taxas de acerto para alguns datasets, mostrando como a posição da máscara tem forte influência sobre as escolhas do modelo acerca de qual termo utilizar. Esta evidência mostra que mesmo modelos menores podem apresentar habilidades para as quais não foram treinados explicitamente. Para o Template 1-BERTimbau, verificou-se que os únicos bons resultados alcançados foram com bases de mesma temática (Covid-19), porém com características distintas, o que põe em voga se a temática teve influência determinante no resultado ou

não, visto que os datasets MM-COVID e FakeCovid não são similares entre si. O primeiro possui textos curtos, o segundo tem textos medianos porém vários iniciando com sinais que poderiam ser reconhecidos pelo modelo como delimitadores, como aspas duplas, vírgulas, dentre outros.

Em suma, os experimentos *zero-shot* com BERTimbau de forma geral não obtiveram bons resultados, com acurácia média de 63% para o segundo template criado, mas podem servir como incentivo para investigações mais profundas sobre as habilidades envolvidas, visto que além da necessidade de se explorar outras formas de construção de abordagem, a classificação via preenchimento de máscara é muito sensível a ruídos no texto, como alta incidência de reticências por exemplo, ou o tamanho das sentenças.

Os modelos do tipo *decoder* obtiveram métricas melhores do que os experimentos deste tipo com BERTimbau, com o Sabiá-3 conseguindo uma acurácia média geral de 70%, com metade dos datasets avaliados por ele superando esta marca individualmente, mesmo recebendo um *prompt* com instruções diretas e de maneira bem simples. Outro *decoder* utilizado foi o Command, que ficou muito próximo do Sabiá-3, com uma acurácia média de 67%. Porém, ele apresentou mais falhas no envio das requisições, como também em relação ao entendimento das instruções recebidas, o que levou ao descarte de mais dados em comparação com o Sabiá-3. Além disso, o Command em alguns casos pareceu não detectar muito bem o idioma dos textos, o que pode ter sido superado por suas versões mais novas.

Embora eles tenham sido melhores em comparação com a iniciativa com BERTimbau sem *fine-tuning* aplicado, seus resultados ainda são insatisfatórios para afirmarmos que esta abordagem é um caminho viável para detecção de *fake news*, visto que os resultados para F1 macro não foram bons, com o melhor resultado geral sendo do próprio Command, com 54%. A criação de instruções para os modelos *decoder* de maneira geral, tem como ameaça à qualidade dos experimentos a limitação para explorar diferentes formas de montagem de *prompt* de instruções, porque embora os créditos tenham sido cedidos pela Cohere e pela Maritaca AI, continuam sendo recursos limitados.

Uma descoberta realizada por meio deste experimento é a possibilidade de utilização de LLMs como assistentes para detectar se um texto representa ou não uma alegação passível de ser verificada. O modelo Sabiá-3 demonstrou esta habilidade inesperada no escopo do estudo quando na respostas das requisições indicava que não poderia analisar se determinado era possivelmente falso ou não, dando explicações coerentes para tal postura. Este modelo ou outros com capacidade similar poderia ser utilizado em experimentos

futuros como um agente auxiliar na preparação das bases, para descartar exemplos que não indicam alegações válidas, e com isso aumentar a qualidade dos datasets de *fake news* disponíveis para uso.

Dados alguns resultados individuais de acurácia obtidos pelos LLMs, em especial o Command, gerados por uma abordagem sem um novo treinamento e com as sentenças dos exemplos sendo enviadas em pequenos blocos, se poderia argumentar que aqui a acurácia tem um peso maior do que nas validações *in-data* e *cross-data*. Contudo, generalizar envolve também saber discernir um conteúdo verdadeiro de um conteúdo mentiroso e medidas como F1 macro nos apoiam neste sentido.

Supondo que uma solução como esta pudesse ser utilizada como ferramenta de moderação de discursos em plataformas digitais por exemplo, ter LLMs que tendem a classificar os conteúdos como falsos, talvez pela incapacidade de recuperação de informações que respaldem suas respostas, poderia implicar no cerceamento da liberdade de indivíduos, mesmo que o conteúdo propagado por eles seja legítimo e legal.

Assim, respondendo à **RQ2**, dados os resultados obtidos pelos experimentos *zero-shot*, ainda não é possível afirmar que esta abordagem, a partir dos modelos aqui trabalhados e do método aplicado sejam meios confiáveis de detecção de *fake news* como suposta habilidade emergente de suas arquiteturas.

6.4.2 Conclusões sobre a RQ3

Para responder à **RQ3**, foi efetuado o *fine-tuning* dos modelos cohere-embeddings, BERTimbau e mT5. Esta questão de pesquisa levanta a seguinte indagação:

> **RQ3**: Qual o desempenho dos classificadores de diferentes tipos de instâncias de Transformers ao serem treinados e testados com dados da mesma base de dados?

A partir do que foi exposto na Tabela 12, os modelos de arquitetura *encoder* utilizados obtiveram resultados satisfatórios ao serem submetidos a dados que compartilham da mesma fonte dos utilizados no treinamento, com cinco classificadores obtendo F1 macro superior ou igual a 80% para o modelo cohere-embeddings, e oito classificadores do modelo BERTimbau com resultados neste mesmo intervalo.

Os classificadores gerados com o modelo mT5, representante da arquitetura *encoder-decoder*, foram muito impactados pela baixa oferta de exemplos em vários datasets, fa-

zendo com que oito classificadores tivessem um desempenho quase nulo ao avaliar dados da mesma base, não à toa, os classificadores treinados com os oito menores datasets. De maneira indireta, este tipo de arquitetura parece não lidar bem com datasets com menos de 2000 exemplos frente ao observado previamente. Uma possível estratégia seria a junção de exemplos de datasets semelhantes, como meio de reduzir a distância na quantidade de exemplos entre os datasets. Porém, para explorar os datasets o mais próximo possível de como os autores disponibilizaram e analisar as suas particularidades, seus dados permaneceram separados.

