UNIVERSIDADE FEDERAL FLUMINENSE

**FERNANDO PEREIRA CARNEIRO**

# BERTWEET.BR: A PRE-TRAINED LANGUAGE MODEL FOR TWEETS IN PORTUGUESE

NITERÓI

2023

**FERNANDO PEREIRA CARNEIRO**

# BERTWEET.BR: A PRE-TRAINED LANGUAGE MODEL FOR TWEETS IN PORTUGUESE

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

Coorientador:
ALEXANDRE PLASTINO DE CARVALHO

Coorientadora:
DANIELA QUITETE DE CAMPOS VIANNA

NITERÓI

2023

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

C289b Carneiro, Fernando Pereira
BERTweet.BR: a pre-trained language model for tweets in portuguese / Fernando Pereira Carneiro. - 2023.
77 f.: il.

Orientador: Aline Martins Paes Carvalho.
Coorientador: Alexandre Plastino; Daniela Quitete de Campos Vianna.
Dissertação (mestrado)-Universidade Federal Fluminense, Instituto de Computação, Niterói, 2023.

1. Análise de sentimentos. 2. Processamento de linguagem natural. 3. Modelo de linguagem. 4. Twitter. 5. Produção intelectual. I. Carvalho, Aline Martins Paes, orientadora. II. Plastino, Alexandre, coorientador. III. Vianna, Daniela Quitete de Campos, coorientadora. IV. Universidade Federal Fluminense. Instituto de Computação.V. Título.

CDD - XXX

Bibliotecário responsável: Debora do Nascimento - CRB7/6368

FERNANDO PEREIRA CARNEIRO

BERTWEET.BR: A PRE-TRAINED LANGUAGE MODEL FOR TWEETS IN PORTUGUESE

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 Agosto de 2023.

BANCA EXAMINADORA

Profa. ALINE MARINS PAES CARVALHO - Orientadora, UFF

Assinado de forma digital por Alexandre Plastino de Carvalho
Dados: 2023.08.24 14:11:33 -03'00'

Prof. ALEXANDRE PLASTINO DE CARVALHO - Coorientador, UFF

Dra. DANIELA QUITETE DE CAMPOS VIANNA - Coorientadora, UFAM

gov.br Documento assinado digitalmente
NADIA FELIX FELIPE DA SILVA
Data: 24/08/2023 13:00:06-0300
Verifique em https://validar.iti.gov.br

Profa. NÁDIA FÉLIX FELIPE DA SILVA, UFC

Profa. FLAVIA CRISTINA BERNARDINI, UFF

Niterói
2023

*This work is dedicated in loving memory of my dearest father, Raimundo Carneiro, who has always been a source of inspiration and unwavering support throughout my academic journey. This accomplishment is a tribute to his legacy.*

# Agradecimentos

I would like to express my gratitude to my advisors, Aline, Alexandre and Daniela, for their invaluable guidance and constructive feedback throughout the course of this work.

I am deeply thankful to my parents, Raimundo and Teresinha for their love and immeasurable sacrifices in providing me with the best possible education. My sisters Fabiana, Fernanda and Flávia for their unwavering support. Your constant belief in my potential has been the foundation of my academic success.

To my loving wife Marianna, thank you for being my rock and for always standing by my side. Your understanding and love have been vital, and I am truly grateful for the bond we share. Finally, to my precious daughter, Pietra. You are the light of my life, and your presence has been a constant inspiration for me to strive for excellence.

# Resumo

A maioria dos avanços recentes nos modelos de língua neurais são avaliados em *benchmarks* e tarefas primordialmente em uma língua, o inglês. Menos atenção é dada a mais de sete mil outras línguas, faladas por aproximadamente 6.5 bilhões de pessoas ao redor do mundo. Uma delas é o português: apesar de ser a sexta língua mais falada no mundo, ainda existem muito menos recursos linguísticos para treinamento e avaliação de redes neurais em português, em comparação com o inglês. Notavelmente, os usuários de língua portuguesa compõem um dos grupos mais ativos de usuários do Twitter; no entanto, nenhum modelo de língua pré-treinado em tweets em português foi estudado extensivamente na literatura. Além da língua, os modelos pré-treinados baseados em tweets devem levar em conta aspectos culturais, o estilo linguístico informal, emprego de símbolos e o número limitado de caracteres. Esta dissertação busca endereçar essa lacuna ao introduzir o **BERTweet.BR**, o primeiro modelo pré-treinado em larga escala específico para o domínio de tweets em português do Brasil. O modelo BERTweet.BR possui a mesma arquitetura do $\text{BERTweet}_{\text{base}}$, tendo sido treinado do zero seguindo o procedimento de pré-treinamento do modelo RoBERTa em um *corpus* de 100M de tweets em português. Na tarefa de análise de sentimentos, os experimentos mostram que o BERTweet.BR supera três modelos multilíngues baseados na arquitetura dos Transformers, além do BERTimbau, um modelo de Transformers genérico pré-treinado especificamente para o português do Brasil. Desta forma, fica demonstrado que o modelo de língua BERTweet.BR possui grande potencial para fomentar novas pesquisas em tarefas analíticas para tweets em Português.

**Palavras-chave**: análise de sentimento , twitter , modelos de linguagem , arquitetura de transformers , extração de atributos de modelos , ajuste fino de modelos , adaptação de domínio , pré-treinamento continuado.

# Abstract

Most recent progress in neural language models predominantly focuses on one language, English. Less attention is given to the more than seven thousand others, spoken by approximately 6.5 billion people around the world. One of these is Portuguese: despite being the sixth most spoken language in the world, still has fewer neural-based linguistic resources compared to English. Notably, Portuguese speakers compose one of the most active groups of Twitter users; however, no pre-trained language model for Portuguese tweets has been extensively studied in the literature. Besides the language, tweets-based pre-trained models must account for the cultural code, informal linguistic style, code-switching, and the limited number of characters. This manuscript addresses this gap by introducing **BERTweet.BR**, the first publicly available large-scale pre-trained model specifically for the Brazilian Portuguese tweets domain. BERTweet.BR has the same architecture as $\text{BERTweet}_{\text{base}}$, a BERT-based model for English tweets, and was trained from scratch following the RoBERTa pre-training procedure on a 100M Portuguese tweets *corpus*. On the sentiment analysis task, experiments show that BERTweet.BR outperforms three multilingual Transformers and BERTimbau, a monolingual general-domain Brazilian Portuguese language model. Thus, BERTweet.BR language model demonstrates significant potential to foster new research in analytical tasks for Portuguese tweets.

**Keywords**: sentiment analysis , twitter , language model , transformer , feature-based , fine-tuning , domain adaptation , continued pre-training.

# List of Figures

# List of Tables

# List of Abbreviations and Acronyms

**BERT** Bidirectional Encoder Representations for Transformers

**DAPT** Domain Adaptative Pre-Training

**GPT** Generative Pre-trained Transformer

**LLM** Large Language Model

**LM** Language Model

**LR** Logistic Regression

**MLM** Masked Language Model

**NER** Named Entity Recognition

**NLP** Natural Language Processing

**NMT** Neural Machine Translation

**NSP** Next Sentence Prediction

**OOV** Out-of-Vocabulary

**PoS** Part-of-Speech

**RNN** Recurrent Neural Network

**TAPT** Task Adaptative Pre-Training

**VSM** Vector Space Models

# Contents

# 1 Introduction

Vector Space Models (VSM) (SALTON; WONG; YANG, 1975) are one of the earliest and most common strategies adopted and for many years remained the standard technique for language representation in Natural Language Processing (NLP) tasks. Proposed in 2013, Word2Vec (MIKOLOV; SUTSKEVER, et al., 2013) was widely adopted for its efficiency and ease of use. Since then, the model training pipeline for NLP tasks using word-embeddings remained essentially unchanged: word embeddings pre-trained on large amounts of unlabeled data through algorithms such as Word2Vec (MIKOLOV; SUTSKEVER, et al., 2013), GloVe (PENNINGTON; SOCHER; MANNING, 2014) and FastText (MIKOLOV; GRAVE, et al., 2018) were widely used to initialize the first layer of a predictive model. While pre-trained word vectors have been immensely dominant and successfully applied to a variety of tasks, they had two significant limitations: i) they are shallow approaches only incorporating previous knowledge in the first layer of the model; ii) they are part of a group of static methods that do not take multiple contexts into account when generating embeddings, meaning that a single vector is generated to represent each word in the dictionary, ignoring the different meanings a word can assume in that language.

The introduction of contextualized embeddings – which address polysemy by allowing distinct embeddings for the same word, depending on the context it appears – opened up a new era for deep learning-based models suited for NLP tasks (PETERS et al., 2018; HOWARD; RUDER, 2018). Notably, the Transformers (VASWANI et al., 2017), further explored in models like BERT (DEVLIN et al., 2019) and GPT (RADFORD et al., 2018), demonstrated that pre-trained multi-layer language models based on attention mechanisms could be easily embedded into a transfer learning strategy to obtain state-of-the-art results in a wide range of downstream tasks[1], even in scenarios with only a few la-

[1]In the context of NLP, downstream tasks are tasks that can be performed using a pre-trained language representation model, such as *BERTweet.BR*. These tasks typically require a deeper understanding of the text than the tasks used to pre-train the model. Some examples of downstream tasks include text classification, natural language inference, named entity recognition (NER), question answering, machine translation, and summarization. Downstream tasks are an important part of NLP because they allow us to use pre-trained models to solve real-world problems. By fine-tuning a pre-trained model on a

beled data. Following BERT, multiple contextualized language models have been trained from heterogeneous and conventional text corpora extracted from sources such as books, Wikipedia[2], and news sites (ZHUANG et al., 2021; SANH et al., 2019; LAN et al., 2020).

While those generic-domain representations have achieved remarkable performance across many tasks with multiple datasets taken from a variety of sources (WANG; SINGH, et al., 2018; WANG; PRUKSACHATKUN, et al., 2019), studies have shown that training domain-specific language models can deliver significant gains when dealing with specific textual contexts (GURURANGAN et al., 2020; LEE et al., 2020). BERTweet (NGUYEN; VU; NGUYEN, 2020), for example, was trained exclusively from tweets in English to capture the informal characteristics present in short texts typical of the Twitter[3] platform. Experiments showed that BERTweet outperforms the baselines RoBERTa$_{\text{base}}$ (ZHUANG et al., 2021) and XLM-R$_{\text{base}}$ (CONNEAU et al., 2020) on three Twitter NLP tasks.

Regarding languages other than English, there is some effort on domain adaptation of large-scale multilingual language models to tweets (BARBIERI; ESPINOSA-ANKE; CAMACHO-COLLADOS, 2022). However, a question arises as to whether the performance of monolingual models is better, given the specific particularities of tweets. This way, many recent works have released large-scale language models adapted to the Twitter domain in languages other than English, such as French (GUO et al., 2021), Indonesian (KOTO; LAU; BALDWIN, 2021), Spanish (HUERTAS-TATO; MARTIN; CAMACHO, 2022; GONZÁLEZ; HURTADO; PLA, 2020), Arabic (ABDELALI et al., 2021), and Italian (POLIGNANO et al., 2019).

On the other hand, several thousand other languages remain neglected. Portuguese, for example, is the sixth most spoken language in the world, with 258 million Portuguese speakers (EBERHARD; SIMONS; FENNIG, 2023), and the fifth most used language on the Internet (INTERNET WORLD STATS, 2020). Brazil has the world's largest population of speakers from Portuguese-speaking countries: approximately 214 million people. Even though only 70% of Brazilians have regular internet access, Brazilians spend more time on the Internet than watching TV. Indeed, Brazil is responsible for 10% of the total time spent on social media globally, positioned in second place, only behind the United States (DATA REPORTAL, 2021). Along with Spanish, Japanese, and Indonesian-speaking users, Portuguese speakers are among the most active voices on Twitter (HONG; CONVERTINO; CHI, 2011), being the fifth country regarding the number of Twitter users (STATISTA, 2021).

*downstream task*, we can improve its performance and make it more useful for practical applications. In the context of NLP, the terms *downstream task* and *final task* are used interchangeably

[2] https://www.wikipedia.org/

[3] http://www.twitter.com

These numbers indicate the enormous potential and need for developing language models specifically addressing Twitter-based NLP tasks in Portuguese. Although some previous studies fine-tuned contextualized embeddings for Portuguese tweets (SOUZA; NOGUEIRA; LOTUFO, 2020), the question remains whether a model trained specifically for Portuguese tweets performs better than adapting existing multilingual or formal language models to tweets-based tasks. However, no pre-trained language model from a large-scale corpus of tweets for Portuguese is extensively studied in the literature.

In *Sentiment analysis in Portuguese tweets: an evaluation of diverse word representation models*, Vianna et al. (2023) thoroughly examined the effectiveness of static embeddings and contextualized Transformers-based language models as feature extractors in the context of Portuguese tweets sentiment classification. In this newly released research that forms the foundation of this dissertation, we carried out experiments using the base weights of pre-trained models along with three additional versions of these models that were adapted through continued pre-training strategies. While the study explored different pipelines, it was constrained by using existing language models only. Then, as a future research direction, Vianna et al. (2023) proposed to investigate whether training a new language model from scratch using Portuguese tweets could advance the research in NLP analytical tasks for Portuguese tweets, given the unique characteristics of the language and the informal and noisy nature of tweets.

In this context, we introduce *BERTweet.BR*, a public large-scale pre-trained model specific to the Brazilian Portuguese tweets domain. *BERTweet.BR* has the same architecture of $\text{BERTweet}_{\text{base}}$ (NGUYEN; VU; NGUYEN, 2020). Likewise, it was trained from scratch following RoBERTa (ZHUANG et al., 2021) pre-training procedure on a *corpus* of approximately 9 GB containing 100M Portuguese tweets.

To evaluate *BERTweet.BR*, we also follow (VIANNA et al., 2023) and selected the sentiment analysis task (LIU, 2020) as the final task, given its broad application scenarios, enabling companies and governments to gain valuable insights into people's attitudes and perceptions, from political opinion (SANTOS; BERNARDINI; PAES, 2021) to stock market (OLIVEIRA CAROSIA; COELHO; SILVA, 2020). The evaluation pipelines we employed include the same collection of eight manually annotated datasets from Vianna et al. (2023), five of which have three classes, while the rest are binary. We compared the performance of *BERTweet.BR* to a broad set of contextualized transformer-based models containing language-specific, multilingual, and Twitter-adapted models. To ascertain the adoption of *BERTweet.BR* and show its predictive superiority on that task, we elicited two groups of

experiments: (i.) adapting the language model that induces contextualized embeddings and (ii.) training classifiers from ready-to-use or adapted embeddings coming from the language models.

## 1.1 Research Questions

In this dissertation, we designed experiments to investigate the following research questions.

- **RQ1:** *How does BERTweet.BR compare to adapting existing language models to the tweets domain?*

  We follow the two approaches of (GURURANGAN et al., 2020) to adapt pre-trained language models and unfold this research question in the following two.

  - **RQ1.1:** *What is the predictive performance gain of the pre-trained BERTweet.BR model, when compared to adapting existing generic monolingual and multilingual models to the generic domain of tweets?*

    To answer this question, we adapted the most used Portuguese monolingual model, BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020), and the multilingual models mBERT (DEVLIN et al., 2019) and XLM-R (CONNEAU et al., 2020) to the same dataset BERTweet.BR was pre-trained following the Domain Adaptative Pre-Training (DAPT) procedure (GURURANGAN et al., 2020).

  - **RQ1.2:** *What is the predictive performance gain of the pre-trained BERTweet.BR model, when compared to adapting existing generic monolingual and multilingual models to sentiment-prone tweets?*

    To answer this question, we adapted the models mentioned in RQ1.1 to tweet sentiment datasets following the Task Adaptative Pre-Training (TAPT) procedure (GURURANGAN et al., 2020).

- **RQ2:** *What is the difference in the predictive performance of fine-tuning to the downstream tasks of the tweets-trained or adapted embeddings compared to using them without fine-tuning?*

  This research question unfolds in the following two.

  - **RQ2.1** *What is the predictive performance gain of extracting the embeddings from BERTweet.BR, when compared to executing this same procedure in the other contextualized models?*

We followed a feature-based approach to answer this question and trained a logistic regression classifier with embeddings extracted from BERTweet.BR, original and adapted BERTimbau, mBERT, and XLM-R. Additionally, we elicited another model trained from multilingual tweets, namely, the XLM-T(BARBIERI; ESPINOSA-ANKE; CAMACHO-COLLADOS, 2022) model.

– **RQ2.2** *What is the predictive performance gain of fine-tuning BERTweet.BR to the downstream datasets, when compared to executing this same procedure in the other contextualized models?*

To answer this question, we fine-tuned the original and adapted contextualized models mentioned in RQ2.1.

In addition to answer the research questions, we compare *BERTweet.BR* model's predictive performance to a fastText-based classifier (MIKOLOV; GRAVE, et al., 2018) as a baseline result. Experiments showed that our model consistently outperforms mBERT, BERTimbau, XLM-R, and XLM-T in most cases and the static word embeddings from fastText (MIKOLOV; GRAVE, et al., 2018) in all the experiments.

## 1.2 Contributions

Brief, we highlight the main contributions of this work:

- A study titled *Sentiment analysis in Portuguese tweets: an evaluation of diverse word representation models* where Vianna et al. (2023) thoroughly examined the effectiveness of static embeddings and contextualized language models following a *feature-based* approach in the context of sentiment classification. This forms the bedrock of this dissertation and has been recently published in the Language Resources and Evaluation (LREV)[4], a Qualis-A1 publication devoted to the acquisition, creation, annotation, and use of language resources.

- We introduce *BERTweet.BR*, a large-scale pre-trained language model for Brazilian Portuguese tweets. We empirically show that it outperforms a large set of training strategies and models in sentiment classification using datasets of distinct characteristics and sizes, certifying the effectiveness of a domain-specific language model pre-trained for Portuguese tweets.

[4] https://www.springer.com/journal/10579

- A set of experiments investigating how static and transformer-based word embeddings for Portuguese, trained on domains other than tweets, perform in the Twitter sentiment analysis task.

- A munuscript called *BERTweet.BR: A Pre-Trained Language Model for Tweets in Portuguese* which has recently been submitted to the Computational Linguistics[5], a Qualis-A1 journal sponsored by the Association for Computational Linguistics (ACL)[6].

- To facilitate future research on Portuguese tweets, we made *BERTweet.BR* publicly available on Huggingface's model hub[7]. Also, we open-sourced all the code and documentation on Github[8].

## 1.3 Organization

The rest of the dissertation is organized as follows. In Section 2, we introduce the concepts necessary for understanding the architecture and pre-training approach of the proposed language model *BERTweet.BR* as well as a description of related studies previously investigated in the literature. Section 3 outlines the architecture, dataset, and optimization setup adopted to pre-train the *BERTweet.BR* language model. Section 4 presents the workflow of the experiments carried out in this dissertation. In Section 5, we present the experimental results achieved by responding to the research questions introduced in this section. Next, in Section 6, we bring a qualitative analysis of the conducted experiments; in Section 7, we present the conclusions of this study and future research directions. Finally, in Appendix A we describe the eight datasets employed to evaluate *BERTweet.BR* on sentiment analysis.

[5]Computational Linguistics is the longest-running publication devoted exclusively to the computational and mathematical properties of language and the design and analysis of natural language processing systems. This highly regarded quarterly offers university and industry linguists, computational linguists, artificial intelligence and machine learning investigators, cognitive scientists, speech specialists, and philosophers the latest information about the computational aspects of all the facets of language research. https://cljournal.org/

[6]https://www.aclweb.org/

[7]https://huggingface.co/melll-uff/bertweetbr

[8]https://github.com/MeLLL-UFF/BERTweet.br

# 2 Key Concepts and Literature Review

This chapter introduces concepts necessary for understanding the architecture and pre-training approach of the proposed language model *BERTweet.BR*, as well as the fundamental terms related to the benchmark models we used to evaluate and compare the performance of *BERTweet.BR*. Also, it describes related studies previously investigated in the literature.

## 2.1 Language Models

In the domain of computational linguistics, *language models* refer to a subset of probabilistic models that are tasked with predicting the next or a masked word, given the previous or the surrounding words in a sentence (JURAFSKY; MARTIN, 2000). They are designed to encapsulate the syntactical structure and semantic context of natural languages, aiming at enabling machines to generate human-like text. With the emergence of deep learning techniques, particularly the introduction of the Transformers architecture (VASWANI et al., 2017), neural language models have become increasingly prominent due to their enhanced ability to capture longer dependencies and model semantic relations more effectively. They are the backbone of several state-of-the-art NLP systems today. More recent models, commonly called large language models (LLM) and powered by billions of parameters, demonstrate an unprecedented capacity to generate human-like text, comprehend complex textual contexts, and perform sophisticated language tasks.

Transformers models like BERT (DEVLIN et al., 2019), RoBERTa (ZHUANG et al., 2021), and BERTweet (NGUYEN; VU; NGUYEN, 2020) are trained as language models, meaning they have been trained on large amounts of raw text in a self-supervised fashion. Self-supervised learning leverages abundant unlabeled text data from sources like books and the Internet and automatically generates labeled data from this unannotated corpus, thereby eliminating humans' need for manual data labeling. On top of this data, this type of model develops a statistical and generic understanding of the language it has

been trained on, but may not readily apply to specific practical tasks. Because of this, the generic *pre-trained model* resulting from this first training stage then goes through a process called *transfer learning*, as shown in Figure 1. During this process, the model is now fine-tuned in a supervised way — using human-annotated labels — on a given downstream task.

**Static and Contextualized Language Models.** There are two primary types of language models:*static* and *contextualized* language models. Static language models like Word2Vec (MIKOLOV; SUTSKEVER, et al., 2013), GloVe (PENNINGTON; SOCHER; MANNING, 2014), and FastText (MIKOLOV; GRAVE, et al., 2018) are models that map words to fixed-length vectors, representing their semantic meanings based on their co-occurrence patterns in a text corpus. They are trained on a large corpus of text and produce a fixed representation of each word independently of the context in which the word appears. For example, the word "bank" can mean a financial institution or a river bank. A static language model would produce the same representation for both meanings, regardless of the context. Static models are relatively simple to train and can be used for various tasks, such as text classification and question answering. While these models have proven effective in various NLP tasks, they exhibit limitations in capturing polysemy, assigning the same vector to a word regardless of its context. This can lead to problems with tasks that require understanding the meaning of a word in a particular scenario, such as sentiment analysis and machine translation. On the other hand, contextualized language models such as BERT (DEVLIN et al., 2019), GPT (RADFORD et al., 2018), ELMo (PETERS et al., 2018), and RoBERTa (ZHUANG et al., 2021) generate dynamic word embeddings based on the surrounding context of a word in a given sentence. They are trained on a very large corpus of text and represent each word that is dependent on the context in which the word appears. This means the same word can have different representations depending on the context. For example, the word "bank" might have a different representation in the sentence "I went to the bank" than in the sentence "The river bank was flooded". Although they lead to better performance in various tasks, they are more computationally expensive, slower to train, and require sophisticated architecture and fine-tuning strategies compared to static models. This increased complexity can make them more challenging to implement and optimize.

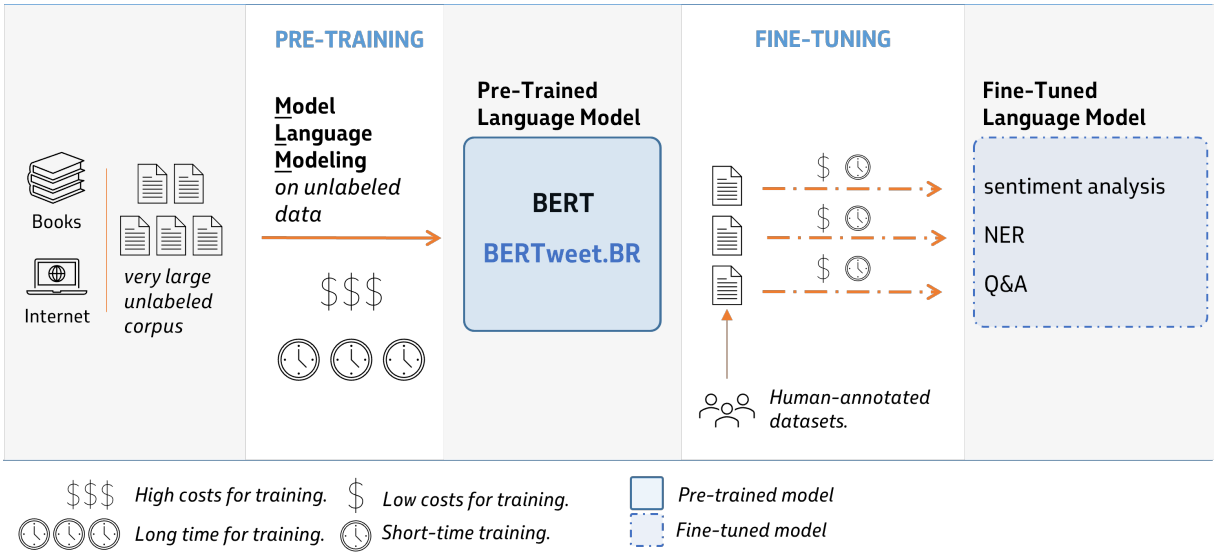


Figure 1: Transfer learning approach for language models: pre-training and fine-tuning.

## 2.2 Training Pipelines

In the standard transfer learning setup (Figure 1), a model is first pre-trained on large amounts of unlabeled data using a language modeling loss such as *casual language modeling* or *masked language modeling* (DEVLIN et al., 2019). The pre-trained model is then fine-tuned on labeled data of a downstream task using a standard cross-entropy loss.

In most cases, the *pre-training* step is called training a model from scratch. The model weights are randomly initialized, and the training begins without prior knowledge. This pre-training phase usually requires a significant amount of unannotated data. Training is usually expensive and can take several weeks to complete.

*Fine-tuning*, on the other hand, is the training done after a model has been pre-trained. To perform fine-tuning, you first acquire a pre-trained language model, then perform additional training with a dataset specific to your task. Typically, the pre-trained model was already trained on a dataset similar to the fine-tuning dataset. The fine-tuning process can thus take advantage of the knowledge acquired by the initial model during pre-training. Since the pre-trained model was already trained on lots of data, the fine-tuning requires way less data to get decent results, and the amount of time and resources needed to get good results are much lower.

For instance, a pre-trained model, initially trained on heterogeneous and general-domain data from books and English Wikipedia, could be subsequently fine-tuned to generate a specialized model designed explicitly for classifying emotions. Fine-tuning this

pre-trained model necessitates a limited amount of data, as the acquired knowledge of the pre-trained model is "transferred" to the target task, thereby demonstrating the principle of transfer learning. Consequently, fine-tuning a model incurs lower temporal, data, financial, and environmental costs than training a model from scratch. This is a very similar pipeline followed, for example, by the text classification model *roberta-base-go_ emotions*[1]. This is a fine-tuned model on top of $\text{RoBERTa}_{\text{base}}$ (ZHUANG et al., 2021) for multi-label emotion classification task from *GoEmotions* (DEMSZKY et al., 2020)dataset[2]. In this case, the $\text{RoBERTa}_{\text{base}}$ is the base language model which has been pre-trained on a large corpus of books, the Wikipedia and news sites and made publicly available by authors. Then, to create *roberta-base-go_ emotions* a public annotated dataset containing 58 thousand comments from Reddit[3] platform was used to fine-tune a sequence classification model for three epochs. The resulting model can now classify unseen comments text into one or more of its 28 categories.

## 2.3 Domain Adaptation

Pre-trained language models such as BERT and RoBERTa have demonstrated remarkable performance in capturing rich semantic representations by training on diverse text sources. However, these models may exhibit suboptimal performance when applied to contexts significantly differing from their source domain. As demonstrated by (GURURANGAN et al., 2020), the more distinct the target and source domains are, the greater the degradation of generic models (source) when used in specific domains (target), and the higher will be the potential of techniques for *model language domain adaptation*. Following this procedure, a pre-trained LM undergoes additional pre-training steps to adapt it to a desired target domain.

Specifically, one strategy involves continuing the pre-training of an LM like BERT using specialized corpus from the context of the target domain (Figure 2). Following this approach, legal documents and scientific papers, for example, were utilized to fine-tune the weights of a BERT model to create adapted versions of this LM that captures the specific jargon and nuances of these new contexts resulting in SciBERT (BELTAGY; LO; COHAN, 2019) and LegalBERT (CHALKIDIS et al., 2020) language models. Exposing the model to domain-specific data during the continued pre-training phase can better understand the target domain and enhance its performance on tasks within that domain. Note that this

[1] https://huggingface.co/SamLowe/roberta-base-go_emotions
[2] https://huggingface.co/datasets/go_emotions
[3] https://www.reddit.com/

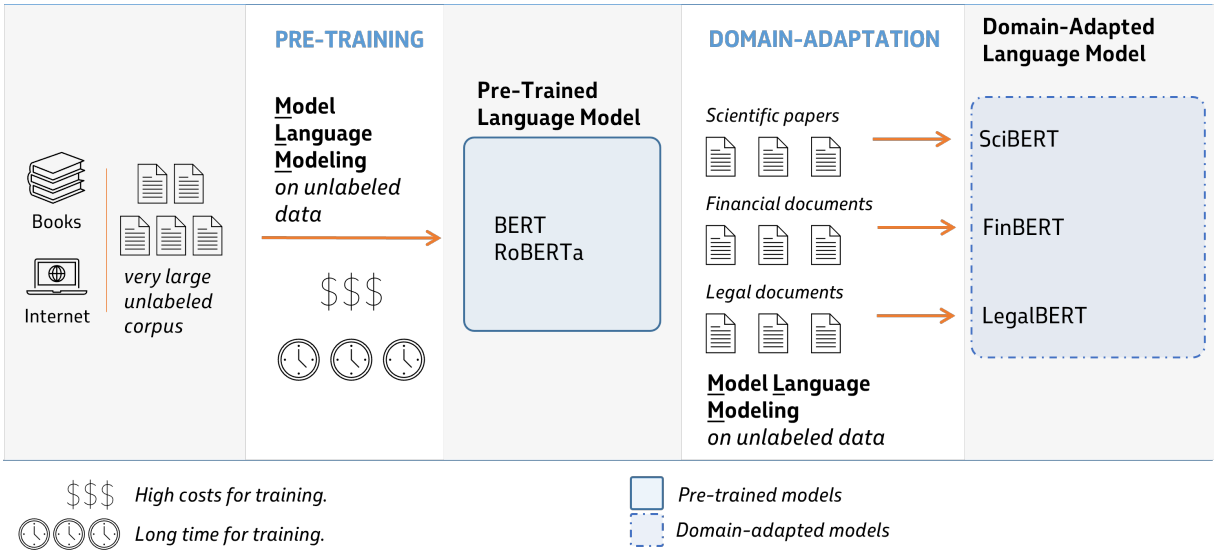


Figure 2: Domain Adaptation.

adaptation involves fine-tuning the language model on additional data before task-specific fine-tuning as done in the standard pipeline (Figure 1) and that the model is adjusted with the pre-training objective, meaning the domain adaptation also requires only unlabeled data.

**Domain definition.** Domain, in language model training, refers to a particular area of knowledge, expertise, or industry with its own specialized vocabulary, terminology, and conventions. In NLP, domain-specific models are trained on text corpora related to the target domain, allowing them to understand and generate relevant and appropriate text for the specific subject matter. Domains can vary widely, from general domains covering everyday language and common topics to specific domains focusing on specialized fields such as finance, medicine, law, or technology. A language model trained in the medical domain would be adept at handling medical terminology and concepts. In contrast, a model trained in the legal domain would be proficient in dealing with legal jargon and contexts. The domain can also be defined by the specific languages and style the model specializes in, like our BERTweet.BR, trained simultaneously for the domains of tweets and Portuguese language. Here, both the language and the Twitter platform are domains. Finally, the domain can also be defined by the tasks the model is being trained to perform, such as question-answering or natural language inference. Domain adaptation is a critical aspect of NLP, as it enables models to perform well in specialized areas and cater to the unique requirements of various industries.

## 2.4 Multi-Head Attention Mechanism

Transformer (VASWANI et al., 2017) was the first transduction model relying entirely on self-attention to compute representations of its input and output without using sequence-aligned RNNs or convolution. The self-attention mechanism enables the model to focus selectively on different parts of the input sequence during the encoding process. It accomplishes this by computing a weighted sum of the values of all the input tokens, where a learned similarity function between each token and every other token in the sequence determines the weights.

In particular, the "Scaled Dot-Product Attention", the attention mechanism proposed by Transformers as illustrated in Figure 3 (c) can be formulated as follows:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{2.1}$$

where $Q$, $K$, and $V$ are matrices composed of query, key, and value vectors, respectively. The origin of such naming can be found in search engines, where a user's query is matched against the internal engine's keys, and several values represent the result.

The goal is to have an attention mechanism in which any element in a sequence can attend to any other while still being efficient to compute. The *dot-product attention* takes as input a set of queries $Q \in \mathbb{R}^{T\times d_k}$, keys $K \in \mathbb{R}^{T\times d_k}$ and values $V \in \mathbb{R}^{T\times d_v}$ where $T$ is the sequence length, and $d_k$ and $d_v$ are the hidden dimensionality for queries/keys and values, respectively. The attention value from element $i$ to $j$ is based on its similarity of the query $Q_i$ and key $K_j$, using the dot product as the similarity metric. The matrix multiplication $QK^T$ performs the dot product for every possible pair of queries and keys, resulting in a matrix of the shape $T \times T$. Each row represents the attention logits for a specific element $i$ to all other elements in the sequence. On these, a softmax is applied and multiplied with the value vector to obtain a weighted mean (the weights being determined by the attention). Finally, the *scaled dot-product attention* introduces the scaling factor $1/\sqrt{d_k}$ to prevent the softmax function from giving values close to 1 for highly correlated vectors and values close to 0 for non-correlated vectors, making gradients more reasonable for back-propagation.

The attention mechanism from Transformers allows a network to attend over a sequence, but often, multiple aspects of a sequence element must be attended to. Therefore, it was extended to multiple heads, where multiple query-key-value triplets are passed

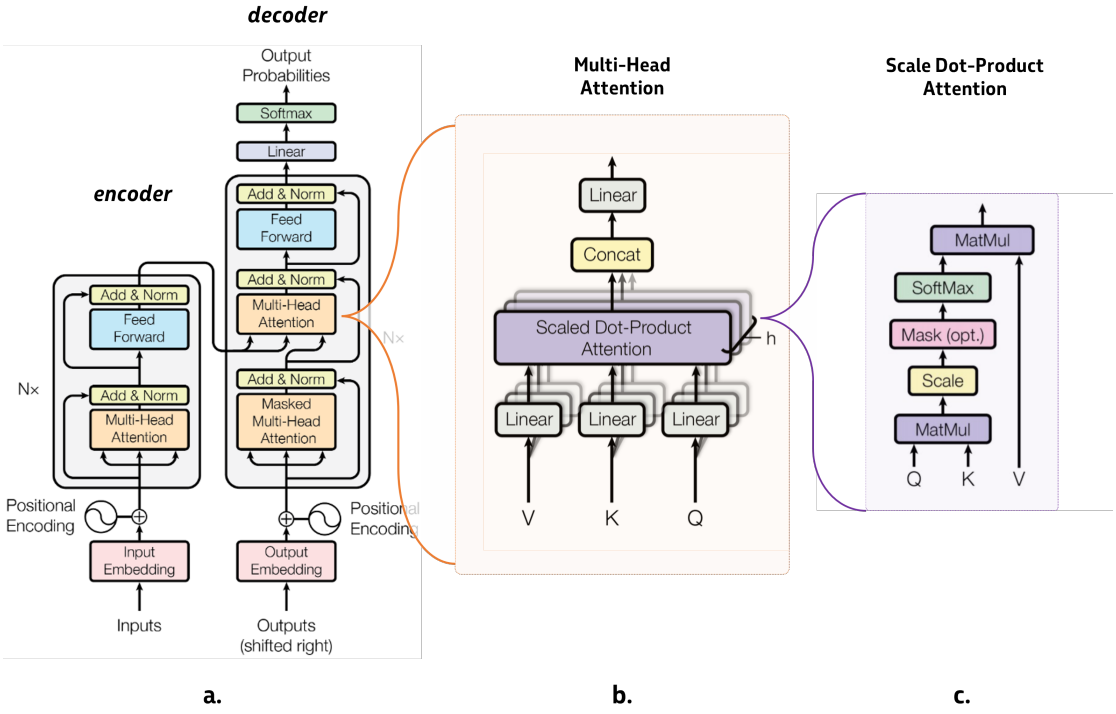


Figure 3: Transformers Architecture by Vaswani et al. (2017).

through the scaled dot product attention independently. The heads are then concatenated and combined with a final weight matrix. For the original architecture of Transformers, the authors employed $h = 8$ parallel attention layers or heads. The building blocks of the *multi-head attention* are depicted in Figure 3 (b), and its operation can be expressed mathematically as follows:

$$\text{Multihead}(Q,K,V) = \text{Concat}(\text{head}_1,...,\text{head}_h)W^O \tag{2.2}$$

$$\text{where head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{2.3}$$

The Transformer model has an encoder-decoder architecture, commonly used in many Neural Machine Translation (NMT) models (BAHDANAU; CHO; BENGIO, 2015). The *encoder* generates an attention-based representation to locate a specific piece of information from a significant context. In its original version, the model consists of a stack of $N = 6$ identical encoder modules, each containing two sub-modules, a *multi-head self-attention* layer and a fully connected feed-forward network as represented in Figure 3(a).

On the other hand, the function of the *decoder* is to retrieve information from the encoded representation. The architecture is quite similar to the *encoder*, except that the decoder contains two multi-head attention sub-modules instead of one in each identical

repeating module. The first multi-head attention sub-module is masked to prevent positions from attending to the future. The *decoder* also comprises a stack of $N = 6$ identical layers in the standard Transformers architecture.

Since their introduction in 2017, Transformers have rapidly become the state-of-the-art approach to tackle tasks in many domains such as Natural Language Processing, speech recognition, and computer vision. Each of its parts can be used independently and has been applied to different language models, depending on the task:

- **Encoder-only models:** Encoder-only models are a type of machine learning model that uses the encoder portion of a Transformer to learn the meaning of a sequence of text. The encoder in an encoder-only model is typically a stack of self-attention layers that takes an input sequence of text and produces a fixed-length vector representation of that text. This type of models are typically simpler and faster to train than decoder-only models and are more suitable for tasks that require an understanding of the input, such as sentence classification and Named Entity Recognition (NER) and extractive question answering. These models are often called *auto-encoding* models. Examples of this family of models include BERT (DEVLIN et al., 2019), RoBERTa (ZHUANG et al., 2021) and DistilBERT (SANH et al., 2019).