Por meio das análises efetuadas, buscamos compreender quais fatores podem influenciar na construção de classificadores de *fake news*, bem como verificar o que pode ter comprometido o treinamento dos classificadores com as demais bases de dados. Datasets com baixa demanda disponível de exemplos não oferecem informação suficiente para que os modelos possam treinar o reconhecimento de suas características e avaliar novos elementos. Alguns são mais suscetíveis, como BERTimbau e o mT5, este último de forma drástica. O modelo cohere-embeddings, por efetuar a classificação com busca semântica, não necessita de uma quantidade muito expressiva de dados.

Ainda sobre os modelos de linguagem utilizados, nos casos em que houve dataset com mais exemplos disponíveis, representados pelos classificadores treinados com as maiores bases, o modelo mT5 demonstrou usufruir disso ao máximo, enquanto o BERTimbau, e com maior dificuldade o cohere-embeddings, em raros momentos alcançavam valores altos na generalização entre datasets. Considerando que o mT5 recebeu um prefixo mínimo e que o seu pré-treinamento em português não é tão robusto, a lição aprendida é de que há um amplo espaço para novas pesquisas investigando a capacidade dos modelos desta arquitetura, seja com melhorias aplicadas ao treinamento, novos processos de *fine-tuning*, ou ainda a adesão de modelos maiores desta mesma família.

Outro fator observado foi o tamanho dos textos, referenciado ao longo deste trabalho preferencialmente pela quantidade média de palavras. Textos com poucas palavras têm dificuldade em transmitir ideias de modo geral, e muitos relatos de *fake news* são pequenas e médias narrativas sobre um determinado assunto. Além disso, se os textos contêm muitos sinais de pontuação, como exclamações e reticências isso pode prejudicar a performance do modelo desenvolvido. É sabido que textos com notícias falsas têm algumas características que dão um tom de alarde à informação transmitida, como o uso de várias palavras em caixa alta, que foi algo observado em todos os datasets, porém modelos menores não costumam lidar muito bem com esta profusão de sinais gráficos, o que sugere a busca de

alternativas de pré-processamento, ou mesmo a utilização de modelos que lidem melhor com a linguagem natural em curso.

Um ponto correlato e que pode gerar uma falsa sensação de detecção de *fake news* é o tipo de texto selecionado para representá-las. Um caso que ilustra bem isso é de que idealmente, os datasets que contêm alegações originais, além de mensagens trocadas em redes sociais, que também podem ser classificadas neste sentido quando bem apuradas, deveriam ter os melhores desempenhos, o que não se concretizou.

Para concluir esta questão de pesquisa, foi demonstrado através dos resultados apresentados para as validações *in-data* que é possível criar bons detectores de *fake news* considerando os dados dos datasets selecionados, porém há diversos ofensores que precisam ser averiguados para cada caso, como a quantidade de dado disponível, o tamanho dos textos, a qualidade dos textos e de maneira adicional o balanceamento de classes.

6.4.3 Conclusões sobre a RQ4

Por meio da utilização dos classificadores gerados no escopo da questão de pesquisa anterior, a **RQ4** propôs o seguinte:

> **RQ4**: Qual o desempenho desses mesmos classificadores ao serem testados com dados de fora da sua base original, de forma similar a como poderiam ser usados no mundo real?

Analisando a capacidade de generalização na classificação de exemplos de bases distintas, foram consideradas promissoras as validações *cross-data*, tendo como parâmetro de generalização aceitável aquelas com valores de métricas principal (F1 macro e similaridade cosseno) iguais ou superiores à 70%. Este valor é 10% abaixo do esperado como resultado mínimo aceitável nas validações *in-data*, pelo entendimento de que a inferência com o cruzamento de datasets é mais complexa e engloba diferentes frentes, que podem influenciar o experimento de muitas maneiras.

Tratando inicialmente dos modelos *encoder*, foram construídos 14 classificadores com o modelo BERTimbau e 11 com o modelo cohere-embeddings. Destes, dez conseguiram resultados na faixa estabelecida para o modelo BERTimbau e oito para o modelo cohere-embeddings. Esta contabilização considerou a existência de pelo menos um resultado a partir de 70% sobre datasets diferentes do utilizado durante o treinamento. Os classificadores na validação *in-data* se saíram melhor para quase todos os datasets, mas para

alguns conjuntos de dados, a diferença entre as métricas do primeiro e segundo melhores classificadores foi pequena, o que abre margem para a possibilidade de que os classificadores *cross-data* nesta situação possam ultrapassar o desempenho dos classificadores *in-data* mediante alguns ajustes.

Os melhores classificadores *in-data* treinados com os datasets Fake.Br, Fakepedia e FakeRecogna, obtiveram os melhores resultados na validação *cross-data*, conseguindo generalizar para mais de uma base de dados além da base de origem. As três bases em questão têm tamanhos próximos, fontes de notícia em comum, e alta similaridade, que pode ser vista tanto do ponto de vista semântico, com exemplos de uma mesma classe gerando agrupamentos de maneira similar, como também através da similaridade léxica, indicando que a capacidade de generalização está relacionada à similaridade da informação e identificar as características que norteiam esta similaridade pode ajudar na escolha do classificador mais apropriado para avaliar determinado conjunto de dados.