- **Decoder-only models:** Decoder-only models are machine learning models that use the decoder portion of a Transformer to generate text. The decoder takes a sequence of input tokens and produces a sequence of output tokens. This is done by iteratively attending to the input tokens and predicting the next output token. Decoder-only models are typically used for generative tasks, such as text generation, translation, and summarization. They are also used for tasks where the output text is more important than the input text, such as machine translation. These models are very effective for various generative tasks and are often used as the basis for more complex models. These models are often called *auto-regressive* models. The pre-training of decoder models usually revolves around predicting the next word in the sentence. GPT (RADFORD et al., 2018) is an example of this family of models.

- **Encoder-decoder models:** Encoder-decoder models (also known as *sequence-to-sequence models*) use both parts of the Transformer architecture: the encoder and the decoder. The encoder takes the input sequence and produces a fixed-length vector representation of the text. The decoder then takes this vector representation and produces a sequence of output tokens. Encoder-decoder models are typically used for tasks that involve understanding and generating text, such as machine

translation, text summarization, and question answering. They are also used for tasks where the input and output sequences are of different lengths, such as speech recognition and image captioning. Some examples of encoder-decoder models include the original Transformer model, BART (LEWIS et al., 2020) and T5 (RAFFEL et al., 2020).

## 2.5 BERT Language Model

BERT is basically a stack of Transformer-encoder layers designed by Devlin et al. (2019) and pre-trained using a combination of Masked Language Model (MLM) and Next Sentence Prediction (NSP) tasks on a large corpus comprising the BooksCorpus (ZHU et al., 2015) (800M words) and the English Wikipedia (2,500M words) corpora.

Devlin et al. (2019) released BERT models in two different sizes: $\text{BERT}_{\text{Base}}$ (110M parameters) and $\text{BERT}_{\text{Large}}$ (340M parameters). Both sizes have a larger number of encoder layers than the original Transformers — 12 for the base version and 24 for the large version. These also have larger feed-forward-networks (768 and 1024 hidden units, respectively) and more attention heads (12 and 16, respectively) than the default configuration in the reference implementation of the Transformer in the original paper (six encoder layers, 512 hidden units, and eight attention heads).

BERT receives as input a single sentence (for single-sequence tasks like sentiment analysis) or a pair of sentences (for sequence-pair tasks like question-answering) represented as a sequence of tokens. The WordPiece algorithm (WU et al., 2016), pre-trained with a vocabulary of 30,000 tokens, tokenizes the input. Every sequence of tokens is supplied with a special classification token, *[CLS]*, which can be used for classification tasks. For representing a pair of sentences ($S_1$,$S_2$), a unique token, *[SEP]*, is placed between them, and a learned embedding must be added to every token indicating whether it belongs to the sentence $S_1$ or $S_2$.

Just like the vanilla encoder of the transformer, BERT takes a sequence of words as input which keep flowing up the stack. Each layer applies self-attention, passes its results through a feed-forward network, and then hands it off to the next encoder as represented in Figure 4.

Devlin et al. (2019) also proposed a framework to fine-tune BERT pre-trained model using labeled data for a specific downstream task to leverage and refine the knowledge acquired during the pre-training process. In addition, they presented a *feature-based* ap-

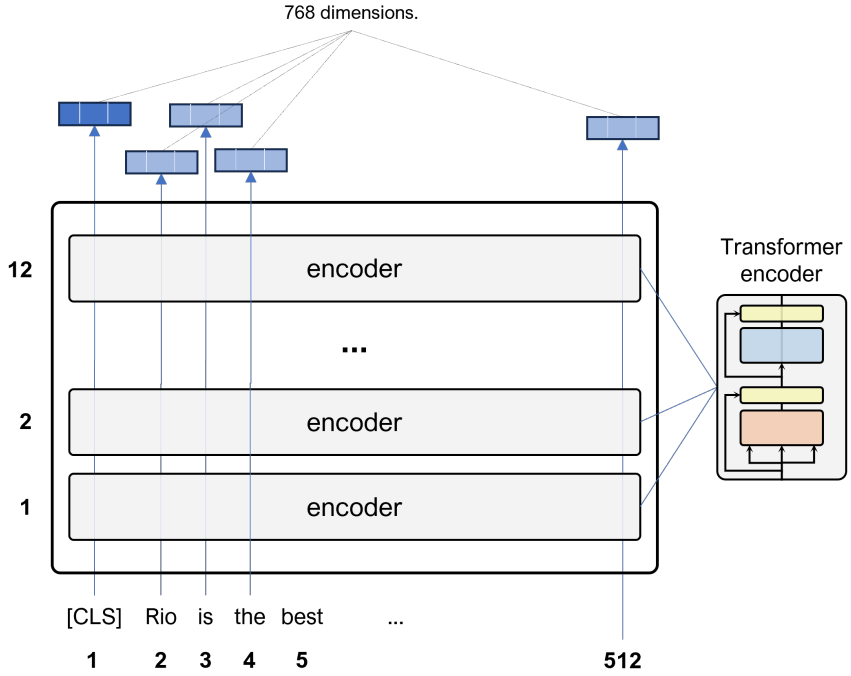


Figure 4: The $\text{BERT}_{\text{Base}}$ Flow.

proach, where fixed features are extracted from the pre-trained model to define embedding representations for tokens. Next, those embeddings can serve as examples to train any classification model.

## 2.6 BERTweet Language Model

Tweets exhibit a distinct and casual linguistic style, characterized by misspellings, slang, hashtags, emoticons, and URL sharing. Consequently, previously existing language models trained on standard text corpora, such as Wikipedia, may not align well with tweets. These models possess a conventional vocabulary that rarely matches the tweet vocabulary. In Transformer-based architectures utilizing subwords-based tokenizers, the noisy vocabulary of tweets can fragment words into numerous small pieces, ultimately distorting the original sentence meaning.

To address this issue, Nguyen, Vu, and Nguyen (2020) developed BERTweet, a publicly available, large-scale, pre-trained language model specifically for English tweets. Its architecture is based on $\text{BERT}_{\text{BASE}}$, featuring 12 layers and 12 heads in each layer. BERTweet incorporates a byte-level Byte Pair Encoding (BPE) tokenizer, which allows for better handling of out-of-vocabulary words frequently found in tweets. BERTweet was trained using the RoBERTa(ZHUANG et al., 2021) pre-training procedure, which,

in comparison to BERT, involves: extended training on more data with larger batches, eliminating the next sentence prediction target task, training on longer sequences, and dynamically altering the masking pattern applied to training data. The corpus utilized for BERTweet's training comprises 850M English tweets, containing 16 billion tokens and occupying approximately 80GB of storage space. Of these 850M tweets, 845M were collected from 01/2012 to 08/2019, while 5M pertain to the COVID-19 pandemic. BERTweet models are publicly accessible via Huggingface(WOLF et al., 2020).

BERTweet was evaluated on three NLP tasks for Twitter data in (NGUYEN; VU; NGUYEN, 2020): Part-Of-Speech (POS) tagging, Named-Entity Recognition (NER), and text classification. The results indicate that BERTweet surpasses previous state-of-the-art models in all three tasks.

## 2.7 Literature Review

Natural Language Processing has seen rapid advancements in recent years mainly due to the extensive use of transfer learning from deep contextualized pre-trained language models. In *sequential transfer learning*, the source and target tasks are different, and training is performed in two steps: *pre-training* and *adaptation* (RUDER, 2019). The general practice is to pre-train representations on a sizeable unlabeled text corpus (pre-training phase) and then adapt these representations to a supervised target task using labeled data (adaptation phase). To maximize the usefulness of sequential transfer learning, the pre-training task should produce a multipurpose representation of the language that might be useful not for one specific but for several target tasks. Intuitively, one basic approach to pursue such universal representations is to pre-train a language model (LM) on a large general-domain corpus extracted from well-written sources readily available on the Internet.

The embeddings of ELMo (PETERS et al., 2018), for example, are learned functions of the internal states of a deep bidirectional language model (biLM), which was pre-trained on a dataset of approximately 800M tokens of news crawl data (CHELBA et al., 2013). Later, in the adaptation phase, the original pre-trained LM is preserved as ELMo follows a *feature-based* strategy to provide pre-trained representations as the input to a separate downstream model. Similarly, ULMFiT (HOWARD; RUDER, 2018) was trained on the generic Wikitext-103 (MERITY et al., 2017) consisting of 28,595 preprocessed Wikipedia articles. In addition to the language model, the authors of ULMFiT also proposed an

effective transfer learning method inspired by computer vision that can be applied to any NLP task. Unlike ELMo, in the adaptation phase, ULMFiT follows a *fine-tuning* approach where the full language model is also updated. Both downstream and pre-trained language models are fine-tuned to learn task-specific features.

Following this paradigm, (VASWANI et al., 2017) introduced the Transformer architecture with a novelty self-attention mechanism that entirely dispenses recurrence and convolutions, denoting a great leap forward for NLP tasks with long-term dependencies. After that, influential models have been proposed, including the *auto-regressive* GPT (RADFORD et al., 2018) and *auto-encoding* BERT (DEVLIN et al., 2019), either building upon the decoder or the encoder component of the original Transformers work (VASWANI et al., 2017). GPT implements an *unidirectional* 12-layer *decoder-only* language model pre-trained for 100 epochs on BooksCorpus (ZHU et al., 2015), a dataset of 7,000 books. BERT (DEVLIN et al., 2019) overcomes the gaps of unidirectionality by introducing a multi-layer *bidirectional* and *encoder-only* Transformer model. The model advanced the state-of-the-art for both sentence-level and token-level NLP tasks after being pre-trained on BooksCorpus (800M words) and English Wikipedia (2,500M words) using two unsupervised tasks: the masked language model task (MLM) and next sentence prediction (NSP). One of the relevant variations of BERT is RoBERTa (ZHUANG et al., 2021), which achieved a significant performance gain by being trained on a corpus considerably larger than BERT (DEVLIN et al., 2019), discarding the next sentence prediction task (NSP) as well as a longer pre-training phase with larger batches.

Parallel work on multilingual understanding extends these systems to more languages, thus enabling the use of language models beyond English. For example, a multilingual version (mBERT), pre-trained on the 100 largest languages in Wikipedia, was also released along with the original BERT (DEVLIN et al., 2019). Despite all the contributions of a multilingual model, authors of BERT acknowledged that the "multilingual model is somewhat worse than a single-language model". Then, specialized monolingual models have been published, outperforming previous multilingual benchmarks as seen for French (LE et al., 2020; MARTIN et al., 2020), Vietnamese (NGUYEN; NGUYEN, 2020), Spanish (CAÑETE et al., 2020), German (CHAN; SCHWETER; MÖLLER, 2020) among others. For Portuguese, (SOUZA; NOGUEIRA; LOTUFO, 2020) replicated BERT's architecture and pre-training procedures to yield BERTimbau, a language model for Brazilian Portuguese that achieved state-of-the-art performances on three downstream NLP tasks after being pre-trained on brWaC (WAGNER FILHO et al., 2018), a large and diverse corpus of web pages.

All aforementioned language models were pre-trained from large volumes of conventional text corpora from books, Wikipedia articles, and news sites. However, despite being able to achieve robust performance across different downstream tasks in various languages, when applied to specialized domains, such as biomedical, scientific, or legal texts, the generic-domain representations have shown to under-perform in many target domain tasks as shown in works like (LEE et al., 2020; BELTAGY; LO; COHAN, 2019; CHALKIDIS et al., 2020). As demonstrated by (GURURANGAN et al., 2020), the more distinct the target and source domains are, the greater the degradation of generic models (source) when used in specific domains (target), and the higher will be the potential of techniques for model language domain adaptation. This is the case of the microblog Twitter where users communicate with each other informally, using typical expressions of social networks slang, and many times with lexicon-syntactic errors or adding special tokens such as hashtags, user mentions, and emojis. Therefore, there is an apparent mismatch between the domains of Twitter (target domain) and Wikipedia, books, and news articles (source domain) – traditionally used for pre-training generic-domain language models.

To fill that gap, (NGUYEN; VU; NGUYEN, 2020) proposed BERTweet, a domain-specific language model trained from scratch on English tweets. Studies have demonstrated that BERTweet outperforms baselines such as RoBERTa$_{\text{base}}$ (ZHUANG et al., 2021) and XLM-R$_{\text{base}}$ (CONNEAU et al., 2020) on three NLP tasks on tweets, thus demonstrating the effectiveness of large-scale language model specially adapted for the specific domain of tweets. Following similar procedures, (POLIGNANO et al., 2019), (GONZÁLEZ; HURTADO; PLA, 2020; HUERTAS-TATO; MARTIN; CAMACHO, 2022), and (ABDELALI et al., 2021) introduced AlBERTo, TWilbert, Bertuit, and QARiB, respectively – pre-trained language models trained from scratch on massive corpora of Italian, Spanish (two models), and Arabic tweets. Later, XLM-T (BARBIERI; ESPINOSA-ANKE; CAMACHO-COLLADOS, 2022) was proposed following the DAPT procedure (GURURANGAN et al., 2020; BARBIERI; CAMACHO-COLLADOS, et al., 2020) to adapt XLM-R (CONNEAU et al., 2020) to the Twitter domain, creating a multilingual model from a corpus containing 198M tweets written in the 30 most represented languages in Twitter. (GUO et al., 2021) continued pre-training the generic-domain model CamemBERT (MARTIN et al., 2020) to provide BERTweetFR, a domain-specific model for French tweets. (KOTO; LAU; BALDWIN, 2021) trained five different versions of the IndoBERTweet model to compare the following approaches: pre-training from scratch based in IndoBERT (KOTO; RAHIMI, et al., 2020) model along with other four adapted models resulting from the domain-adaptive pre-

training procedure with distinct vocabulary adaptation strategies. The study reveals that it is feasible to adapt an off-the-shelf pre-trained model very efficiently and obtain better average performance than training from scratch.

Portuguese, one of the most active languages on Twitter, has been remarkably neglected by this tendency of pre-training massive language-specific LM for tweets. That is where the scope of this work applies to. We propose a replication of BERTweet aiming to advance the research of NLP in Portuguese on top of a language model capable of accurately inheriting leanings, nuances, and biases from Portuguese tweets, which can later be usefully applied to any downstream task in Twitter and informal scenarios.

# 3 Building BERTweet.BR

This section outlines the architecture, dataset, and optimization setup adopted to pre-train the *BERTweet.BR* language model.

## 3.1 Architecture

*BERTweet.BR* follows the same architecture and pre-training procedure as the BERTweet model (NGUYEN; VU; NGUYEN, 2020), which replicates RoBERTa and pre-trains the model based on Masked Language Model objective only. MLM was originally introduced as a "Cloze task" by Taylor (1953) and enforces bidirectional learning from text by masking (hiding) a word in a sentence at random and forcing the training model to bidirectionally use the words on either side of the covered word to predict the masked word. As made by Devlin et al. (2019), we let the training data generator chooses 15% of the token positions at random for prediction.

We employed a multi-layer bidirectional Transformer architecture using the same configuration as the base version of BERT (DEVLIN et al., 2019) with 12 layers, 768 hidden dimensions, and 12 attention heads, adding up to a total of approximately 135M parameters.

## 3.2 Tokenizer

We adapted the normalization step of the original BERTweet tokenizer (BERTweetTokenizer) to deal with demojizer for Portuguese as the default language of the method demojize from emoji library[1] is set to English (Figure 5)[2].

[1] https://pypi.org/project/emoji/

[2] In a loose translation to English, the three tweets provided as examples in Figure 5 would be: i) *"Looking for a love good for me...I will look for it and I go to the end"* (An excerpt from a song called *Segredos* https://www.letras.mus.br/frejat/64374/ by the Brazilian artist Roberto Frejat https://pt.wikipedia.org/wiki/Frejat; ii) *What a match yesterday @cristiano*; iii) *Demojizer for*

| Input Tweet | Normalized Tweet |
|---|---|
| Procuro um amor , que seja bom pra mim ... vou procurar , eu vou até o fim 🎶 | Procuro um amor , que seja bom pra mim ... vou procurar , eu vou até o fim **:nota_musical:** |
| Que jogo ontem @cristiano 🙏 | Que jogo ontem **@USER :mãos_juntas:** |
| Demojizer para Python é 👍 e está disponível em https://pypi.org/project/emoji/ | Demojizer para Python é **:polegar_para_cima:** e está disponível em **HTTPURL** |

Figure 5: Examples of emoji library method demojize for Portuguese Language (bold) and normalization process to convert user mentions and web or url links into the special tokens (red).

The *BERTweet.BR* tokenizer was trained to have a 64K token vocabulary and also used fastBPE (SENNRICH; HADDOW; BIRCH, 2016) to segment words into subword units. Subword tokenization algorithms like fastBPE rely on the principle that frequently used words should not be split into smaller subwords, but rare words should be decomposed into meaningful subwords. Consequently, it allows the model to have a reasonable vocabulary size while being able to learn meaningful context-independent representations. In addition, subword tokenization enables the model to process words it has never seen before, by decomposing them into known subwords. In particular, BPE tokenizer family doesn't aim at being linguistically correct, but rather at being a good compromise between speed, correctness, and coverage.

Given the short-length nature of tweets, we set the maximum sequence length of the tokenizer to 128 tokens, meaning sentences longer than that are truncated before passing to the model.

## 3.3 Pre-training dataset

We downloaded tweets from the collection grabbed by the Archive Team[3], containing tweets streamed from the general Twitter stream from 2004 to 2020. We filtered Portuguese tweets by setting the field *lang* to *pt* ("lang" = "pt"). We tokenized sentences using the *TweetTokenizer* class from the NLTK library (BIRD, 2006) and used the *demojize* method of the emoji library[4] to translate emotion icons into text strings in Portuguese. We also converted user mentions and web or url links into the special tokens @USER and HTTPURL, respectively. The corpus preprocessing step is illustrated in Figure 5 with

*Python is great and is available at https://pypi.org/project/emoji/*

[3] https://archive.org/details/twitterstream

[4] https://pypi.org/project/emoji/

| Raw Tweet | Processed Tweet (with masked tokens) |
|---|---|
| Procuro um amor , que seja bom pra mim ... vou procurar , eu vou até o fim 🎶 | Procuro um amor , que seja bom **<mask>** mim ... vou procurar , **<mask>** vou até o **<mask>** :@@ music@@ al_@@ no@@ te: |
| Que jogo ontem @cristiano 🙏 | Que jogo ontem **@USER** :@@ fol@@ de@@ **<mask>** _@@ hand@@ s: |
| Demojizer para Python é 👍 e está disponível em https://pypi.org/project/emoji/ | D@@ emo@@ **<mask>** izer para Py@@ thon é :@@ thum@@ b@@ s_@@ up@@ **<mask>** e está disponível em **HTTPURL** |

Figure 6: Examples of tweets after normalization and tokenization. For any given input sequence, 15% of the tokens are chosen for possible replacement with <mask> token.

some examples of raw tweets after BERTweet.BR normalization.

Finally, we obtained a corpus of 100M unique tweets split into 90/10 percent proportion for training and test sets, respectively. The resulting dataset is approximately 9 GB large.

## 3.4 Pre-training

*BERTweet.BR* was pre-trained on the Masked Language Modeling (MLM) task. In any given input sequence, 15% of the tokens were chosen for possible replacement so that the model was subsequently trained to predict tokens replaced by <MASK> using cross-entropy loss (Figure 6)[5].

The cross-entropy loss is used to measure the distance between the distribution of probability of the masked tokens and the distribution of probability of the model's predictions. The model is then adjusted to minimize the cross-entropy loss, which means that the model is learning to predict the masked tokens with the highest possible probability. The cross-entropy loss is a powerful cost function that can be used to train a variety of language models. It is an efficient and easy-to-optimize cost function, and it is able to learn robust language representations.

The mathematical formulation of the cross-entropy loss used in the MLM task is as follows:

[5]In a loose translation to English, the three tweets provided as examples in Figure 6 would be: i) *"Looking for a love good for me...I will look for it and I go to the end"* (An excerpt from a song called *Segredos* https://www.letras.mus.br/frejat/64374/ by the Brazilian artist Roberto Frejat https://pt.wikipedia.org/wiki/Frejat; ii) *What a match yesterday @cristiano*; iii) *Demojizer for Python is great and is available at https://pypi.org/project/emoji/*

$$L = -\sum_{i=1}^{N} y_i \log \hat{y}_i \tag{3.1}$$

where:

- $y_i$ is the true label of token $i$
- $\hat{y}_i$ is the model's prediction for token $i$
- $N$ is the number of tokens in the input

We leveraged the RoBERTa implementation of the Transformers library (WOLF et al., 2020) and followed the language model training script *run_mlm.py*[6] with the PyTorch (PASZKE et al., 2019) distributed package in a Linux CentOS 7 server.

We optimized the model using Adam with a batch size of 96 across four GPUs (NVIDIA Tesla V100-SXM2-32GB) and a peak learning rate of 0.0001. We pre-trained *BERTweet.BR* for 30 epochs in about three weeks (the first 50K training steps were used for warming up the learning rate). The model was evaluated every 50K steps during pre-training as illustrated in Figure 7. We then named *BERTweet.BR* the very last checkpoint as that was the best-performing version after approximately 7M training steps.

[6] https://github.com/huggingface/transformers/blob/main/examples/pytorch/language-modeling/run_mlm.py

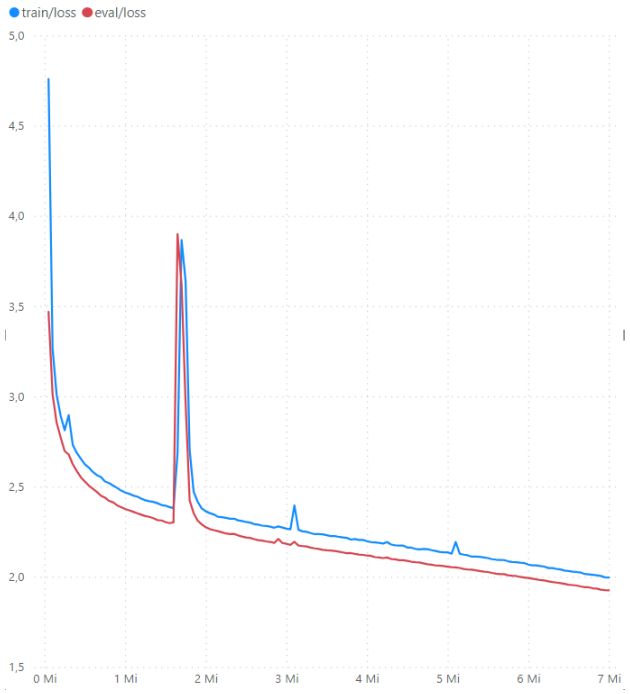


Figure 7: Training and evaluation loss progress of BERTweet.BR model, trained from scratch for 30 epochs.

# 4 Experiments Design

The experiments conducted in this section aim at answering the research questions introduced in Section 1. To this end, we benchmark the performance of *BERTweet.BR* in the sentiment analysis task in tweets using a set of different pipelines. The evaluation includes static to contextualized embeddings, ranging from language specific to multilingual models and Twitter-specialized options as shown in Figure 8 where we summarize all the experiments. The former relies on fastText, the best-performing static option in this context, as demonstrated by Vianna et al. (2023). The latter leverages transformer-based architectures with a comprehensive list of four models: BERTimbau, mBERT (multilingual version of BERT), XLM-R (multilingual version of RoBERTa), and XLM-T, an adaptation of XLM-R for the Twitter domain.

When applicable, strategies for applying pre-trained language representations to downstream tasks encompass a *feature-based* and a *fine-tuning* procedure. Therefore, as detailed in subsection 4.2, for all transformer-based models, we conduct both strategies, while for the pre-trained word vector fastText, to which *fine-tuning* is not applicable, we only employed the standard *feature-based* strategy.

We compare *BERTweet.BR* with up to three different benchmarks of each model as shown in Table 1. First and foremost, we conducted the experiments on top of the *off-the-shelf* publicly available pre-trained weights for all the benchmark language models. These are the original pre-trained versions released by their authors, called here as *original checkpoints* (column *Original*).

Then, we employ the continuous pre-training strategy to account for adapting the original checkpoints to the tweets domain as per the research question *RQ1*.

To this end, we adopted the *domain-adaptive pre-training* (DAPT) and *task-adaptive pre-training* (TAPT) procedures of (GURURANGAN et al., 2020). Then, concerning *RQ2*, the resulting models were used to perform the same set of experiments of sentiment analysis using the *feature-based* and *fine-tuning* strategies again, now over these two ad-

Table 1: Summary of experiment baselines of each model language and domain adaptation procedures explored in this work.

| | *Original* | *TAPT* | *DAPT* |
|---|---|---|---|
| *fastText* | ✓ | | |
| *mBERT* | ✓ | ✓ | ✓ |
| *XLM-R* | ✓ | ✓ | ✓ |
| *XLM-T* | ✓ | | |
| *BERTimbau* | ✓ | ✓ | ✓ |
| *BERTweet.BR* | ✓ | | |

ditional variations of language models (DAPT and TAPT). In summary, we compare *BERTweet.BR* to a total of 11 different versions of language models (Table 1). As we apply both *feature-based* and *fine-tuning* approaches to contextualized LM and only the first of the methods to fastText, we reach 23 experiments in this study.

## 4.1 Sentiment Analysis Experiment

Sentiment analysis is a supervised sequence classification task in which we inspect a given text and identify the prevailing opinion within it, typically to determine a writer's attitude as *positive*, *negative*, or *neutral*. In this work, given a specific tweet, the goal is to determine whether it reveals a positive or negative opinion (binary mode) or expresses a neutral, positive, or negative message (multiclass mode).

Sentiment analysis is a crucial task in natural language processing that enables governments, organizations, and other entities to gain valuable insights into people's attitudes and perceptions toward various topics. This downstream task's significance makes it a natural choice when evaluating the benefits of language models.

The application of sentiment analysis has far-reaching implications across various domains, including marketing, politics, and healthcare. For instance, sentiment analysis can assist marketers in understanding consumer preferences and feedback, helping them to tailor their campaigns and improve customer engagement. Similarly, sentiment analysis can be leveraged in the political domain to gauge public opinion on governmental policies and actions and monitor the spread of propaganda and misinformation. Also, during crises such as pandemics or natural disasters, sentiment analysis can provide real-time insights into the public's sentiments, allowing policymakers to gauge public reactions and implement measures accordingly.

Table 2: Target datasets are grouped by the number of classes and ordered by the number of tweets. Length is computed before normalization by splitting tweets in white space.

| | | tweets | positive | | negative | | neutral | | duplicated | | min | max | avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *binary* | *covidbr* | 600 | 300 | 50.0% | 300 | 50.0% | - | - | 2 | 0.3% | 5 | 53 | 28.0 |
| | *sentbr* | 7769 | 4773 | 61.4% | 2996 | 38.6% | - | - | 9 | 0.1% | 1 | 53 | 11.9 |
| | *fiat* | 8827 | 4437 | 50.3% | 4390 | 49.7% | - | - | 0 | 0.0% | 1 | 58 | 16.7 |
| | | **17196** | **9510** | **55.3%** | **7686** | **44.7%** | - | - | **11** | **0.1%** | **1** | **58** | |
| *multiclass* | *narrpt* | 772 | 297 | 38.5% | 213 | 27.6% | 262 | 33.9% | 8 | 1.0% | 2 | 31 | 14.05 |
| | *mining* | 2018 | 166 | 8.2% | 1299 | 64.4% | 553 | 27.4% | 59 | 2.9% | 2 | 31 | 14.81 |
| | *compbr* | 2281 | 197 | 8.6% | 407 | 17.8% | 1677 | 73.5% | 177 | 7.8% | 1 | 47 | 16.44 |
| | *tweemg* | 8199 | 3300 | 40.2% | 2446 | 29.8% | 2453 | 29.9% | 2424 | 29.6% | 1 | 32 | 16.14 |
| | *unilex* | 12665 | 3715 | 29.3% | 4197 | 33.1% | 4753 | 37.5% | 0 | 0.0% | 1 | 62 | 14.30 |
| | | **25935** | **7675** | **29.6%** | **8562** | **33.0%** | **9698** | **37.4%** | **2668** | **10.3%** | **1** | **62** | |
| | | **43131** | **17185** | **39.84%** | **16248** | **37.68%** | **9698** | **22.48%** | **2668** | **6.18%** | **1** | **62** | |

**Datasets.** We retrieved a plural set of eight human-annotated datasets from various domains for sentiment analysis. More details about each dataset of the collection can be found in Appendix A. As described in Table 2, they are also varied in size and number of classes, with three being binary datasets containing *negative* and *positive* classes. In contrast, five others are ternary, having the additional *neutral* label. For each dataset, Table 2 shows the total number of tweets (column *tweets*) and the number and percentage of rows labeled with each of the three classes (columns *positive*, *negative*, and *neutral*). Datasets are grouped by the number of classes (rows *binary* and *multiclass*) and ordered by the number of tweets. Also, it is provided the number and percentage of duplicated tweets (retweets – column *duplicated*) along with the minimum, maximum and average length of tweets of each dataset (columns *min*, *max*, and *avg*, respectively). Length is computed before normalization by splitting tweets in white space. These labeled datasets can be downloaded from `https://bityli.com/RvhFax`. We also made the collection available in the transformers datasets library[1].

## 4.2 Evaluation

As proposed in research question *RQ2*, we apply *feature-based* and *fine-tuning* approaches to sentiment analysis to assess the reliability of *BERTweet.BR* language model. We evaluate the performance of models on each dataset following a 10-fold stratified cross-validation strategy in which the sample distribution for each class is preserved in all folds. The results are expressed by the average weighted F1-score across the ten folds on

[1] `https://huggingface.co/melll-uff/`

the validation set. Datasets are shuffled before splitting. We ensure experiments across models are comparable even in the fold level as a *seed* is set to control the randomness.

### 4.2.1 Feature-based Approach

In the feature-based approach, we use pre-trained representations as input features for a downstream task without changing the original language models. We pass sentences through the model that outputs embeddings which are then used to fit a classifier. In this work, we adopted the Logistic Regression (LR) classification algorithm due to its good performance in sentiment analysis as demonstrated by Vianna et al. (2023). Also for its simplicity, efficiency, and interpretability. We used the LR implementation from scikit-learn (PEDREGOSA et al., 2011)[2].

Static embeddings from fastText (MIKOLOV; GRAVE, et al., 2018) are obtained for whole sentences after normalization. Given a *tweet*, we get a single vector representation with *get_sentence_vector* method on top of the Portuguese pre-trained word vectors[3] shared by the fastText team with the dimension of 300.

When applied to transformer-based models, the *feature-based* approach is comprised of extracting the contextualized embeddings from one or more layers without fine-tuning any parameters. In this work, we sum the outputs of the last four hidden layers as described by (DEVLIN et al., 2019) when applying BERT with the *feature-based* approach. We pass the normalized *tweets* as inputs and focus only on the first position of the *hidden_states* outputs generated by the pre-trained models, which is the output for the special token used as the aggregate sequence for sentence classification tasks. The resulting contextualized embeddings are a vector of the same size as the pre-trained model's *hidden_size*. All transformer-based models used in this work have *hidden_size* of 768, including *BERTweet.BR*.

The experiments described in this section are aimed at inquiries posed in *RQ2.1*.

### 4.2.2 Fine-tuning Approach

Following a fine-tuning approach is the standard pipeline when employing transformer-based models into target downstream tasks. Fine-tuning is the training done after a language model has been pre-trained, and, as opposed to feature-based, it involves ad-

[2] https://scikit-learn.org/

[3] https://dl.fbaipublicfiles.com/fasttext/vectors-crawl/cc.pt.300.bin.gz

justing the pre-trained model's weights on a specific task. Each downstream task has its appropriate fine-tuned models with a task-specific head but initialized with the same pre-trained parameters of language models.

Concerning *RQ2.2*, we employ the framework of `transformers` library (WOLF et al., 2020) to fine-tune our *BERTweet.BR* and other benchmarking models for the downstream task with each of the eight datasets. To this end, for each dataset and each model, we instantiate a `AutoModelForSequenceClassification`[4] object which loads a task-specific architecture formed by the original transformer model initialized with the pre-trained weights and an additional untrained sequence classification head on top. Then, using the labeled data from downstream datasets shown in Table 2, all parameters of these combined models are adjusted for sentiment analysis for three epochs. We use *AdamW* (LOSHCHILOV; HUTTER, 2019) with a fixed learning rate of 5e−5 and a batch size of 32 tweets, reporting results of the last model checkpoint on the validation set.

## 4.3 Domain Adaptation

To tackle the issues raised in research question *RQ1*, we also conduct the experiments described in Section 4.1 on top of adjusted models, resulting from a second phase of the pre-training procedure from their original checkpoints (continued pre-training). The objective of these experiments is to adapt the three originally general-domain language models — mBERT, XLM-R, and BERTimbau — to the domain of tweets, the object of this study, and then also compare our *BERTweet.BR* to these additional language models fine-tuned to tweets in Portuguese. To this end, we follow both *domain-adaptive pre-training* (DAPT) and *task-adaptive pre-training* (TAPT) procedures of (GURURANGAN et al., 2020). In this sense, we seek to verify if only employing a lower-resource pre-training approach through domain adaptation of existing models would produce better results than pre-training a new language model from scratch.

### 4.3.1 Domain adaptive pre-training (DAPT)

The procedure we follow for *domain-adaptive pre-training (DAPT)* is straightforward — we continue pre-training the three target language models on the same unlabeled corpus of

[4] https://huggingface.co/docs/transformers/model_doc/auto#transformers.AutoModelForSequenceClassification

tweets we pre-trained *BERTweet.BR*. We also use the same training script *run_mlm.py*[5] and set of hyper-parameters, running on the same hardware configuration, but this time for three epochs. We keep the original tokenizers and their vocabularies. This second phase of pre-training results in three domain-adapted LMs, which are then used in the benchmark experiments of Section 4.1 to answer *RQ1.1*. In this work, we refer to these models as <original-model-name>-DAPT as shown in Figure 8.

### 4.3.2 Task-adaptive pre-training (TAPT)

*Task-adaptive pre-training (TAPT)* refers to pre-training an LM on an unlabeled training set initially curated for a supervised given task. TAPT is much less expensive because it uses a far smaller corpus but can still produce competitive results for domain adaptation (GURURANGAN et al., 2020).

Instead of continuing pre-training LMs on each of the eight datasets individually, our approach to TAPT considers the entire collection as one augmented dataset. It employs a *LOO (Leave One dataset Out)* strategy for training. Specifically, we take each dataset once as the target dataset while the tweets from the remaining seven datasets are combined to tune the language model. Next, to answer *RQ1.2*, the resulting model is used to evaluate the performance in the target dataset on the experiments described in section 4.1. Therefore, the procedure repeats eight times for each language model candidate for TAPT: mBERT, XLM-R, and BERTimbau. Note that for the pre-training phase, which is a self-supervised task, we omit the labels of the combined dataset. In contrast, labels are preserved for the supervised sentiment analysis task on the target dataset. To this end, we employ the `transformers` library (WOLF et al., 2020) and customize the training script *run_mlm_no_trainer.py*[6] with its default hyper-parameter values to continue pre-training models using a masked language modeling loss for 20 epochs. Similarly to DAPT, we also keep the original tokenizers and their vocabularies. Finally, the last checkpoint produced is then used in the benchmark experiments. In this work, we refer to these models as <original-model-name>-TAPT as shown in Figure 8.

[5] https://github.com/huggingface/transformers/blob/main/examples/pytorch/language-modeling/run_mlm.py

[6] https://github.com/huggingface/transformers/blob/main/examples/pytorch/language-modeling/run_mlm_no_trainer.py

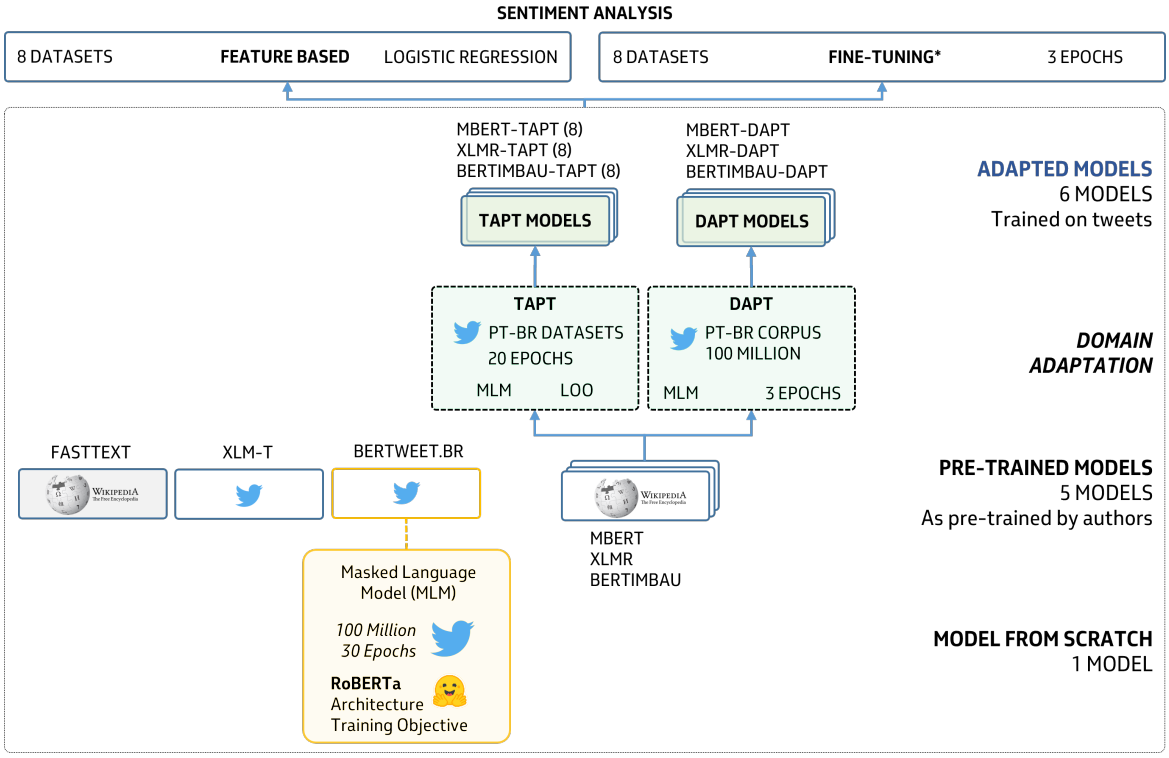


Figure 8: The summary of experiments. *BERTweet.BR* was pre-trained from scratch on the Masked Language Modeling task using RoBERTa architecture over 100 million tweets for 30 epochs.

# 5 Experimental Results

This chapter presents the experimental results obtained from benchmarking *BERTweet.BR* in the sentiment analysis task using the range of different pipelines described in Chapter 4. We report and discuss the results with which we answer to inquiries about both research questions *RQ1* and *RQ2*.