Seguindo esta linha, os classificadores treinados com datasets similares, além de obterem bons resultados, o seu desempenho é condicionado, a julgar pelos experimentos executados, ao tamanho do dataset utilizado para treinamento. Deste modo, o classificador treinado com mais dados consegue avaliar melhor os dados de base similar, talvez até melhor do que o classificador gerado com a própria base menor, como ocorreu na validação *cross-data* do modelo cohere-embeddings do classificador treinado com o dataset FakeNewsSet na avaliação dos dados do dataset Factck.BR, que alcançou 67%, enquanto que o classificador treinado com a mesma base obteve 57% de F1 macro.

Através da análise das bases, algumas limitações para a generalização foram observadas, como a dificuldade em detectar *fake news* sob textos com estilos diferentes. Classificadores que foram treinados com notícias de agências de checagem de fatos e de mídia tradicional ou governamental encontraram dificuldade em avaliar exemplos de redes sociais. Isso indica um entrave enorme, já que é neste ecossistema que a desinformação vigora, portanto detectar conteúdo falso imerso em diferentes estilos é um grande passo na alavancagem da capacidade de generalização de modelos voltados para este domínio.

Os resultados apresentados respondem à RQ4, mostrando que é viável construir modelos generalizáveis voltados para *fake news*, como meio de simular a utilização destes modelos no auxílio à detecção de conteúdos deste tipo no mundo real. Contudo, à luz de todos os experimentos e análises efetuadas, a capacidade de generalização se dá em níveis, começando por dados similares, onde ela tende a ser maior, e depois se expandindo para avaliar conteúdos de outras fontes e com estilos diferentes, o que ainda precisa ser inves-

Tabela 14: Comparativo do desempenho da metodologia aplicada para cada dataset, com os classificadores que obtiveram os melhores resultados de F1 macro para cada conjunto de dados.

Dataset	**Modelo**	**Base de treino**	**F1_macro**
BRACIS2019	cohere-embeddings	Central de Fatos	1
Central de Fatos	BERTimbau	in-data	0,75
COVID19BR	BERTimbau	in-data	0,82
Factck.BR	cohere-embeddings	FakeNewsSet	0,80
Fake.Br	BERTimbau	in-data	1
FakeCovid	BERTimbau	BRACIS2019, Central de Fatos, Factck.BR, FakeRecogna, MuMiN, X-FACT	1
	cohere-embeddings	BRACIS2019, Central de Fatos, FakeNewsSet, X-FACT	
FakeNewsSet	BERTimbau	in-data	0,93
Fakepedia	BERTimbau	in-data	0,99
FakeRecogna	BERTimbau	in-data	0,98
FakeTweet.Br	cohere-embeddings	in-data	0,80
FakeWhatsApp.Br	BERTimbau	in-data	0,83
MM-COVID	BERTimbau cohere-embeddings	COVID19BR	0,50
MuMiN	cohere-embeddings	Fake.Br	0,65
X-FACT	BERTimbau	in-data	0,68

tigado de modo mais profundo. A Tabela 14 apresenta os resultados vistos e discutidos para sabermos quais classificadores obtiveram melhor desempenho para cada dataset.

7 Conclusões

Esta dissertação investigou a capacidade de generalização de classificadores de *fake news* em português baseados em modelos de linguagem em diferentes instâncias da arquitetura Transformer. Para esta investigação, foram coletadas 14 bases de dados disponibilizadas em trabalhos anteriores, com diversas características, como fonte de coleta, quantidade de exemplos e forma de anotação. A investigação principal procurou identificar se um classificador treinado para uma base poderia ser utilizado para fazer inferências em outra base, em uma tentativa de elucidar se tais classificadores podem ser adotados no mundo real.

Ademais, foi investigado quais seriam as características mais propícias para a criação de classificadores generalizáveis, considerando a base de dados usada para o treinamento e a escolha da estratégia de treinamento. A metodologia proposta para avaliação da capacidade de generalização de modelos sobre bases de *fake news* em português não possui antecessores na literatura até onde sabemos.

Como estas bases foram curadas por diferentes autores, com diferentes objetivos e em diferentes momentos, foi observado que algumas delas eram mais alinhadas com a alegação original, principalmente por removerem resquícios indevidos do conteúdo dos sites que serviram como fonte de coleta destes dados, além de não permitirem duplicação. Bases de dados que não priorizaram tais tratamentos apresentaram menor qualidade, gerando classificadores menos adequados. Por meio da análise das bases de dados, também foi possível identificar práticas que podem ser adotadas na construção de novos datasets voltados para este domínio, vislumbrando a geração de conjuntos de dados mais completos e fáceis de manusear.

Ao final, concluímos que os classificadores de maneira geral ainda não conseguem discriminar de forma genérica *fake news* de notícias reais. Entretanto, alguns resultados apontam pesquisas futuras promissoras: os classificadores da arquitetura BERT mesmo em sua menor versão alcançaram resultados melhores do que os de modelos generativos

comerciais adotados nesta dissertação, os LLMs, indicando que modelos menores têm o potencial de serem adaptados para realizar tarefas específicas e sensíveis como a identificação de notícias falsas, com um custo menor. Além disso, os classificadores da arquitetura mT5, quando expostos a um grande número de exemplos foram capazes de generalizar e gerar rótulos muito próximos das classes originais dos exemplos, sendo mais um caminho promissor de pesquisa.