Tables 3, 4, and 5 present results for our *BERTweet.BR* as well as model baselines regarding both off-the-shelf checkpoints and domain-adapted language models resulting from *task-adaptive pre-training (TAPT)* and *domain-adaptive pre-training (DAPT)*, respectively. In all result tables we report the weighted f1-score in the test set for each of the eight datasets for sentiment analysis. The models are evaluated in a stratified 10-fold strategy using both *feature-based (fb)* and *fine-tuning (ft)* approaches. In bold, it is indicated the best performance recorded for each dataset and separated by approach. Then, for each group, the best performance previous to *BERTweet.BR* is also reported. After the scores of *BERTweet.BR* in parentheses, it is reported the percentage gain (or loss) of *BERTweet.BR* results compared to the previous best results. Here, the goal is to highlight the *BERTweet.BR* performance in comparison to the scenario where there wasn't pre-trained specific models to tweets in Portuguese. The best previous performance is then the highest score reported considering all benchmark models excluding *BERTweet.BR*. Finally, the rightmost columns (wins and wins*) indicate the number of datasets each language model got the best f1-score; respectively, in the scenarios where our *BERTweet.BR* model is considered or not. Again, in the column wins* it is indicated the number of datasets any given model obtained the best result in the scenario excluding *BERTweet.BR*. Note that for all three tables the fine-tuning approach gives the best performance for all datasets.

*BERTweet.BR* reported the absolute best F1-score on 36 of the 48 experiments while matching the previous best performances on two scenarios. Our new model, trained from scratch, achieved the best scores in approximately 80% of all experiments. Taking into consideration the overall results by dataset as shown in Table 6, the *fine-tuning* approach

Table 3: The weighted f1-score in the test set for each of the eight datasets for sentiment analysis on top of the original checkpoints of the models.

| | | | covidbr | sentbr | fiat | narrpt | mining | compbr | tweemg | unilex | wins | wins* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ORIGINAL CHECKPOINTS | *feature-based* | *fasttext* | 0.770 | 0.828 | 0.815 | 0.776 | 0.776 | 0.781 | 0.945 | 0.661 | **0** | 0 |
| | | *bertimbau* | **0.805** | 0.842 | 0.826 | 0.804 | 0.784 | 0.821 | **0.954** | 0.668 | **2** | 2 |
| | | *mbert* | 0.716 | 0.738 | 0.774 | 0.651 | 0.706 | 0.752 | 0.927 | 0.602 | **0** | 0 |
| | | *xlmr* | 0.771 | 0.803 | 0.807 | 0.738 | 0.777 | 0.799 | 0.942 | 0.649 | **0** | 0 |
| | | *xlmt* | 0.768 | 0.851 | 0.826 | 0.838 | 0.802 | 0.834 | 0.949 | 0.672 | **0** | 6 |
| | | ***previous*** | ***0.805*** | *0.851* | *0.826* | *0.838* | *0.802* | *0.834* | ***0.954*** | *0.672* | - | - |
| | | ***BERTweet.BR*** | 0.778 *(-3.4)* | **0.881** *(3.5)* | **0.827** *(0.1)* | **0.843** *(0.5)* | **0.824** *(2.7)* | **0.848** ***(1.7)*** | **0.954** *(0.0)* | **0.679** *(1.1)* | **7** | - |
| | *fine-tuning* | *bertimbau* | **0.864** | 0.877 | 0.891 | 0.855 | 0.828 | 0.855 | 0.972 | 0.732 | **1** | 5 |
| | | *mbert* | 0.786 | 0.841 | 0.878 | 0.688 | 0.759 | 0.773 | 0.972 | 0.703 | **0** | 0 |
| | | *xlmr* | 0.768 | 0.825 | 0.862 | 0.742 | 0.732 | 0.742 | 0.967 | 0.715 | **0** | 0 |
| | | *xlmt* | 0.806 | 0.882 | 0.888 | 0.873 | 0.829 | 0.842 | 0.971 | 0.723 | **0** | 3 |
| | | ***previous*** | ***0.864*** | *0.882* | *0.891* | *0.873* | *0.829* | *0.855* | *0.972* | *0.732* | - | - |
| | | ***BERTweet.BR*** | 0.828 *(-4.2)* | **0.902** *(2.2)* | **0.892** *(0.1)* | **0.909** *(4.2)* | **0.857***(3.4)* | **0.874** *(2.2)* | **0.973** *(0.1)* | **0.733** *(0.2)* | **7** | - |
| | | ***summary*** | bertimbau | bertweetbr | bertweetbr | bertweetbr | bertweetbr | bertweetbr | bertweetbr | bertweetbr | - | - |
| | | | 0.864 | 0.902 | 0.892 | 0.909 | 0.857 | 0.874 | 0.973 | 0.733 | - | - |

Table 4: The weighted f1-score in the test set for each of the eight datasets for sentiment analysis on top of adapted models resulting from *task adaptive pre-trained (TAPT)* procedure.

| | | | covidbr | sentbr | fiat | narrpt | mining | compbr | tweemg | unilex | wins | wins* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TAPT | *feature-based* | *bertimbau* | ↓ 0.791 | ↓ 0.824 | ↓ 0.825 | ↓ 0.797 | 0.789 | 0.826 | ↓ 0.950 | ↓ 0.654 | **0** | 0 |
| | | *mbert* | 0.743 | 0.767 | 0.792 | 0.690 | 0.721 | 0.768 | 0.932 | 0.634 | **0** | 0 |
| | | *xlmr* | **0.805** | 0.819 | 0.825 | 0.787 | 0.787 | 0.814 | 0.952 | 0.657 | **1** | 2 |
| | | *xlmt* | 0.768 | 0.851 | 0.826 | 0.838 | 0.802 | 0.834 | 0.949 | 0.672 | **0** | 6 |
| | | ***previous*** | ***0.805*** | *0.851* | *0.826* | *0.838* | *0.802* | *0.834* | ↓ *0.952* | *0.672* | - | - |
| | | ***BERTweet.BR*** | 0.778 *(-3.4)* | **0.881** *(3.5)* | **0.827** *(0.1)* | **0.843** *(0.5)* | **0.824** *(4.5)* | **0.848** *(1.7)* | **0.954** *(0.2)* | **0.679** *(1.1)* | **7** | |
| | *fine-tuning* | *bertimbau* | ↓ **0.853** | ↓ 0.871 | ↓ 0.889 | 0.872 | 0.834 | 0.856 | **0.974** | ↓ 0.728 | **2** | 6 |
| | | *mbert* | ↓ 0.781 | 0.843 | 0.885 | 0.763 | 0.801 | 0.815 | ↓ 0.969 | ↓ 0.702 | **0** | 0 |
| | | *xlmr* | 0.799 | 0.856 | 0.884 | 0.786 | 0.792 | 0.810 | 0.971 | ↓ 0.698 | **0** | 0 |
| | | *xlmt* | 0.806 | 0.882 | 0.888 | 0.873 | 0.829 | 0.842 | 0.971 | 0.723 | **0** | 2 |
| | | ***previous*** | ↓ ***0.853*** | *0.882* | ↓ *0.889* | *0.873* | *0.834* | *0.856* | ***0.974*** | ↓ *0.728* | - | - |
| | | ***BERTweet.BR*** | 0.828 *(-2.9)* | **0.902** *(2.2)* | **0.892** *(0.4)* | **0.909** *(4.2)* | **0.857** *(2.7)* | **0.874** *(2.1)* | 0.973 *(-0.1)* | **0.733** *(0.7)* | **6** | - |
| | | ***summary*** | bertimbau | bertweetbr | bertweetbr | bertweetbr | bertweetbr | bertweetbr | bertimbau | bertweetbr | - | - |
| | | | 0.853 | 0.902 | 0.892 | 0.909 | 0.857 | 0.874 | 0.974 | 0.733 | - | - |

applying *BERTweet.BR* dominates the best scores, yielding the highest f1-scores for six of the eight datasets. Analyzing the results per method, we observe that *BERTweet.BR* registered the best scores in half of the datasets regarding the *feature-based* approach, while for *fine-tuning* our model achieved the highest performances in 75% of the cases. This overview reinforces the predictive effectiveness of *BERTweet.BR* and indicates a positive response to the inquiries of both research questions *RQ1* and *RQ2*.

With respect to *RQ1.1*, Table 5 shows that *BERTweet.BR* outperforms DAPT-adapted models in about 70% of the experiments (11 out of 16). When compared to TAPT models, responding to *RQ1.2*, our model goes beyond, demonstrating gains in over 80% (13 out of 16) of the scenarios, as reported in Table 4. Regarding *RQ2.1*, we isolate the results

Table 5: Weighted f1-score in the test set for each of the eight datasets for sentiment analysis on top of adapted models resulting from *domain adaptive pre-trained (DAPT)* procedure.

| | | | **covidbr** | **sentbr** | **fiat** | **narrpt** | **mining** | **compbr** | **tweemg** | **unilex** | **wins** | **wins*** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **DAPT** | *feature-based* | *bertimbau* | ↓ **0.781** | 0.863 | **0.830** | 0.840 | 0.816 | 0.834 | 0.955 | 0.674 | **2** | 3 |
| | | *mbert* | 0.773 | 0.861 | 0.821 | 0.848 | 0.809 | 0.830 | 0.956 | 0.673 | **0** | 0 |
| | | *xlmr* | ↓ 0.766 | 0.867 | 0.824 | **0.861** | 0.821 | 0.831 | **0.961** | 0.675 | **2** | 5 |
| | | *xlmt* | 0.768 | 0.851 | 0.826 | 0.838 | 0.802 | 0.834 | 0.949 | 0.672 | **0** | 0 |
| | | ***previous*** | ↓ ***0.781*** | *0.867* | ***0.830*** | ***0.861*** | *0.821* | *0.834* | ***0.961*** | *0.675* | - | - |
| | | ***BERTweet.BR*** | 0.778 *(-0.4)* | **0.881** *(1.7)* | 0.827 *(-0.4)* | 0.843 *(-2.2)* | **0.824** *(0.4)* | **0.848** *(1.7)* | 0.954 *(-0.6)* | **0.679** *(0.7)* | **4** | - |
| | *fine-tuning* | *bertimbau* | ↓ 0.824 | 0.890 | ↓ 0.889 | 0.887 | 0.852 | 0.864 | 0.972 | **0.735** | **1** | 2 |
| | | *mbert* | 0.826 | 0.885 | 0.888 | 0.882 | 0.848 | 0.866 | **0.973** | 0.734 | **1** | 2 |
| | | *xlmr* | 0.823 | 0.894 | 0.890 | 0.889 | 0.849 | 0.869 | 0.972 | 0.731 | **0** | 4 |
| | | *xlmt* | 0.806 | 0.882 | 0.888 | 0.873 | 0.829 | 0.842 | 0.971 | 0.723 | **0** | 0 |
| | | ***previous*** | ↓ *0.826* | *0.894* | ↓ *0.890* | *0.889* | *0.852* | *0.869* | ***0.973*** | ***0.735*** | - | - |
| | | ***BERTweet.BR*** | **0.828** *(0.2)* | **0.902** *(0.8)* | **0.892** ***(0.3)*** | **0.909** *(2.2)* | **0.857** *(0.5)* | **0.874** *(0.5)* | **0.973** *(0.0)* | 0.733 *(-0.3)* | **7** | - |
| | | ***summary*** | bertweetbr<br>0.828 | bertweetbr<br>0.902 | bertweetbr<br>0.892 | bertweetbr<br>0.909 | bertweetbr<br>0.857 | bertweetbr<br>0.874 | bertweetbr<br>0.974 | bertimbau<br>0.733 | -<br>- | -<br>- |

Table 6: Summary of the best results by dataset and approaches feature-based and fine-tuning. The scores are provided as weighted f1-score in the test set for each of the eight datasets for sentiment analysis.

| | **covidbr** | **sentbr** | **fiat** | **narrpt** | **mining** | **compbr** | **tweemg** | **unilex** |
|---|---|---|---|---|---|---|---|---|
| *feature-based* | bertimbau | bertweetbr | bertimbau-dapt | bertimbau-dapt | bertweetbr | bertweetbr | xlmr-dapt | bertweetbr |
| | 0.805 | 0.881 | 0.830 | 0.861 | 0.824 | 0.848 | 0.961 | 0.679 |
| *fine-tuning* | bertimbau | bertweetbr | bertweetbr | bertweetbr | bertweetbr | bertweetbr | bertimbau-tapt | bertweetbr<br>mbert-dapt<br>bertimbau-dapt |
| | **0.864** | **0.902** | **0.892** | **0.909** | **0.857** | **0.874** | **0.974** | **0.735** |
| **best** | **bertimbau**<br>**0.864** | **bertweetbr**<br>**0.902** | **bertweetbr**<br>**0.892** | **bertweetbr**<br>**0.909** | **bertweetbr**<br>**0.857** | **bertweetbr**<br>**0.874** | **bertimbau-tapt**<br>**0.974** | **bertweetbr**<br>**mbert-dapt**<br>**bertimbau-dapt**<br>**0.735** |

of the *feature-based* approach from Tables 3, 4 and 5 to highlight that *BERTweet.BR* achieved the highest scores in 18 out of the 24 cases (75%). Now, shifting the focus to the *fine-tuning* approach and responding to *RQ2.2*, we highlight that *BERTweet.BR* obtained the best predictive performance in 20 out of the 24 experiments (83%).

Conversely, the results indicate that *BERTweet.BR* did not perform so well when applied to the *opcovidbr* dataset. Although our model yielded competitive results across all datasets, it failed to obtain the highest score in any of the experiments with *opcovidbr* dataset, with the sole exception of the fine-tuning after DAPT (Table 5). However, this only victory of our model in *opcovidbr* was due to an atypical decrease in performance of BERTimbau in this very specific scenario. One probable explanation for this sub-optimal performance of *BERTweet.BR* in *opcovidbr* is that the corpus used to pre-train our model only includes tweets from 2004 to 2020. This may have resulted in a lack of knowledge for this particular domain related to the COVID-19 pandemic, as the debate on the COVID-19 crisis has been mostly made after 2020.

It is worth noting that regardless of not following state-of-the-art architectures based on Transformers (VASWANI et al., 2017), the *feature-based* approach with fastText (MIKOLOV; GRAVE, et al., 2018) achieved competitive scores and even delivered performances somewhat better than the multilingual models mBERT (DEVLIN et al., 2019) and XLM-R (CONNEAU et al., 2020). In this sense, we can highlight and suggest that in scenarios of low computational resources and restrictions of time, it is reasonable adopting the fastText model in a sentence classification pipeline and still get adequate accuracy in downstream tasks. Therefore, in scenarios where there is a lack of language-specific pre-trained transformer models, using pre-trained static word embeddings can still be a good choice.

Disregarding our *BERTweet.BR*, Portuguese-native *BERTimbau* and Twitter-native multilingual *XLM-T* ranked the highest among the best-performing models, followed by the generic multi-lingual *XLM-R* and mBERT. In this setting, without *BERTweet.BR*, *BERTimbau* and *XLM-T* accounted for 75% of the best scores in the experiments. In particular, *BERTimbau* achieved the best results when the *fine-tuning* method was applied, whereas *XLM-T* performed better in most cases when a *feature-based* approach was applied. This is shown by the rightmost column *wins** of Tables 3, 4, and 5 and is consistent with previous conclusions that off-the-shelf general-domain pre-trained models are suboptimal for domain-specific tasks.

In addition, we grouped the results by approach, consolidating the highest scores for

each method and dataset, as shown in Table 6. Concerning the two existing strategies for applying language models to downstream tasks, *feature-based* and *fine-tuning*, we observed that *fine-tuning* is indeed the best approach when working with transformer-based models as it outperformed *feature-based* in all scenarios in this study, yielding a 5% boost on average and up to 9% gain over *feature-based*, as reported in *DAPT* for the *Unilex* dataset. Regarding the *feature-based* approach in the context of *RQ2.1*, *BERTweet.BR* achieved the highest scores in 50% of the cases, more than all the other benchmark models. Concerning the *fine-tuning* approach under *RQ2.2*, our model achieved the top performance for six out of the eight datasets in this study.

Regarding the domain adaptation strategy, the results show that TAPT and DAPT both led to performance gains as originally proposed by (GURURANGAN et al., 2020). Specifically, the average enhancements observed for TAPT and DAPT over the original checkpoints amounted to 1.8% and 6.3%, respectively. However, these procedures do not always provide benefits. For example, as indicated in Table 4, when applied to Portuguese-native BERTimbau and multilingual mBERT models, *TAPT* resulted in a loss of performance or null gains in a few cases. We found that *DAPT* consistently boosts the performance of the original models, yielding a gain of up to 10%. Simultaneously, the method also reported a loss in very few cases of a specific dataset, as indicated by the ↓ symbol in Table 5. Additionally, *DAPT* showed an average gain of more than 4% compared with *TAPT*, and up to approximately 9% as we see with *mBERT* with the *feature-based* approach. Overall, applying the *fine-tuning* approach over *DAPT* models proved to be the best pipeline among all the different versions of the language models considered in this study, from the original checkpoints to adapted models through TAPT and DAPT. This is demonstrated when we examine models individually and scores by dataset, as shown in Table 6.

Although we confirmed that continued pre-training strategies confer essential additional knowledge to language models, as observed mainly with *DAPT*, experiments revealed that such enhancements are still suboptimal compared to training from scratch, as our *BERTweet.BR* provides better results for the majority of the scenarios in this study, whether we apply the *feature-based* or *fine-tuning* approach. These findings address research question *RQ1*, as we have demonstrated the effectiveness of training a completely new model for tweets. Compared with the previous best results per method, *BERTweet.BR* delivered gains of up to 4.5% in *feature-based* and up to 4.2% in the *fine-tuning approach*.

In summary, the results presented in this section highlight the potential of models crafted for a specific language and domain as being more suited to handle social media tasks. In particular, *BERTweet.BR* fills the gap in the domain of Portuguese tweets, as it substantially outperforms the baselines in most scenarios.

# 6 Qualitative Analysis

Upon examining the results reported in this study, it is noteworthy that all studied models reported their highest and lowest performances in two particular datasets: *tweemg*, as the one with the top f1-scores, and *unilex* with nearly all the lowest scores, as depicted in Table 7. There, it is shown the performance ranking of benchmark models, listing in ascending order the datasets according to the weighted f1-score obtained by each model when applied *feature-based* and *fine-tuning* approaches on top of the *original checkpoints* of the models and adapted versions resulting from *domain adaptive pre-trained* (DAPT) procedure. In fact, the evaluation of this table discloses a notable similarity in the performance rankings across the models, despite their distinct domains and pre-training procedures. For instance, besides *tweetmg*, all models ranked *sentbr* and *comput* datasets within the top four highest performances when following the *feature-based* approach. Likewise, the models also placed *mining* and *covidbr* datasets within their four lowest-performing lists, along with *unilex*. Moreover, *BERTweet.BR* and *XLM-T*, which are the sole models initially pre-trained for tweets, appear to concur on the degree of difficulty across datasets. They reported an identical order in their rankings following *fine-tuning* and exhibited only one discrepancy concerning *feature-based* approach. The other remaining models, pre-trained for generic-domain, also share most of their rankings as they reported identical lists of the top four highest and bottom four performance, independently of the approach. More precisely, when *fine-tuning* is applied they all positioned *covidbr*, *sentbr*, *fiat* and *tweemg* datasets as top performers in ascending order. Lastly, we highlight that generic-domain models display a higher degree of congruence after *domain adaptive pre-training*. As illustrated by the DAPT rankings in the latter half of Table 7, the three adapted models now exhibit a perfect match across rankings when implementing the *feature-based* method and only a single difference when *fine-tuning* is applied. As previously generic models become adapted to tweets, their results increasingly resemble those of XLM-T and *BERTweet.BR*, which confirms the effectiveness of adaptation methods. As a result, these models now share the lists of the top four and bottom four performances with XLM-T and *BERTweet.BR*, although the exact order varies.