A respeito das bases de dados, identificamos que o desbalanceamento das bases ainda é um ponto que merece investigação. Se por um lado, priorizamos investigar as bases o mais próximo do que elas foram disponibilizadas, alguns conjuntos ficaram com mais de 90% de exemplos de notícias falsas, fazendo com que o treinamento dos classificadores não devolvesse resultados discriminatórios satisfatórios. No entanto, classificadores treinados com bases mais equilibradas e com mais exemplos disponíveis (a partir de 5.000) tiveram bons resultados para algumas bases com características próximas às utilizadas para o treinamento.

7.1 Limitações

Considerando os modelos de linguagem, a principal limitação para a utilização do BERTimbau e do mT5 foi a restrição quanto ao número máximo de *tokens* aceitos, de 512. Como consequência, a maioria dos textos precisou ser truncado para caber nesta quantidade de *tokens*, com o risco de descartar informações que poderiam conter dados relevantes para indicar a veracidade de uma notícia.

Quanto ao treinamento, conforme mencionado anteriormente, optamos por não balancear as bases, para que elas ficassem próximas a como foram disponibilizadas. Entretanto, estratégias de balanceamento poderiam gerar classificadores mais eficazes. A divisão aleatória dos dados embaralhados em somente três conjuntos de treino, teste e validação também podem ter introduzido vieses aos classificadores baseados no modelo cohere-embeddings, o que foi necessário considerando a quantidade de experimentos e os custos computacionais para executá-los.

Ainda considerando as estratégias de treinamento, os *templates* formulados para os experimentos *zero-shot* foram criados de maneira empírica. Outras formulações poderiam acarretar em resultados distintos, embora tenha ficado claro que a maioria dos modelos codificadores não consegue se adaptar a este tipo de inferência.

7.2 Trabalhos futuros

Mais bases de dados em português sobre *fake news* foram disponibilizadas após o período de levantamento de datasets estipulado neste trabalho, as quais poderiam ser submetidas à metodologia desenvolvida e gerar novos resultados para avaliação. Analogamente, outros modelos de linguagem também podem ter a sua capacidade de generalização explorada. Futuras frentes poderiam avaliar o impacto do aprimoramento dos datasets mapeados, visto que muitos deles sofrem com a presença de textos com ruído e dados faltantes. A realização de novas coletas de dados, combinadas a refinamentos no pré-processamento, poderiam elevar as métricas obtidas com os dados anteriores.

Como vimos ao longo desta dissertação, muitos datasets dispõem de poucos exemplos, o que prejudicou a construção de mais classificadores suficientemente generalizáveis. Uma possível abordagem seria juntar datasets que compartilham as mesmas fontes, gerando conjuntos de treinamento mais representativos. Outro aspecto correlato é a questão temporal dos dados, que não foi explorada neste trabalho, mas que poderia ser estudada com o intuito de criarmos conjuntos de dados que representassem determinado período. Para a avaliação da generalização dos modelos, os dados mais antigos seriam delegados ao treinamento dos modelos, e os dados mais recentes formariam os conjuntos de teste, simulando cenários reais.

Mais um caminho possível de extensão da pesquisa é a investigação da geração de classificadores generalizáveis entre diferentes estilos textuais, visto que a maior parte dos dados disponíveis pelos datasets são de notícias formais, enquanto que a propagação deste tipo de conteúdo se dá principalmente nas redes sociais, cuja comunicação escrita segue padrões totalmente distintos de matérias jornalísticas tradicionais. Por meio das análises realizadas sobre os dados, vimos ainda que exemplos que vêm de portais de notícias tendem a ter textos muito extensos, esbarrando em uma limitação dos modelos devido ao número máximo de *tokens* permitido. Além da avaliação do cruzamento de estilos de escrita, os textos mais extensos poderiam ser sumarizados, para reduzir o tamanho da informação sem perder o conteúdo da alegação.

Um ponto que pode ser explorado são novas estratégias para completar máscaras e sentenças, considerando sinônimos e palavras correlatas além do rótulo das classes. Um último caminho que se desenhou foi a investigação da capacidade dos modelos de linguagem utilizados conseguirem justificar as suas respostas, à luz do que foi visto nos retornos do modelo Sabiá-3 por exemplo, o que talvez contribua para a explicabilidade

da decisão dos modelos a partir da metodologia criada. Finalmente, esperamos que este trabalho possa ser um ponto de partida para a investigação de estratégias generalizáveis de geração de classificadores que os tornem adequados para serem adotados no mundo real.

REFERÊNCIAS

AIMEUR, Esma; AMRI, Sabrine; BRASSARD, Gilles. Fake news, disinformation and misinformation in social media: a review. **Social Network Analysis and Mining**, Springer, v. 13, n. 1, p. 30, 2023.

ALGHAMDI, Jawaher; LIN, Yuqing; LUO, Suhuai. Fake news detection in low-resource languages: A novel hybrid summarization approach. **Knowledge-Based Systems**, Elsevier, v. 296, p. 111884, 2024.

ALGHAMDI, Jawaher; LUO, Suhuai; LIN, Yuqing. A comprehensive survey on machine learning approaches for fake news detection. **Multimedia Tools and Applications**, Springer, v. 83, n. 17, p. 51009–51067, 2024.

ALLCOTT, Hunt; GENTZKOW, Matthew. Social media and fake news in the 2016 election. **Journal of economic perspectives**, American Economic Association 2014 Broadway, Suite 305, Nashville, TN 37203-2418, v. 31, n. 2, p. 211–236, 2017.

BALSHETWAR, Sarita V; RS, Abilash. Fake news detection in social media based on sentiment analysis using classifier techniques. **Multimedia tools and applications**, Springer, v. 82, n. 23, p. 35781–35811, 2023.

BERRAR, Daniel. Cross-Validation. In: RANGANATHAN, Shoba et al. (Ed.). **Encyclopedia of Bioinformatics and Computational Biology**. Oxford: Academic Press, 2019. P. 542–545. ISBN 978-0-12-811432-2. DOI: https://doi.org/10.1016/B978-0-12-809633-8.20349-X. Disponível em: <https://www.sciencedirect.com/science/article/pii/B978012809633820349X>.

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

CABRAL, Lucas et al. FakeWhastApp. BR: NLP and Machine Learning Techniques for Misinformation Detection in Brazilian Portuguese WhatsApp Messages. In: ICEIS (1). [S. l.: s. n.], 2021. P. 63–74.

CAPUANO, Nicola et al. Content-based fake news detection with machine and deep learning: a systematic review. **Neurocomputing**, Elsevier, v. 530, p. 91–103, 2023.

CASE, Donald O; GIVEN, Lisa M. **Looking for information: A survey of research on information seeking, needs, and behavior**. [S. l.]: Emerald Group Publishing, 2016.

CENTER FOR INFORMATION TECHNOLOGY AND SOCIETY. **A Brief History of Fake News**. Accessed: 2024-02-18. University of California, Santa Barbara. 2024. Disponível em: <https://cits.ucsb.edu/fake-news/brief-history>.

CHANG, Yupeng et al. A survey on evaluation of large language models. **ACM Transactions on Intelligent Systems and Technology**, ACM New York, NY, v. 15, n. 3, p. 1–45, 2024.

CHARLES, Anderson Cordeiro; RUBACK, Livia; OLIVEIRA, Jonice. Fakepedia corpus: A flexible fake news corpus in portuguese. In: SPRINGER. INTERNATIONAL Conference on Computational Processing of the Portuguese Language. [S. l.: s. n.], 2022. P. 37–45.

CITS - CENTER FOR INFORMATION TECHNOLOGY & SOCIETY. **A Brief History of Fake News**. [S. l.: s. n.], 2024. https://cits.ucsb.edu/fake-news/brief-history. [Online; Acesso em: 18 de fevereiro de 2024].

COOKE, Nicole A. Posttruth, truthiness, and alternative facts: Information behavior and critical information consumption for a new age. **The library quarterly**, University of Chicago Press Chicago, IL, v. 87, n. 3, p. 211–221, 2017.

CORDEIRO, Paulo Roberto; PINHEIRO, Vladia. Um corpus de notıcias falsas do twitter e verificaçao automática de rumores em lngua portuguesa. In: PROCEEDINGS of the Symposium in Information and Human Language Technology. [S. l.: s. n.], 2019. P. 219–228.

COUTO, Joao MM et al. Central de fatos: Um repositório de checagens de fatos. In: SBC. ANAIS do III Dataset Showcase Workshop. [S. l.: s. n.], 2021. P. 128–137.

DAME ADJIN-TETTEY, Theodora. Combating fake news, disinformation, and misinformation: Experimental evidence for media literacy education. **Cogent arts & humanities**, Taylor & Francis, v. 9, n. 1, p. 2037229, 2022.

DEVLIN, Jacob et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: PROCEEDINGS of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). [S. l.]: Association for Computational

Linguistics, 2019. P. 4171–4186. Disponível em: <https://arxiv.org/abs/1810.04805>.

DU, Jiangshu et al. Cross-lingual COVID-19 Fake News Detection. In: 2021 International Conference on Data Mining Workshops (ICDMW). [S. l.: s. n.], 2021. P. 859–862. DOI: 10.1109/ICDMW53433.2021.00110.

ECKER, Ullrich KH et al. The psychological drivers of misinformation belief and its resistance to correction. **Nature Reviews Psychology**, Nature Publishing Group US New York, v. 1, n. 1, p. 13–29, 2022.

FACTCHECK.ORG. **Our Staff - Brooks Jackson**. [S. l.: s. n.], 2024. https://www.factcheck.org/our-staff/. Acessado em: 28-April-2024.

FATOS, Aos. **O que é checagem de fatos — ou fact-checking?** [S. l.: s. n.], 2024. [Online; accessed 20-April-2024]. Disponível em: <https://www.aosfatos.org/checagem-de-fatos-ou-fact-checking/>.

FAUSTINI, Pedro; COVÕES, Thiago. Fake news detection using one-class classification. In: IEEE. 2019 8th Brazilian Conference on Intelligent Systems (BRACIS). [S. l.: s. n.], 2019. P. 592–597.

FIDALGO, Diana. **Cresce o jornalismo de checagem**. [S. l.: s. n.], 2017. [Online; accessed 20-April-2024]. Disponível em: <http://jornaldapuc.vrc.puc-rio.br/cgi/cgilua.exe/sys/start.htm?infoid=5272&sid=29>.

FISCHER, Marcelo et al. Identifying fake news in brazilian portuguese. In: SPRINGER. INTERNATIONAL Conference on Applications of Natural Language to Information Systems. [S. l.: s. n.], 2022. P. 111–118.

FONSECA, Bruno. **O que é fact-checking?** [S. l.: s. n.], 2017. https://apublica.org/checagem/2017/06/truco-o-que-e-fact-checking/. Acessado em 10 de abril de 2024.

GARCIA, Gabriel L; AFONSO, Luis CS; PAPA, João P. Fakerecogna: A new brazilian corpus for fake news detection. In: SPRINGER. INTERNATIONAL Conference on Computational Processing of the Portuguese Language. [S. l.: s. n.], 2022. P. 57–67.

GRAVES, Lucas; AMAZEEN, Michelle A. Fact-Checking as Idea and Practice in Journalism. **Oxford Research Encyclopedia of Communication**, 2019. Disponível em: <https://api.semanticscholar.org/CorpusID:159328645>.