Table 7: Performance ranking of benchmark models, listing in ascending order the datasets according to the weighted f1-score obtained by each model.

| | | **fasttext** | | **bertimbau** | | **mbert** | | **xlmr** | | **xlmt** | | **bertweetbr** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **ORIGINAL CHECKPOINTS** | *feature-based* | **unilex** | **0.661** | **unilex** | **0.668** | **unilex** | **0.602** | **unilex** | **0.649** | **unilex** | **0.672** | **unilex** | **0.679** |
| | | *covidbr* | 0.770 | *mining* | 0.784 | *narrpt* | 0.651 | *narrpt* | 0.738 | *covidbr* | 0.768 | *covidbr* | 0.778 |
| | | *narrpt* | 0.776 | *narrpt* | 0.804 | *mining* | 0.706 | *covidbr* | 0.771 | *mining* | 0.802 | *mining* | 0.824 |
| | | *mining* | 0.776 | *covidbr* | 0.805 | *covidbr* | 0.716 | *mining* | 0.777 | *fiat* | 0.826 | *fiat* | 0.827 |
| | | *comput* | 0.781 | *comput* | 0.821 | *sentbr* | 0.738 | *comput* | 0.799 | *comput* | 0.834 | *narrpt* | 0.843 |
| | | *fiat* | 0.815 | *fiat* | 0.826 | *comput* | 0.752 | *sentbr* | 0.803 | *narrpt* | 0.838 | *comput* | 0.848 |
| | | *sentbr* | 0.828 | *sentbr* | 0.842 | *fiat* | 0.774 | *fiat* | 0.807 | *sentbr* | 0.851 | *sentbr* | 0.881 |
| | | **tweemg** | **0.945** | **tweemg** | **0.954** | **tweemg** | **0.927** | **tweemg** | **0.942** | **tweemg** | **0.949** | **tweemg** | **0.954** |
| | *fine-tuning* | | | **unilex** | 0.732 | **narrpt** | **0.688** | **unilex** | **0.715** | **unilex** | **0.723** | **unilex** | **0.733** |
| | | | | *mining* | 0.828 | *unilex* | 0.703 | *mining* | 0.732 | *covidbr* | 0.806 | *covidbr* | 0.828 |
| | | | | *narrpt* | 0.855 | *mining* | 0.759 | *comput* | 0.742 | *mining* | 0.829 | *mining* | 0.857 |
| | | | | *comput* | 0.855 | *comput* | 0.773 | *narrpt* | 0.742 | *comput* | 0.842 | *comput* | 0.874 |
| | | | | *covidbr* | 0.864 | *covidbr* | 0.786 | *covidbr* | 0.768 | *narrpt* | 0.873 | *fiat* | 0.892 |
| | | | | *sentbr* | 0.877 | *sentbr* | 0.841 | *sentbr* | 0.825 | *sentbr* | 0.882 | *sentbr* | 0.902 |
| | | | | *fiat* | 0.891 | *fiat* | 0.878 | *fiat* | 0.862 | *fiat* | 0.888 | *narrpt* | 0.909 |
| | | | | **tweemg** | **0.972** | **tweemg** | **0.972** | **tweemg** | **0.967** | **tweemg** | **0.971** | **tweemg** | **0.973** |
| | | | | **bertimbau** | | **mbert** | | **xlmr** | | | | | |
| **DAPT** | *feature-based* | | | **unilex** | **0.674** | **unilex** | **0.673** | **unilex** | **0.675** | | | | |
| | | | | *covidbr* | 0.781 | *covidbr* | 0.773 | *covidbr* | 0.766 | | | | |
| | | | | *mining* | 0.816 | *mining* | 0.809 | *mining* | 0.821 | | | | |
| | | | | *fiat* | 0.830 | *fiat* | 0.821 | *fiat* | 0.824 | | | | |
| | | | | *comput* | 0.834 | *comput* | 0.830 | *comput* | 0.831 | | | | |
| | | | | *narrpt* | 0.840 | *narrpt* | 0.848 | *narrpt* | 0.861 | | | | |
| | | | | *sentbr* | 0.863 | *sentbr* | 0.861 | *sentbr* | 0.867 | | | | |
| | | | | **tweemg** | **0.955** | **tweemg** | **0.956** | **tweemg** | **0.961** | | | | |
| | *fine-tuning* | | | **unilex** | **0.735** | **unilex** | **0.735** | **unilex** | **0.731** | | | | |
| | | | | *covidbr* | 0.824 | *covidbr* | 0.826 | *covidbr* | 0.823 | | | | |
| | | | | *mining* | 0.852 | *mining* | 0.848 | *mining* | 0.849 | | | | |
| | | | | *comput* | 0.864 | *comput* | 0.866 | *comput* | 0.869 | | | | |
| | | | | *narrpt* | 0.887 | *narrpt* | 0.882 | *narrpt* | 0.889 | | | | |
| | | | | *fiat* | 0.889 | *sentbr* | 0.885 | *fiat* | 0.890 | | | | |
| | | | | *sentbr* | 0.890 | *fiat* | 0.888 | *sentbr* | 0.894 | | | | |
| | | | | **tweemg** | **0.972** | **tweemg** | **0.973** | **tweemg** | **0.972** | | | | |

What we observe from the aforementioned findings is that despite the variation in weighted F1-scores across different models and datasets, there exists a discernible pattern of performance clusters, grouping datasets into levels of difficulty and their corresponding compatibility with models. This motivated the examination of two datasets of the clusters to clarify potential factors accounting for the highest and lowest performances of the collection: *unilex* and *tweetmg*.

## 6.1 Unilex Dataset

The *unilex* dataset presents the most significant challenge among the collection, as evidenced by weighted f1-scores ranging between 0.602 and 0.735. While these scores may not be deemed low, they underscore that the *unilex* dataset does pose the most difficulty for all of the models.

First, the topic of politics is a subjective matter in which opinions diverge among individuals based on their personal experiences, values, and cultural backgrounds. This is particularly evident in recent contexts, marked by heightened polarization in political views, made even more intense by the popularity of social networks. In this regard, determining whether a tweet carries a positive or negative opinion out of this subjectivity is indeed a more difficult task. To illustrate, when we extract the list of most frequent words in positive and negative tweets of *unilex*, on average three-fifths of them are presented in both lists. Also, more than 40% of the most frequent words are present in all subsets of positive, negative, and neutral tweets. This evidences the subjectivity of the matter of politics and how challenging it is to spot the sentiment present in its tweets.

Particularly, when we look at the twenty most frequent words in tweets of *unilex* that are labeled as positive and negative as shown in Table 8, we see that words like *#brasil, #dilma, #lula, #pcdob, #pmdb, #psdb, #pt*[1], *brasil, dilma*[2] and *governo*[3] exists in both subsets, emphasizing the absence of a discernible pattern to distinguish sentiment. In Table 8, words in bold indicate that they appear concomitantly in all three subsets. Underlined words appear in both *positive* and *negative* subsets. In italic, words that appear in both *positive* and *neutral* subsets. Double-underlined words appear in both *negative* and *neutral* subsets.

Names of Brazilian politicians and political parties as well as words such as *brasil* and *governo* — which are more likely to convey a neutral connotation — are consistently found in both positive and negative tweets. Additionally, when analyzing the twenty most frequent words in neutral tweets, an almost identical list of intersecting terms emerges: *#brasil, #dilma, #pcdob, #pmdb, #psdb, #pt* and *governo*. Thereby reaffirming the lack of a pattern for identifying sentiment as the same words are frequent in tweets with

[1] PT, PSDB, PMDB and PCdoB are Brazilian political parties. https://en.wikipedia.org/wiki/List_of_political_parties_in_Brazil

[2] Dilma Rouseff, a politician who served as the 36th president of Brazil, holding the position from 2011 until her impeachment and removal from office on 31 August 2016. https://en.wikipedia.org/wiki/Dilma_Rousseff

[3] *government* in Portuguese

three different sentiments. The absence of *#lula*[4] in this neutral list of frequent words implies that the name of the former president is associated with either positive or negative sentiment, but not neutral, a fact that once more exemplifies the politically polarized climate in Brazil.

Table 8: The twenty most frequent words in *Unilex* dataset and their absolute frequency within each subset of tweets labeled as *positive*, *negative*, and *neutral* sentiments.

| word | pos | neg | neu |
|---|---|---|---|
| ***#pt*** | 1521 | 2568 | 1354 |
| ***#psdb*** | 508 | 453 | 183 |
| ***#brasil*** | 238 | 350 | 416 |
| ***#pmdb*** | 171 | 388 | 412 |
| ***#pcdob*** | 154 | 268 | 488 |
| *#psd* | 137 | - | 368 |
| brasil | 183 | 256 | - |
| *#pp* | 128 | - | 339 |
| *#pr* | 253 | - | 256 |
| ***#dilma*** | 140 | 270 | 131 |
| governo | 136 | 269 | - |
| *#novo* | 129 | - | 361 |
| golpe | - | 270 | - |
| psd | - | - | 351 |
| #mst | - | - | 228 |
| *#rede* | 134 | - | 141 |
| #pdt | - | - | 230 |
| contra | - | 212 | - |
| #golpe | - | 176 | 131 |
| #lula | 106 | 192 | - |

In fact, a comprehensive examination of the dataset vocabularies and their intersections, as illustrated in Table 9, demonstrates that *covidbr*, *mining* and *fiat* compose the group of datasets that exhibit the highest degree of similarity relative to *unilex* among the entire collection. In this table, it is presented the vocabulary similarity matrix, which represents the lexical congruence among various datasets, considering the top 1,000 most frequent words within each dataset, after excluding stopwords in the NLTK Portuguese dictionary. The degree of similarity is quantified by the proportion of shared terms present

[4]Luis Inacio Lula da Silva, also known as Lula da Silva or simply Lula, is Brazil's 39th and current president. https://en.wikipedia.org/wiki/Luiz_In%C3%A1cio_Lula_da_Silva

in both vocabularies, with a maximum of 100. The five highest similarities are emphasized in bold, while the five lowest percentage similarities are underlined.

Remarkably, these four datasets *covidbr*, *mining*, *fiat* and *unilex* correspond to the same set of the four most challenging datasets for the Twitter-domain models, as shown in Table 7. This finding implies that vocabularies do play a crucial role in influencing the performance of models in the sentiment classification tasks, as corroborated by the *feature-based* scores of Twitter models in the form of original checkpoints (*BERTweet.BR* and XLM-T) or after adaptation of mBERT, BERTimbau, and XLM-R.

Table 9: The Vocabulary similarity matrix represents the lexical congruence among various datasets, considering the top 1,000 most frequent words within each dataset, after excluding stopwords in the NLTK Portuguese dictionary.

| | ***covidbr*** | ***sentbr*** | ***fiat*** | ***narrpt*** | ***mining*** | ***compbr*** | ***tweemg*** | ***unilex*** |
|---|---|---|---|---|---|---|---|---|
| ***covidbr*** | x | 28.6 | 31.0 | 25.3 | 33.1 | 24.9 | 28.3 | **42.8** |
| ***sentbr*** | 28.6 | x | 38.9 | 35.9 | 39.2 | 33.5 | 17.6 | 39.7 |
| ***fiat*** | 31.0 | 38.9 | x | 36.7 | **41.2** | 38.7 | 25.4 | **42.4** |
| ***narrpt*** | 25.3 | 35.9 | 36.7 | x | 37.9 | 38.0 | 18.8 | 35.0 |
| ***mining*** | 33.1 | 39.2 | **41.2** | 37.9 | x | **42.3** | 24.1 | **43.0** |
| ***compbr*** | 24.9 | 33.5 | 38.7 | 38.0 | **42.3** | x | 17.8 | 33.4 |
| ***tweemg*** | 28.3 | 17.6 | 25.4 | 18.8 | 24.1 | 17.8 | x | 32.1 |
| ***unilex*** | **42.8** | 39.7 | **42.4** | 35.0 | **43.0** | 33.4 | 32.1 | x |

To conclude, an additional challenging aspect of *unilex* is the high presence of very short tweets, with approximately 25% of its messages comprising fewer than ten words. Although a substantial proportion of tweets of *unilex* does contain hashtags (90%), there is a relatively low incidence of user mentions (about 20% of tweets) and minimal inclusion of emoticons and emojis (less than 10%). The presence of URLs and email addresses is virtually absent. Given that tweets are typically characterized by concise sentences that frequently rely on user mentions, hashtags, and emoticons, it can be suggested that *unilex* does not accurately adhere to a conventional *tweet-like* dataset, resulting in decreased compatibility with models specifically designed for the tweets domain. As a result, this gap makes applying pre-trained Twitter models like *BERTweet.BR* and XLM-T more challenging. Finally, we also observed that this difficulty applies to the Twitter-adapted versions of models such as mBERT, BERTimbau, and XLM-R.

## 6.2 TweetsMG Dataset

The *tweemg* dataset emerges as the best-performing dataset for all models analyzed in this study, achieving a minimum score of 0.927. It boasts the highest number of retweets (29.6%) by a significant margin and presents the highest proportion of emoticons (83.92%), followed by *covidbr* (70.83%). Additionally, it has the highest proportion of URLs (82.23%), with *covidbr* trailing at 69.00%. Regarding user mentions, *tweemg* holds the second place with a proportion of 49.20%, only surpassed by *mining* at 68.29%. The attributes above illustrate a notable distinction between *tweetmg* and *unilex* with regards to their adherence to *tweet-like* datasets and the impact of it on their average scores. What we observed is that the degree of alignment between a dataset and the *standard tweet* positively impacts the performance of the model, as evidenced by the strong results obtained with *tweemg* in contrast to the relatively low average scores of *unilex*.

*Tweemg* integrates intrinsic features that facilitate the clustering of its tweets, thereby simplifying the text classification process. First, it follows a balanced distribution in terms of classes. Then, we observed that, on average, positive tweets in *tweemg* tend to exhibit shorter lengths than negative ones. Furthermore, in contrast to *unilex*, vocabularies of *tweemg* display low intersection rates (averaging less than 15%) when evaluating the most frequent words within the groups of positive, negative, and neutral tweets. Mainly, when analyzing the twenty most common words in *tweemg* by sentiment, only three appear concurrently in both positive and negative subsets. This suggests the presence of virtually independent vocabularies for each sentiment category, unlike what we observe with *unilex*, where we encounter three times as many the number of shared words among all distinct groups. This indicates the lack of a discernible pattern to differentiate classes in the most challenging dataset in the collection, positioning *unilex* and *tweemg* in opposite directions. Objectively, the relative independence of vocabularies of the *tweemg* dataset results in reduced entropy, which subsequently facilitates the job of machine learning algorithms in splitting tweets into groups of different sentiments.

This specific ability of *tweemg* is further evidenced when we plot the embeddings extracted from pre-trained models after dimensionality reduction using t-SNE (VAN DER MAATEN, 2014) as shown in Figure 9. In the presented figures, distinct colors represent different labels. Each point corresponds to an individual tweet in the dataset, with 768-dimension embeddings derived from pre-trained models BERTweet.BR and BERTimbau. Dimensionality reduction was performed using t-SNE. The color scheme denotes the three sentiment labels: red for negative sentiment, blue for positive sentiment, and orange for

neutral sentiment. Distinct sentiment clusters are discernible, highlighting the similarity among tweets sharing the same label and potentially indicating the ease of detecting sentiments within the *tweemg* dataset. Embeddings generated by *BERTweet.BR* and BERTimbau suggest that groups of tweets with the same labels in the *tweemg* dataset also possess similar semantic meanings, as illustrated by the proximity of their vector representations in the two-dimensional space. Groups of tweets with the same label appear clustered together while distant from tweets of other sentiments. Consequently, it becomes easier to infer the sentiment of a given tweet visually. In fact, this is an intrinsic characteristic of this dataset as, regardless of the pre-trained model used to generate the embeddings, tweets from the *tweemg* dataset prove to be reasonably distinguishable when visualized in the plots. Precisely, samples from the *tweemg* dataset are more readily identifiable using t-SNE technique compared to all other datasets in the collection. For instance, samples with different labels in *unilex* dataset are virtually indistinguishable, as depicted in Figure 10. That is, in contrast to the *tweemg* dataset, this plot visually illustrates the inherent challenge in discerning sentiment within *unilex*, as it is virtually impossible to identify a clear semantic separation among the different labels of the tweets.

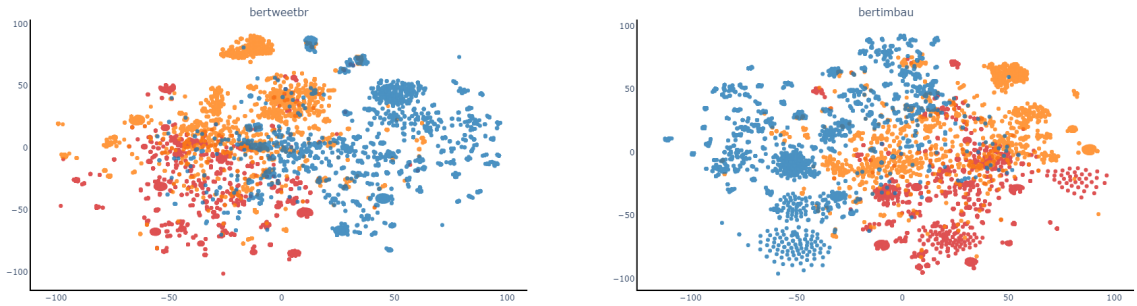


Figure 9: Visualization of embeddings for the *tweemg* dataset. Each point corresponds to an individual tweet in the dataset, with 768-dimension embeddings derived from pre-trained models BERTweet.BR and BERTimbau.

## 6.3 The Effect of specific Tokenizer

Although training a specific vocabulary on tweets cannot guarantee the absence of *out-of-vocabulary* tokens, it does provide a more concise way of representing sentences, as demonstrated in Table 10 where we show various statistics of each dataset after normalization and tokenization. For each dataset is indicated the minimum, maximum,

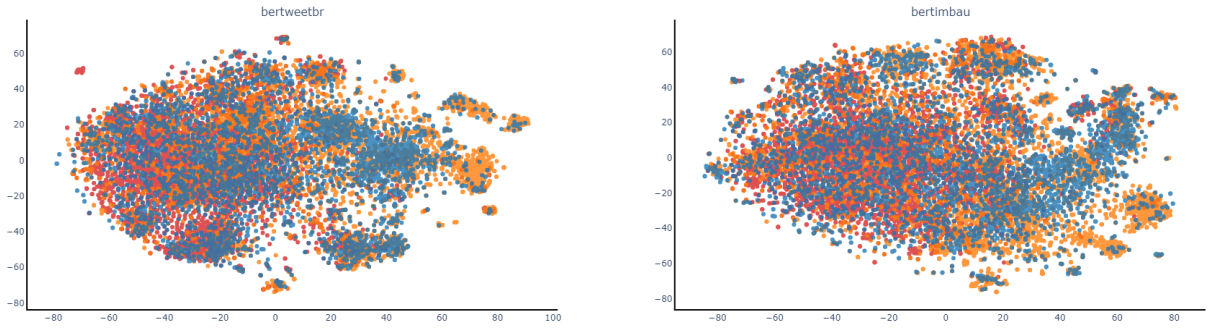


Figure 10: Visualization of embeddings for the *unilex* dataset. Each point corresponds to an individual tweet in the dataset, with 768-dimension embeddings derived from pre-trained models BERTweet.BR and BERTimbau.

and average length in tokens after each model tokenizer. In *out-of-vocabulary (OOV)*, it is provided the number of rows, the total number of occurrences, and the quantity of unique OOV found in each dataset and after each particular tokenizer (*#has*, *#total* and *#unique* respectively). The *unique max* column indicates the maximum number of occurrences of one OOV token in the same tweet. Finally, it is presented the three most frequent *out-of-vocabulary* tokens along with their frequency in each of the datasets. Note that for *XLM-R* and *XLM-T*, models with the largest vocabularies, tokenizers were able to decode the whole dataset.