GUPTA, Ashim; SRIKUMAR, Vivek. X-fact: A new benchmark dataset for multilingual fact checking. **arXiv preprint arXiv:2106.09248**, 2021.

HAMED, Suhaib Kh; AB AZIZ, Mohd Juzaiddin; YAAKUB, Mohd Ridzwan. A review of fake news detection approaches: A critical analysis of relevant studies and highlighting key challenges associated with the dataset, feature representation, and data fusion. **Heliyon**, Elsevier, 2023.

HARRISON, Matt. **Machine Learning–Guia de referência rápida: trabalhando com dados estruturados em Python**. [S. l.]: Novatec Editora, 2019.

HASHMI, Ehtesham et al. Advancing fake news detection: hybrid deep learning with fasttext and explainable AI. **IEEE Access**, IEEE, 2024.

HOY, Nathaniel; KOULOURI, Theodora. A Systematic Review on the Detection of Fake News Articles. **arXiv preprint arXiv:2110.11240**, 2021.

_____. Exploring the generalisability of fake news detection models. In: IEEE. 2022 IEEE International Conference on Big Data (Big Data). [S. l.: s. n.], 2022. P. 5731–5740.

HU, Beizhe et al. Bad actor, good advisor: Exploring the role of large language models in fake news detection. In: 20. PROCEEDINGS of the AAAI Conference on Artificial Intelligence. [S. l.: s. n.], 2024. v. 38, p. 22105–22113.

IFCN. **The Commitments - IFCN Code of Principles**. Accessed: 2024-04-29. Poynter Institute. 2024. Disponível em: <https://www.ifcncodeofprinciples.poynter.org/the-commitments>.

JONES-JANG, S Mo; MORTENSEN, Tara; LIU, Jingjing. Does media literacy help identification of fake news? Information literacy helps, but other literacies don't. **American behavioral scientist**, Sage Publications Sage CA: Los Angeles, CA, v. 65, n. 2, p. 371–388, 2021.

JURAFSKY, Daniel; MARTIN, James H. **Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition**. 3. ed. USA: Prentice Hall PTR, 2023.

KHAN, Junaed Younus et al. A benchmark study of machine learning models for online fake news detection. **Machine Learning with Applications**, v. 4, p. 100032, 2021. ISSN 2666-8270. DOI: https://doi.org/10.1016/j.mlwa.2021.100032. Disponível em: <https://www.sciencedirect.com/science/article/pii/S266682702100013X>.

KIM, Youngwook; PARK, Shinwoo; HAN, Yo-Sub. Generalizable implicit hate speech detection using contrastive learning. In: PROCEEDINGS of the 29th International Conference on Computational Linguistics. [S. l.: s. n.], 2022. P. 6667–6679.

LAZER, David MJ et al. The science of fake news. **Science**, American Association for the Advancement of Science, v. 359, n. 6380, p. 1094–1096, 2018.

LEWIS, Mike et al. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. **arXiv preprint arXiv:1910.13461**, 2019.

LI, Yichuan et al. Mm-covid: A multilingual and multimodal data repository for combating covid-19 disinformation. **arXiv preprint arXiv:2011.04088**, 2020.

LILLIE, Anders Edelbo; MIDDELBOE, Emil Refsgaard. Fake news detection using stance classification: A survey. **arXiv preprint arXiv:1907.00181**, 2019.

LOTH, Alexander; KAPPES, Martin; PAHL, Marc-Oliver. Blessing or curse? A survey on the Impact of Generative AI on Fake News. **arXiv preprint arXiv:2404.03021**, 2024.

LUPA. **De onde vem o fact-checking**. [S. l.: s. n.], 2015. https://lupa.uol.com.br/institucional/2015/10/15/de-onde-vem-o-fact-checking. Acessado em 28 de abril de 2024.

MAGAZINE, Politico. **The long and brutal history of fake news**. [S. l.: s. n.], 2016. Disponível em: https://www.politico.com/magazine/story/2016/12/fake-news-history-long-violent-214535/. Acesso em: 02 de fevereiro 2024.

MARTINS, Antônio Diogo Forte et al. COVID19. br: A dataset of misinformation about COVID-19 in brazilian portuguese whatsapp messages. In: SBC. ANAIS do III Dataset Showcase Workshop. [S. l.: s. n.], 2021. P. 138–147.

MASON, Lance E; KRUTKA, Dan; STODDARD, Jeremy. Media literacy, democracy, and the challenge of fake news. **Journal of Media Literacy Education**, v. 10, n. 2, p. 1–10, 2018.

MILLER, Stacy; MENARD, Philip; BOURRIE, David. I'm not fluent: How linguistic fluency, new media literacy, and personality traits influence fake news engagement behavior on social media. **Information & Management**, Elsevier, v. 61, n. 2, p. 103912, 2024.

MONTEIRO, Rafael A et al. Contributions to the study of fake news in portuguese: New corpus and automatic detection results. In: SPRINGER. COMPUTATIONAL Processing of the Portuguese Language: 13th International Conference, PROPOR 2018, Canela, Brazil, September 24–26, 2018, Proceedings 13. [S. l.: s. n.], 2018. P. 324–334.

MOREIRA, Lara Souto et al. A study of algorithm-based detection of fake news in brazilian election: Is bert the best. **IEEE Latin America Transactions**, IEEE, v. 21, n. 8, p. 897–903, 2023.

MORENO, João; BRESSAN, Graça. Factck. br: a new dataset to study fake news. In: PROCEEDINGS of the 25th Brazillian Symposium on Multimedia and the Web. [S. l.: s. n.], 2019. P. 525–527.

NAKOV, Preslav. Can We Spot the"Fake News"Before It Was Even Written? **arXiv preprint arXiv:2008.04374**, 2020.

NG, Lynnette Hui Xian; CARLEY, Kathleen M. Is my stance the same as your stance? A cross validation study of stance detection datasets. **Information Processing & Management**, Elsevier, v. 59, n. 6, p. 103070, 2022.

NIELSEN, Dan S; MCCONVILLE, Ryan. Mumin: A large-scale multilingual multimodal fact-checked misinformation social network dataset. In: PROCEEDINGS of the 45th international ACM SIGIR conference on research and development in information retrieval. [S. l.: s. n.], 2022. P. 3141–3153.

O'NEIL, Julie; GEDDES, David. An examination of the validity, reliability and best practices related to the standards for traditional media. **Research Journal of the Institute for Public Relations**, v. 2, n. 1, 2015.

OLAN, Femi et al. Fake news on social media: the impact on society. **Information Systems Frontiers**, Springer, v. 26, n. 2, p. 443–458, 2024.

OLIVEIRA, Thaiane et al. Confronting misinformation related to health and the environment: a systematic review. **Journal of Science Communication**, SISSA Medialab srl, v. 23, n. 1, p. v01, 2024.

PAES, Aline; VIANNA, Daniela; RODRIGUES, Jessica. Modelos de linguagem. In: CASELI, H. M.; NUNES, M. G. V. (Ed.). **Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português**. 2. ed. [S. l.]: BPLN, 2024. cap. 15. ISBN 978-65-00-95750-1. Disponível em: <https://brasileiraspln.com/livro-pln/2a-edicao/parte-modelos/cap-modelos-linguagem/cap-modelos-linguagem.html>.

PAN, Liangming; ZHANG, Yunxiang; KAN, Min-Yen. Investigating Zero-and Few-shot Generalization in Fact Verification. In: PROCEEDINGS of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers). [S. l.: s. n.], 2023. P. 511–524.

PARK, Minjung; CHAI, Sangmi. Constructing a User-Centered Fake News Detection Model by Using Classification Algorithms in Machine Learning Techniques (Jan 2023). **IEEE Access**, IEEE, 2023.

PAWLICKA, Aleksandra et al. AI vs linguistic-based human judgement: Bridging the gap in pursuit of truth for fake news detection. **Information Sciences**, Elsevier, v. 679, p. 121097, 2024.

PEI, Wenbin et al. A survey on unbalanced classification: How can evolutionary computation help? **IEEE Transactions on Evolutionary Computation**, IEEE, 2023.

PENNYCOOK, Gordon; RAND, David G. The psychology of fake news. **Trends in cognitive sciences**, Elsevier, v. 25, n. 5, p. 388–402, 2021.

PIRES, Ramon et al. Sabiá: Portuguese large language models. In: SPRINGER. BRAZILIAN Conference on Intelligent Systems. [S. l.: s. n.], 2023. P. 226–240.

POLETTO, Fabio et al. **Resources and benchmark corpora for hate speech detection: a systematic review**. v. 55. [S. l.]: Springer, 2021. P. 477–523.

POSETTI, Julie; MATTHEWS, Alice. A short guide to the history of 'fake news' and disinformation. **International Center for Journalists**, v. 7, n. 2018, p. 2018–07, 2018.

QUIÑONERO-CANDELA, Joaquin et al. **Dataset shift in machine learning**. [S. l.]: Mit Press, 2022.

RADFORD, Alec; NARASIMHAN, Karthik et al. Improving language understanding by generative pre-training. OpenAI, 2018.

RADFORD, Alec; WU, Jeffrey et al. Language models are unsupervised multitask learners. **OpenAI blog**, v. 1, n. 8, p. 9, 2019.

RAFFEL, Colin et al. Exploring the limits of transfer learning with a unified text-to-text transformer. **Journal of machine learning research**, v. 21, n. 140, p. 1–67, 2020.

RETHMEIER, Nils; AUGENSTEIN, Isabelle. **A Primer on Contrastive Pretraining in Language Processing: Methods, Lessons Learned, and Perspectives**. v. 55. New York, NY, USA: Association for Computing Machinery, fev. 2023. DOI: 10.1145/3561970. Disponível em: <https://doi.org/10.1145/3561970>.

RUDER, Sebastian. **Why You Should Do NLP Beyond English**. [S. l.: s. n.], 2020. http://ruder.io/nlp-beyond-english.

SANDRINI, Luca; SOMOGYI, Robert. Generative AI and deceptive news consumption. **Economics Letters**, Elsevier, v. 232, p. 111317, 2023.

SENO, Eloize Rossi Marques et al. Semântica distribucional. In: CASELI, H. M.; NUNES, M. G. V. (Ed.). **Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português**. 2. ed. [S. l.]: BPLN, 2024. cap. 10. ISBN 978-65-00-95750-1. Disponível em: <https://brasileiraspln.com/livro-pln/2a-edicao/parte-significado/cap-semantica-distribucional/cap-semantica-distribucional.html>.

SHAHI, Gautam Kishore; NANDINI, Durgesh. FakeCovid–A multilingual cross-domain fact check news dataset for COVID-19. **arXiv preprint arXiv:2006.11343**, 2020.

SHARMA, Karishma et al. Combating fake news: A survey on identification and mitigation techniques. **ACM Transactions on Intelligent Systems and Technology (TIST)**, ACM New York, NY, USA, v. 10, n. 3, p. 1–42, 2019.

SILVA, Flávio Roberto Matias da et al. Fakenewssetgen: A process to build datasets that support comparison among fake news detection methods. In: PROCEEDINGS of the Brazilian Symposium on Multimedia and the Web. [S. l.: s. n.], 2020. P. 241–248.