As illustrated in the table, on average, *BERTweet.BR* tokenizer generates the shortest decoded sentences across all datasets, demonstrating that our model's tokenizer has successfully learned highly specific tokens for a corpus of Portuguese tweets, enabling it to generate concise representations. Comparatively, using the plain Portuguese tokenizer of BERTimbau on the same datasets resulted in decoded sentences 33% longer.

The XLM-R and XLM-T multilingual tokenizers possess the largest vocabulary among all models, encompassing 250,002 entries. This is over double the size of the second-largest vocabulary, which contains 119,547 entries and belongs to the mBERT model. As expected, the vocabulary size influences the number of *out-of-vocabulary (OOV)* instances. The only dataset in which XLM-R and XLM-T tokenizers exhibit *OOV* occurrences is *unilex*. It can be expected that a low frequency of *OOV* tokens may contribute to improved results, as evidenced by the XLM-T model, which has virtually no *OOV* tokens across all datasets. However, it is also observed that models such as BERTimbau, which has the smallest vocabulary size and the highest frequency of *OOV* in all datasets, as shown in Table 10, recorded significant results trailing only behind *BERTweet.BR* in several instances.

Table 10: Statistics of each dataset after normalization and tokenization. *(*) Based on classical TwitterTokenizer from NLTK package.*

| | | Tokens | | | Out-of-Vocabulary | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | **min** | **avg** | **max** | **#has** | **#total** | **#unique** | **unique max** | **most freq.** |
| *covidbr* | *fasttext** | 6 | 32.9 | 68 | 514 | 1456 | 484 | 22 | *covid (75); #coronavírus (57); #coronavirus (45)* |
| | *bertimbau* | 13 | 52.6 | 138 | 31 | 41 | 16 | 2 | *q (22); 2ª (3); 25 (2)* |
| | *mbert* | 14 | 54.2 | 146 | 31 | 55 | 10 | 2 | *“ (20); ” (19); ‘ (6)* |
| | *xlmr* | 11 | 52.6 | 164 | 0 | 0 | 0 | 0 | - |
| | *xlmt* | 11 | 52.6 | 164 | 0 | 0 | 0 | 0 | - |
| | *BERTweet.BR* | 8 | 39.9 | 85 | 18 | 49 | 19 | 2 | 🙏 *(8); antónio (2);* 👋🏻 *(2)* |
| *sentbr* | *fasttext** | 2 | 13.6 | 47 | 7769 | 12532 | 1354 | 11 | *#masterchefbr (2761); #encontro (1091); #videoshowaovivo (960)* |
| | *bertimbau* | 6 | 24.3 | 82 | 454 | 568 | 43 | 6 | *q (424); ñ (33); ♡ (28)* |
| | *mbert* | 7 | 24.8 | 97 | 8 | 15 | 11 | 6 | *” (2); ‘ (2); — (1)* |
| | *xlmr* | 7 | 24.0 | 124 | 0 | 0 | 0 | 0 | - |
| | *xlmt* | 7 | 24.0 | 124 | 0 | 0 | 0 | 0 | - |
| | *BERTweet.BR* | 4 | 18.1 | 56 | 29 | 44 | 20 | 6 | ☺️ *(11); \u200d ♀(5);* ☹️ *(4)* |
| *fiat* | *fasttext** | 2 | 19.1 | 43 | 6946 | 11831 | 2295 | 11 | *#fiat (140); #cartolafc (116); 15 07 (71)* |
| | *bertimbau* | 7 | 30.9 | 111 | 387 | 471 | 72 | 3 | *q (280); ¬ (34); ñ (16)* |
| | *mbert* | 7 | 30.4 | 72 | 5 | 5 | 3 | 1 | *´ (3); ¤(1); ‘ (1)* |
| | *xlmr* | 6 | 29.3 | 70 | 0 | 0 | 0 | 0 | - |
| | *xlmt* | 6 | 29.3 | 70 | 0 | 0 | 0 | 0 | - |
| | *BERTweet.BR* | 4 | 24.3 | 63 | 244 | 247 | 13 | 2 | *\x97 (226);\x99 (3); ariquemes (3)* |
| *narrpt* | *fasttext** | 2 | 16.6 | 38 | 504 | 929 | 353 | 6 | *:d (13); #sorteio (12); #promoção (10)* |
| | *bertimbau* | 5 | 27.5 | 59 | 53 | 69 | 17 | 2 | *q (33); ¬ (11); qe (16)* |
| | *mbert* | 6 | 27.4 | 54 | 16 | 22 | 7 | 2 | *“ (8), ” (6), — (4)* |
| | *xlmr* | 4 | 24.9 | 50 | 0 | 0 | 0 | 0 | - |
| | *xlmt* | 4 | 24.9 | 50 | 0 | 0 | 0 | 0 | - |
| | *BERTweet.BR* | 4 | 20.7 | 43 | 0 | 0 | 0 | 0 | - |
| *mining* | *fasttext** | 2 | 16.9 | 44 | 2018 | 4846 | 753 | 12 | *#claro (104); #bradesco (91); #vivoemrede (85)* |
| | *bertimbau* | 6 | 28.5 | 83 | 156 | 196 | 19 | 2 | *q (138); ¬ (14); ñ (10)* |
| | *mbert* | 7 | 29.4 | 56 | 0 | 0 | 0 | 0 | - |
| | *xlmr* | 6 | 28.2 | 53 | 0 | 0 | 0 | 0 | - |
| | *xlmt* | 6 | 28.2 | 53 | 0 | 0 | 0 | 0 | - |
| | *BERTweet.BR* | 4 | 21.3 | 46 | 64 | 64 | 5 | 1 | *#bancodobrasil (57); #bancocentralbr (2); #bancodobradesco (2)* |
| *compbr* | *fasttext** | 1 | 19.1 | 55 | 1647 | 3197 | 673 | 11 | *#dell (75); #notebook (32); v14t (22)* |
| | *bertimbau* | 5 | 33.1 | 89 | 107 | 134 | 27 | 3 | *q (81); 5ª (7); ñ (6)* |
| | *mbert* | 6 | 31.0 | 95 | 28 | 33 | 7 | 2 | *- (13); ” (9); —– (4)* |
| | *xlmr* | 4 | 27.9 | 92 | 0 | 0 | 0 | 0 | - |
| | *xlmt* | 4 | 27.9 | 92 | 0 | 0 | 0 | 0 | - |
| | *BERTweet.BR* | 4 | 24.9 | 72 | 4 | 4 | 4 | 1 | ☺️ *(1); \x99 (1); aspect (1)* |
| *tweemg* | *fasttext** | 1 | 18.2 | 38 | 8038 | 16885 | 847 | 10 | *#timbeta (206); #globo (130); #operacaobetalab (121)* |
| | *bertimbau* | 6 | 31.2 | 62 | 471 | 1100 | 49 | 3 | *q (676); ñ (312); 16ª (15)* |
| | *mbert* | 5 | 31.1 | 58 | 101 | 121 | 6 | 2 | *– (37); ‘ (28); “ (19)* |
| | *xlmr* | 4 | 29.2 | 62 | 0 | 0 | 0 | 0 | - |
| | *xlmt* | 4 | 29.2 | 62 | 0 | 0 | 0 | 0 | - |
| | *BERTweet.BR* | 3 | 22.3 | 51 | 102 | 107 | 11 | 2 | *#raynniere (75); timóteo (13);* ☺️ *(4)* |
| unilex | *fasttext** | 1 | 17.2 | 73 | 11632 | 43693 | 9531 | 27 | *#pt ( 5444); #psdb (1144); #brasil (1004)* |
| | *bertimbau* | 3 | 32.3 | 176 | 792 | 1123 | 147 | 7 | *q (469); ñ (105); • (47)* |
| | *mbert* | 4 | 33.3 | 202 | 288 | 483 | 51 | 7 | *“ (147); ”, (133); – (43)* |
| | *xlmr* | 3 | 32.4 | 205 | 62 | 179 | 80 | 10 | 😂 *(13);* 💙 *(12);* 😍 *(8)* |
| | *xlmt* | 3 | 32.4 | 205 | 62 | 179 | 80 | 10 | 😂 *(13);* 💙 *(12);* 😍 *(8)* |
| | *BERTweet.BR* | 3 | 24.3 | 72 | 480 | 1208 | 169 | 7 | 👋🏻 *(85); #dilmarousseff (79); paschoal (38)* |

# 7 Conclusion and Future Work

In this work, we presented the first public large-scale pre-trained model specific to the Brazilian Portuguese tweet domain. To assess our model, we explored several pipelines, applying both *feature-based* and *fine-tuning* approaches on top of *off-the-shelf* language models as well as adjusted models to the context of Twitter in Portuguese. We found that *BERTWeet.BR*, trained from scratch, performed better than its competitors in most scenarios of sentiment classification with datasets of varying sizes. *BERTweet.BR* achieved the absolute best F1-score on 36 out of 48 experiments while matching the previous best performances on two scenarios, which certifies the effectiveness of a domain-specific language model pre-trained for Portuguese tweets. This clearly provides a positive response to inquiries about both research questions *RQ1* and *RQ2*.

In this regard, our work proved to be a relevant step forward in the field of natural language processing in Portuguese, in which we release model and code from the transformers library[1] and Github[2] with the aim of advancing future research in analytical tasks for Portuguese. However, it is essential to acknowledge potential limitations, such as the model's ability to capture more recent linguistic trends and its generalization to other task types. Additionally, the rapidly evolving NLP landscape, especially with the emergence of Large Language Model (LLM), may impact performance benchmarks and future research directions.

The corpus utilized for pre-training our model encompasses data spanning from 2004 to 2020. Due to the dynamic nature of debates on the Twitter platform, one natural next step is to incorporate new knowledge into our model with the latest tweets from 2020 onwards. In particular, our findings showed that *BERTWeet.BR* did not do so well with the *opcovidbr* dataset, as the debate on the COVID-19 crisis has been mostly made after 2020. Thus, acquiring data after that point and retraining the models could lead to different results. With more new data, we would also adopt a longer training process

[1] https://huggingface.co/melll-uff/bertweetbr
[2] https://github.com/MeLLL-UFF/BERTweet.br

and also release a large version of the model — BERTweet (NGUYEN; VU; NGUYEN, 2020), for example, was pretrained for ten more epochs in a corpus eight times larger than *BERTweet.BR*. On the other hand, a lighter and cheaper version of *BERTweet.BR* following the training procedures of DistilBERT (SANH et al., 2019) would result in an alternative model, more viable for production, especially in scenarios with costs and low latency constraints.

The model's performance is evaluated on a collection of eight datasets for sentiment analysis. While this provides a comprehensive evaluation, it is still possible that the model's performance may vary when applied to other tasks, domains, or datasets that were not part of the evaluation process. For this reason, we are also planning to extend this study evaluating our model on different downstream tasks, specifically token-level classification tasks like Named Entity Recognition (NER) and Part-of-Speech (PoS) tagging. We would like to investigate *BERTWeet.BR* performance on other classification and sequence-based tweets tasks.

Finally, in light of the significant recent advancements in NLP related to large language models (LLMs) like GPT-4 (OPENAI, 2023), LLaMA (TOUVRON et al., 2023), and BLOOM (WORKSHOP et al., 2023), next we plan to follow in-context learning applying the few/one/zero-shot approaches (BROWN et al., 2020) to LLMs and compare these performances to *BERTWeet.BR*, a scenario we did not explore in this study.

**Ethics Statement**

**Datasets.** All datasets considered in this manuscript were gathered from previous work that made them publicly available. Although we have not directly collected any tweets, we are aware that using data collected from the Twitter platform should raise ethical reflections. Even though Twitter users assume their posts are not private, they are usually not explicitly informed that what they write can be used for scientific – our case – or commercial – not our case – purposes. Besides, they might usually assume that their tweets are ephemeral whilst they, in fact, can be collected and stored by anyone anywhere. We tried our best not to include sensitive content in our examples and not disclose the identity of their authors.

**Language model.** Given that this work strongly relies on large-scale language models and datasets composed of social media texts, despite the best intentions, we anticipate possible ethical and social risks by perpetuating social biases and providing false or misleading information. In the case of language models, these risks usually spring from the chosen training *corpora* used to pre-train such large models. If your intent is to use

our pre-trained model or a fine-tuned version in production, please be aware that, while *BERTweet.BR* like many other models is a powerful tool, it comes with limitations. To enable pre-training on large amounts of data, we scrape all the content we could find from Twitter until the year 2020, taking the best as well as the worst of what was available on this social media.

# REFERENCES


ABDELALI, Ahmed; HASSAN, Sabit; MUBARAK, Hamdy; DARWISH, Kareem; SAMIH, Younes. Pre-training bert on arabic tweets: Practical considerations. *arXiv preprint arXiv:2102.10684*, 2021.

BAHDANAU, Dzmitry; CHO, Kyung Hyun; BENGIO, Yoshua. Neural machine translation by jointly learning to align and translate. English (US). In: 3rd International Conference on Learning Representations, ICLR 2015 ; Conference date: 07-05-2015 Through 09-05-2015.

BARBIERI, Francesco; CAMACHO-COLLADOS, Jose; ESPINOSA ANKE, Luis; NEVES, Leonardo. TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification. In: FINDINGS of the Association for Computational Linguistics: EMNLP 2020. Online: Association for Computational Linguistics, Nov. 2020. P. 1644–1650. DOI: 10.18653/v1/2020.findings-emnlp.148. Available from: <https://aclanthology.org/2020.findings-emnlp.148>.

BARBIERI, Francesco; ESPINOSA-ANKE, Luis; CAMACHO-COLLADOS, Jose. XLM-T: Multilingual Language Models in Twitter for Sentiment Analysis and Beyond. In: PROCEEDINGS of LREC. [S.l.: s.n.], 2022.

BELTAGY, Iz; LO, Kyle; COHAN, Arman. SciBERT: A Pretrained Language Model for Scientific Text. In: PROCEEDINGS of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong, China: Association for Computational Linguistics, Nov. 2019. P. 3615–3620. DOI: 10.18653/v1/D19-1371. Available from: <https://aclanthology.org/D19-1371>.

BIRD, Steven. NLTK: the natural language toolkit. In: PROCEEDINGS of the COLING/ACL 2006 Interactive Presentation Sessions. [S.l.: s.n.], 2006. P. 69–72.

BROWN, Tom; MANN, Benjamin; RYDER, Nick; SUBBIAH, Melanie; KAPLAN, Jared D; DHARIWAL, Prafulla; NEELAKANTAN, Arvind; SHYAM, Pranav; SASTRY, Girish; ASKELL, Amanda; AGARWAL, Sandhini; HERBERT-VOSS, Ariel; KRUEGER, Gretchen; HENIGHAN, Tom; CHILD, Rewon; RAMESH, Aditya; ZIEGLER, Daniel; WU, Jeffrey; WINTER, Clemens;

HESSE, Chris; CHEN, Mark; SIGLER, Eric; LITWIN, Mateusz; GRAY, Scott; CHESS, Benjamin; CLARK, Jack; BERNER, Christopher; MCCANDLISH, Sam; RADFORD, Alec; SUTSKEVER, Ilya; AMODEI, Dario. Language Models are Few-Shot Learners. In: LAROCHELLE, H.; RANZATO, M.; HADSELL, R.; BALCAN, M.F.; LIN, H. (Eds.). *Advances in Neural Information Processing Systems*. [S.l.]: Curran Associates, Inc., 2020. v. 33, p. 1877–1901. Available from: <https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf>.

BRUM, Henrico; GRAÇAS VOLPE NUNES, Maria das. Building a Sentiment Corpus of Tweets in Brazilian Portuguese. In: CHAIR), Nicoletta Calzolari (Conference; CHOUKRI, Khalid; CIERI, Christopher; DECLERCK, Thierry; GOGGI, Sara; HASIDA, Koiti; ISAHARA, Hitoshi; MAEGAARD, Bente; MARIANI, Joseph; MAZO, HÚlŔne; MORENO, Asuncion; ODIJK, Jan; PIPERIDIS, Stelios; TOKUNAGA, Takenobu (Eds.). *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA), May 2018. ISBN 979-10-95546-00-9.

CAÑETE, José; CHAPERON, Gabriel; FUENTES, Rodrigo; HO, Jou-Hui; KANG, Hojin; PÉREZ, Jorge. Spanish Pre-Trained BERT Model and Evaluation Data. In: PML4DC at ICLR 2020. [S.l.: s.n.], 2020.

CHALKIDIS, Ilias; FERGADIOTIS, Manos; MALAKASIOTIS, Prodromos; ALETRAS, Nikolaos; ANDROUTSOPOULOS, Ion. LEGAL-BERT: The Muppets straight out of Law School. In: FINDINGS of the Association for Computational Linguistics: EMNLP 2020. Online: Association for Computational Linguistics, Nov. 2020. P. 2898–2904. DOI: 10.18653/v1/2020.findings-emnlp.261. Available from: <https://aclanthology.org/2020.findings-emnlp.261>.

CHAN, Branden; SCHWETER, Stefan; MÖLLER, Timo. German's Next Language Model. In: PROCEEDINGS of the 28th International Conference on Computational Linguistics. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020. P. 6788–6796. DOI: 10.18653/v1/2020.coling-main.598. Available from: <https://aclanthology.org/2020.coling-main.598>.

CHELBA, Ciprian; MIKOLOV, Tomas; SCHUSTER, Mike; GE, Qi; BRANTS, Thorsten; KOEHN, Phillipp; ROBINSON, Tony. *One Billion Word Benchmark for Measuring Progress in Statistical Language Modeling*. [S.l.], 2013. Available from: <http://arxiv.org/abs/1312.3005>.

CONNEAU, Alexis; KHANDELWAL, Kartikay; GOYAL, Naman; CHAUDHARY, Vishrav; WENZEK, Guillaume; GUZMÁN, Francisco; GRAVE, Edouard; OTT, Myle; ZETTLEMOYER, Luke; STOYANOV, Veselin. Unsupervised Cross-lingual Representation Learning at Scale. In: PROCEEDINGS of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, July 2020. P. 8440–8451. DOI: 10.18653/v1/2020.acl-main.747. Available from: <https://aclanthology.org/2020.acl-main.747>.

DATA REPORTAL. *Digital 2021: Local Country Headlines*. [S.l.: s.n.], 2021. https://datareportal.com/reports/digital-2021-local-country-headlines. Accessed: 2021-10-30.

DEMSZKY, Dorottya; MOVSHOVITZ-ATTIAS, Dana; KO, Jeongwoo; COWEN, Alan; NEMADE, Gaurav; RAVI, Sujith. GoEmotions: A Dataset of Fine-Grained Emotions. In: 58TH Annual Meeting of the Association for Computational Linguistics (ACL). [S.l.: s.n.], 2020.

DEVLIN, Jacob; CHANG, Ming-Wei; LEE, Kenton; TOUTANOVA, Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: BURSTEIN, Jill; DORAN, Christy; SOLORIO, Thamar (Eds.). *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. [S.l.]: Association for Computational Linguistics, 2019. P. 4171–4186. DOI: 10.18653/v1/n19-1423. Available from: <https://doi.org/10.18653/v1/n19-1423>.

EBERHARD, David M.; SIMONS, Gary F.; FENNIG, Charles D. *Ethnologue: Languages of the World*. Twenty-sixth. Dallas, Texas: SIL International, 2023. Available from: <http://www.ethnologue.com>.