SILVA, Renato M et al. Towards automatically filtering fake news in Portuguese. **Expert Systems with Applications**, v. 146, p. 1–14, 2020.

SONG, Yisheng et al. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. **ACM Computing Surveys**, ACM New York, NY, v. 55, 13s, p. 1–40, 2023.

SOUZA, Fábio; NOGUEIRA, Rodrigo; LOTUFO, Roberto. BERTimbau: Pretrained BERT Models for Brazilian Portuguese. In_____. **Intelligent Systems: 9th Brazilian Conference**. Cham: Springer International Publishing, 2020. P. 403–417. ISBN 978-3-030-61377-8.

TAYLOR, Wilson L. "Cloze procedure": A new tool for measuring readability. **Journalism quarterly**, SAGE Publications Sage CA: Los Angeles, CA, v. 30, n. 4, p. 415–433, 1953.

THORNTON, Brian. The moon hoax: Debates about ethics in 1835 New York newspapers. **Journal of mass media ethics**, Taylor & Francis, v. 15, n. 2, p. 89–100, 2000.

TRAJANO, Douglas; BORDINI, Rafael H; VIEIRA, Renata. Olid-br: offensive language identification dataset for brazilian portuguese. **Language Resources and Evaluation**, Springer, p. 1–27, 2023.

TUNSTALL, Lewis; VON WERRA, Leandro; WOLF, Thomas. **Natural language processing with transformers**. [S. l.]: "O'Reilly Media, Inc.", 2022.

VALENTINI, Felipe; DAMASIO, Bruno Figueiredo. Average Variance Extracted and Composite Reliability: Reliability Coefficients/Variancia Media Extraida e Confiabilidade Composta: Indicadores de Precisao. **Psicologia: Teoria e Pesquisa**, v. 32, n. 2, na–na, 2016.

VAN DER MAATEN, Laurens; HINTON, Geoffrey. Visualizing data using t-SNE. **Journal of machine learning research**, v. 9, n. 11, 2008.

VAROL, Onur et al. Online human-bot interactions: Detection, estimation, and characterization. In: 1. PROCEEDINGS of the international AAAI conference on web and social media. [S. l.: s. n.], 2017. v. 11, p. 280–289.

VASWANI, Ashish et al. Attention is all you need. **Advances in neural information processing systems**, v. 30, 2017.

WĘCEL, Krzysztof et al. Artificial intelligence—friend or foe in fake news campaigns. **Economics and Business Review**, v. 9, n. 2, p. 41–70, 2023.

XUE, Linting et al. mT5: A massively multilingual pre-trained text-to-text transformer. **arXiv preprint arXiv:2010.11934**, 2020.

APÊNDICE A – INFORMAÇÕES ADICIONAIS SOBRE OS DATASETS SELECIONADOS

A Tabela 15 traz informações sobre a coleta dos dados originais, com o endereço dos repositórios disponibilizados por cada autor, a indicação da aplicação de transformação de dados sobre cada dataset e também se possui informações de data dos exemplos ali presentes. Também foi verificado nos artigos de divulgação dos datasets quais foram os períodos de coleta de dados, porém nem todos informaram este dado. Muitos trabalhos em contrapartida indicaram o período no qual estão compreendidas as notícias ou publicações que compõem as bases de dados. Por conta disso, a última coluna da Tabela 15 comporta os dois tipos de informação.

Tabela 15: Informações adicionais dos datasets utilizados nos experimentos. Todos os locais consultados foram obtidos através de informações fornecidas pelos autores dos trabalhos selecionados em suas respectivas publicações.

Dataset	**Local consultado**	**Passou por transformação**	**Possui informação de data**	**Período de coleta ou publicação**
BRACIS2019	https://github.com/phfaustini/BRACIS2019_FAKENEWS	✓		notícias coletadas no início de 2018
Central de Fatos	https://zenodo.org/records/5191798	✓	✓	notícias publicadas entre 2013 e 2021
COVID19BR	https://zenodo.org/records/5193932		✓	notícias coletadas entre abril de 2020 e junho de 2020
Factck.BR	https://github.com/jghm-f/FACTCK.BR		✓	notícias de agências de checagem de fatos brasileiras publicadas desde o começo de sua atuação até junho de 2019
Fake.Br	https://github.com/roneysco/Fake.br-Corpus/tree/master	✓		notícias publicadas entre janeiro de 2016 e janeiro de 2018
FakeCovid	https://github.com/Gautamshahi/FakeCovid/tree/master		✓	notícias publicadas entre 04/01/2020 e 15/05/2020
FakeNewsSet	https://github.com/kamplus/FakeNewsSetGen/		✓	notícias publicadas entre março de 2017 e maio de 2020
Fakepedia	https://github.com/andersoncordeiro/Fakepedia-Corpus	✓		notícias coletadas entre 2013 e 2021
FakeRecogna	https://github.com/recogna-lab/datasets/tree/master/FakeRecogna		✓	notícias publicadas entre 2019 e 2021
FakeTweet.Br	https://github.com/prc992/FakeTweet.Br	✓	✓	
FakeWhatsApp.Br	https://github.com/cabrau/FakeWhatsApp.Br		✓	notícias publicadas entre março de 2017 e maio de 2020
MM-COVID	https://github.com/bigheiniu/MM-COVID	✓	✓	notícias coletadas entre fevereiro de 2020 e julho de 2020
MuMiN	https://github.com/MuMiN-dataset	✓	✓	
X-FACT	https://github.com/utahnlp/x-fact/	✓	✓	