GONZÁLEZ, José Ángel; HURTADO, Lluís-F.; PLA, Ferran. TWilBert: Pre-trained Deep Bidirectional Transformers for Spanish Twitter. *Neurocomputing*, 2020. ISSN 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2020.09.078. Available from: <http://www.sciencedirect.com/science/article/pii/S0925231220316180>.

GUO, Yanzhu; RENNARD, Virgile; XYPOLOPOULOS, Christos; VAZIRGIANNIS, Michalis. BERTweetFR : Domain Adaptation of Pre-Trained Language Models for French Tweets. In: XU, Wei; RITTER, Alan; BALDWIN, Tim; RAHIMI, Afshin (Eds.). *Proceedings of the Seventh Workshop on Noisy User-generated Text, W-NUT 2021, Online, November 11, 2021*. [S.l.]: Association for Computational Linguistics, 2021. P. 445–450. DOI:

10.18653/v1/2021.wnut-1.49. Available from: <https://doi.org/10.18653/v1/2021.wnut-1.49>.

GURURANGAN, Suchin; MARASOVIC, Ana; SWAYAMDIPTA, Swabha; LO, Kyle; BELTAGY, Iz; DOWNEY, Doug; SMITH, Noah A. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In: JURAFSKY, Dan; CHAI, Joyce; SCHLUTER, Natalie; TETREAULT, Joel R. (Eds.). *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. [S.l.]: Association for Computational Linguistics, 2020. P. 8342–8360. DOI: 10.18653/v1/2020.acl-main.740. Available from: <https://doi.org/10.18653/v1/2020.acl-main.740>.

HONG, Lichan; CONVERTINO, Gregorio; CHI, Ed H. Language Matters In Twitter: A Large Scale Study. In: ADAMIC, Lada A.; BAEZA-YATES, Ricardo; COUNTS, Scott (Eds.). *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*. [S.l.]: The AAAI Press, 2011. Available from: <http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2856>.

HOWARD, Jeremy; RUDER, Sebastian. Universal Language Model Fine-tuning for Text Classification. In: PROCEEDINGS of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne, Australia: Association for Computational Linguistics, July 2018. P. 328–339. DOI: 10.18653/v1/P18-1031. Available from: <https://aclanthology.org/P18-1031>.

HUERTAS-TATO, Javier; MARTIN, Alejandro; CAMACHO, David. BERTuit: Understanding Spanish language in Twitter through a native transformer. *arXiv preprint arXiv:2204.03465*, 2022.

INTERNET WORLD STATS. *Internet World Users By Language*. [S.l.: s.n.], 2020. https://www.internetworldstats.com/stats7.htm. Accessed: 2021-04-07.

JURAFSKY, Daniel; MARTIN, James H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 1st. USA: Prentice Hall PTR, 2000. ISBN 0130950696.

KOTO, Fajri; LAU, Jey Han; BALDWIN, Timothy. IndoBERTweet: A Pretrained Language Model for Indonesian Twitter with Effective Domain-Specific Vocabulary Initialization. In: PROCEEDINGS of the 2021 Conference on Empirical Methods in Natural Language Processing. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021. P. 10660–10668. DOI: 10.18653/v1/2021.emnlp-main.833. Available from: <https://aclanthology.org/2021.emnlp-main.833>.

KOTO, Fajri; RAHIMI, Afshin; LAU, Jey Han; BALDWIN, Timothy. IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP. In: PROCEEDINGS of the 28th COLING. [S.l.: s.n.], 2020.

LAN, Zhenzhong; CHEN, Mingda; GOODMAN, Sebastian; GIMPEL, Kevin; SHARMA, Piyush; SORICUT, Radu. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In: 8TH International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. [S.l.]: OpenReview.net, 2020. Available from: <https://openreview.net/forum?id=H1eA7AEtvS>.

LE, Hang; VIAL, Loïc; FREJ, Jibril; SEGONNE, Vincent; COAVOUX, Maximin; LECOUTEUX, Benjamin; ALLAUZEN, Alexandre; CRABBÉ, Benoît; BESACIER, Laurent; SCHWAB, Didier. FlauBERT: Unsupervised Language Model Pre-training for French. In: PROCEEDINGS of The 12th Language Resources and Evaluation Conference. Marseille, France: European Language Resources Association, May 2020. P. 2479–2490. Available from: <https://www.aclweb.org/anthology/2020.lrec-1.302>.

LEE, Jinhyuk; YOON, Wonjin; KIM, Sungdong; KIM, Donghyeon; KIM, Sunkyu; SO, Chan Ho; KANG, Jaewoo. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, Oxford University Press, v. 36, n. 4, p. 1234–1240, 2020.

LEWIS, Mike; LIU, Yinhan; GOYAL, Naman; GHAZVININEJAD, Marjan; MOHAMED, Abdelrahman; LEVY, Omer; STOYANOV, Veselin; ZETTLEMOYER, Luke. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In: PROCEEDINGS of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, July 2020. P. 7871–7880. DOI: 10.18653/v1/2020.acl-main.703. Available from: <https://aclanthology.org/2020.acl-main.703>.

LIU, Bing. *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. 2. ed. [S.l.]: Cambridge University Press, 2020. (Studies in Natural Language Processing). DOI: 10.1017/9781108639286.

LOSHCHILOV, Ilya; HUTTER, Frank. Decoupled Weight Decay Regularization. In: INTERNATIONAL Conference on Learning Representations. [S.l.: s.n.], 2019. Available from: <https://openreview.net/forum?id=Bkg6RiCqY7>.

MARTIN, Louis; MULLER, Benjamin; ORTIZ SUÁREZ, Pedro Javier; DUPONT, Yoann; ROMARY, Laurent; CLERGERIE, Éric Villemonte de la; SEDDAH, Djamé; SAGOT, Benoıt.

CamemBERT: a Tasty French Language Model. In: PROCEEDINGS of the 58th Annual Meeting of the Association for Computational Linguistics. [S.l.: s.n.], 2020.

MARTINS, Renato F; PEREIRA, Adriano; BENEVENUTO, Fabrício. An approach to sentiment analysis of web applications in portuguese. In: PROCEEDINGS of the 21st Brazilian Symposium on Multimedia and the Web. [S.l.: s.n.], 2015. P. 105–112.

MERITY, Stephen; XIONG, Caiming; BRADBURY, James; SOCHER, Richard. Pointer Sentinel Mixture Models. In: INTERNATIONAL Conference on Learning Representations. [S.l.: s.n.], 2017. Available from: <https://openreview.net/forum?id=Byj72udxe>.

MIKOLOV, Tomas; GRAVE, Edouard; BOJANOWSKI, Piotr; PUHRSCH, Christian; JOULIN, Armand. Advances in Pre-Training Distributed Word Representations. In: PROCEEDINGS of the International Conference on Language Resources and Evaluation (LREC 2018). [S.l.: s.n.], 2018.

MIKOLOV, Tomas; SUTSKEVER, Ilya; CHEN, Kai; CORRADO, Greg; DEAN, Jeffrey. Distributed Representations of Words and Phrases and Their Compositionality. In: PROCEEDINGS of the 26th International Conference on Neural Information Processing Systems - Volume 2. Lake Tahoe, Nevada: [s.n.], 2013. (NIPS'13), p. 3111–3119.

MORAES, Silvia MW; SANTOS, André LL; REDECKER, Matheus; MACHADO, Rackel M; MENEGUZZI, Felipe R. Comparing approaches to subjectivity classification: A study on portuguese tweets. In: SPRINGER. COMPUTATIONAL Processing of the Portuguese Language: 12th International Conference, PROPOR 2016, Tomar, Portugal, July 13-15, 2016, Proceedings 12. [S.l.: s.n.], 2016. P. 86–94.

NGUYEN, Dat Quoc; NGUYEN, Anh Tuan. PhoBERT: Pre-trained language models for Vietnamese. In: FINDINGS of the Association for Computational Linguistics: EMNLP 2020. [S.l.: s.n.], 2020. P. 1037–1042.

NGUYEN, Dat Quoc; VU, Thanh; NGUYEN, Anh Tuan. BERTweet: A pre-trained language model for English Tweets. In: PROCEEDINGS of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. [S.l.: s.n.], 2020. P. 9–14.

OLIVEIRA CAROSIA, Arthur Emanuel de; COELHO, Guilherme Palermo; SILVA, Ana Estela Antunes da. Analyzing the Brazilian Financial Market through Portuguese Sentiment Analysis in Social Media. *Applied Artificial Intelligence*, v. 34, p. 1–19, 2020.

OPENAI. *GPT-4 Technical Report*. [S.l.: s.n.], 2023. arXiv: 2303.08774 [cs.CL].

PASZKE, Adam; GROSS, Sam; MASSA, Francisco; LERER, Adam; BRADBURY, James; CHANAN, Gregory; KILLEEN, Trevor; LIN, Zeming; GIMELSHEIN, Natalia; ANTIGA, Luca; DESMAISON, Alban; KOPF, Andreas; YANG, Edward; DEVITO, Zachary; RAISON, Martin; TEJANI, Alykhan; CHILAMKURTHY, Sasank; STEINER, Benoit; FANG, Lu; BAI, Junjie; CHINTALA, Soumith. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: WALLACH, H.; LAROCHELLE, H.; BEYGELZIMER, A.; D'ALCHÉ-BUC, F.; FOX, E.; GARNETT, R. (Eds.). *Advances in Neural Information Processing Systems 32*. [S.l.]: Curran Associates, Inc., 2019. P. 8024–8035. Available from: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011.

PENNINGTON, Jeffrey; SOCHER, Richard; MANNING, Christopher. GloVe: Global Vectors for Word Representation. In: PROCEEDINGS of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar: Association for Computational Linguistics, Oct. 2014. P. 1532–1543. DOI: 10.3115/v1/D14-1162. Available from: <https://aclanthology.org/D14-1162>.

PETERS, Matthew E.; NEUMANN, Mark; IYYER, Mohit; GARDNER, Matt; CLARK, Christopher; LEE, Kenton; ZETTLEMOYER, Luke. Deep Contextualized Word Representations. In: PROCEEDINGS of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). New Orleans, Louisiana: Association for Computational Linguistics, June 2018. P. 2227–2237. DOI: 10.18653/v1/N18-1202. Available from: <https://aclanthology.org/N18-1202>.

POLIGNANO, Marco; BASILE, Pierpaolo; DE GEMMIS, Marco; SEMERARO, Giovanni; BASILE, Valerio. Alberto: Italian BERT language understanding model for NLP challenging tasks based on tweets. In: CEUR. 6TH Italian Conference on Computational Linguistics, CLiC-it 2019. [S.l.: s.n.], 2019. v. 2481, p. 1–6.

RADFORD, Alec; NARASIMHAN, Karthik; SALIMANS, Tim; SUTSKEVER, Ilya. Improving language understanding by generative pre-training, 2018.

RAFFEL, Colin; SHAZEER, Noam; ROBERTS, Adam; LEE, Katherine; NARANG, Sharan; MATENA, Michael; ZHOU, Yanqi; LI, Wei; LIU, Peter J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.*, JMLR.org, v. 21, n. 1, Jan. 2020. ISSN 1532-4435.

RUDER, Sebastian. *Neural transfer learning for natural language processing*. 2019. PhD thesis – NUI Galway.

SALTON, Gerard; WONG, Anita; YANG, Chung-Shu. A vector space model for automatic indexing. *Communications of the ACM*, ACM New York, NY, USA, v. 18, n. 11, p. 613–620, 1975.

SANH, Victor; DEBUT, Lysandre; CHAUMOND, Julien; WOLF, Thomas. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

SANTOS, Jéssica Soares dos; BERNARDINI, Flávia Cristina; PAES, A. A survey on the use of data and opinion mining in social media to political electoral outcomes prediction. *Social Network Analysis and Mining*, v. 11, p. 1–39, 2021.

SENNRICH, Rico; HADDOW, Barry; BIRCH, Alexandra. Neural Machine Translation of Rare Words with Subword Units. In: PROCEEDINGS of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Berlin, Germany: Association for Computational Linguistics, Aug. 2016. P. 1715–1725. DOI: 10.18653/v1/P16-1162. Available from: <https://aclanthology.org/P16-1162>.

SOUZA, Fábio; NOGUEIRA, Rodrigo; LOTUFO, Roberto. BERTimbau: pretrained BERT models for Brazilian Portuguese. In: 9TH Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear). [S.l.: s.n.], 2020.

SOUZA, Karine França de; PEREIRA, Moisés Henrique Ramos; DALIP, Daniel Hasan. Unilex: Método léxico para análise de sentimentos textuais sobre conteúdo de tweets em português brasileiro. *Abakós*, v. 5, n. 2, p. 79–96, 2017.

STATISTA. *Leading countries based on number of Twitter users as of July 2021*. [S.l.: s.n.], 2021. https://www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries. Accessed: 2021-10-30.

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.

TOUVRON, Hugo; LAVRIL, Thibaut; IZACARD, Gautier; MARTINET, Xavier; LACHAUX, Marie-Anne; LACROIX, Timothée; ROZIÈRE, Baptiste; GOYAL, Naman; HAMBRO, Eric; AZHAR, Faisal; RODRIGUEZ, Aurelien; JOULIN, Armand; GRAVE, Edouard; LAMPLE, Guillaume. *LLaMA: Open and Efficient Foundation Language Models*. [S.l.: s.n.], 2023. arXiv: 2302.13971 [cs.CL].

VAN DER MAATEN, Laurens. Accelerating T-SNE Using Tree-Based Algorithms. *J. Mach. Learn. Res.*, JMLR.org, v. 15, n. 1, p. 3221–3245, Jan. 2014. ISSN 1532-4435.

VASWANI, Ashish; SHAZEER, Noam; PARMAR, Niki; USZKOREIT, Jakob; JONES, Llion; GOMEZ, Aidan N.; KAISER, Lukasz; POLOSUKHIN, Illia. Attention is All you Need. In: GUYON, Isabelle; LUXBURG, Ulrike von; BENGIO, Samy; WALLACH, Hanna M.; FERGUS, Rob; VISHWANATHAN, S. V. N.; GARNETT, Roman (Eds.). *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. [S.l.: s.n.], 2017. P. 5998–6008. Available from: <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html>.

VIANNA, Daniela; CARNEIRO, Fernando; CARVALHO, Jonnathan; PLASTINO, Alexandre; PAES, Aline. Sentiment analysis in Portuguese tweets: an evaluation of diverse word representation models. *Language Resources and Evaluation*, Springer, p. 1–50, 2023.

WAGNER FILHO, Jorge A.; WILKENS, Rodrigo; IDIART, Marco; VILLAVICENCIO, Aline. The brWaC Corpus: A New Open Resource for Brazilian Portuguese. In: PROCEEDINGS of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). Miyazaki, Japan: European Language Resources Association (ELRA), May 2018. Available from: <https://aclanthology.org/L18-1686>.

WANG, Alex; PRUKSACHATKUN, Yada; NANGIA, Nikita; SINGH, Amanpreet; MICHAEL, Julian; HILL, Felix; LEVY, Omer; BOWMAN, Samuel. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. In: WALLACH, H.; LAROCHELLE, H.; BEYGELZIMER, A.; D'ALCHÉ-BUC, F.; FOX, E.; GARNETT, R. (Eds.). *Advances in Neural Information Processing Systems*. [S.l.]: Curran Associates, Inc., 2019. v. 32. Available from: <https://proceedings.neurips.cc/paper/2019/file/4496bf24afe7fab6f046bf4923da8de6-Paper.pdf>.

WANG, Alex; SINGH, Amanpreet; MICHAEL, Julian; HILL, Felix; LEVY, Omer; BOWMAN, Samuel. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In: PROCEEDINGS of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Brussels, Belgium: Association

for Computational Linguistics, Nov. 2018. P. 353–355. DOI: 10.18653/v1/W18-5446. Available from: <https://aclanthology.org/W18-5446>.

WOLF, Thomas; DEBUT, Lysandre; SANH, Victor; CHAUMOND, Julien; DELANGUE, Clement; MOI, Anthony; CISTAC, Pierric; RAULT, Tim; LOUF, Rémi; FUNTOWICZ, Morgan; DAVISON, Joe; SHLEIFER, Sam; PLATEN, Patrick von; MA, Clara; JERNITE, Yacine; PLU, Julien; XU, Canwen; SCAO, Teven Le; GUGGER, Sylvain; DRAME, Mariama; LHOEST, Quentin; RUSH, Alexander M. Transformers: State-of-the-Art Natural Language Processing. In: PROCEEDINGS of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Online: Association for Computational Linguistics, Oct. 2020. P. 38–45. Available from: <https://www.aclweb.org/anthology/2020.emnlp-demos.6>.

WORKSHOP, BigScience et al. *BLOOM: A 176B-Parameter Open-Access Multilingual Language Model*. [S.l.: s.n.], 2023. arXiv: 2211.05100 [cs.CL].

WU, Yonghui; SCHUSTER, Mike; CHEN, Zhifeng; LE, Quoc V.; NOROUZI, Mohammad; MACHEREY, Wolfgang; KRIKUN, Maxim; CAO, Yuan; GAO, Qin; MACHEREY, Klaus; KLINGNER, Jeff; SHAH, Apurva; JOHNSON, Melvin; LIU, Xiaobing; KAISER, Lukasz; GOUWS, Stephan; KATO, Yoshikiyo; KUDO, Taku; KAZAWA, Hideto; STEVENS, Keith; KURIAN, George; PATIL, Nishant; WANG, Wei; YOUNG, Cliff; SMITH, Jason; RIESA, Jason; RUDNICK, Alex; VINYALS, Oriol; CORRADO, Greg; HUGHES, Macduff; DEAN, Jeffrey. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *CoRR*, abs/1609.08144, 2016. arXiv: 1609.08144. Available from: <http://arxiv.org/abs/1609.08144>.

ZHU, Y.; KIROS, R.; ZEMEL, R.; SALAKHUTDINOV, R.; URTASUN, R.; TORRALBA, A.; FIDLER, S. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books. In: 2015 IEEE International Conference on Computer Vision (ICCV). Los Alamitos, CA, USA: IEEE Computer Society, Dec. 2015. P. 19–27. DOI: 10.1109/ICCV.2015.11. Available from: <https://doi.ieeecomputersociety.org/10.1109/ICCV.2015.11>.

ZHUANG, Liu; WAYNE, Lin; YA, Shi; JUN, Zhao. A Robustly Optimized BERT Pre-training Approach with Post-training. English. In: PROCEEDINGS of the 20th Chinese National Conference on Computational Linguistics. Huhhot, China: Chinese Information Processing Society of China, Aug. 2021. P. 1218–1227. Available from: <https://aclanthology.org/2021.ccl-1.108>.

# APPENDIX A – Datasets

Despite Portuguese being among the top ten languages utilized on the Internet as of January 2020[1], there remains a scarcity of resources for sentiment analysis in this language. Motivated by the necessity to assemble a comprehensive Portuguese corpus, we explored the literature for annotated resources pertaining to sentiment analysis and successfully acquired eight datasets by contacting authors. These datasets, comprising human-annotated tweets, exhibit diversity in subject matter, with three being binary and five being multiclass. The labeled datasets can be accessed at https://bityli.com/RvhFax, and we have also made the collection available in the transformers datasets library[2].

## A.1 OPCovid-BR

The OPCovid-BR dataset consists of 600 manually labeled tweets about the Covid-19 pandemic posted by Brazilian Twitter users. This is the smallest dataset in our collection but also the one with the longest tweets (approximately 28 words on average, almost double the second dataset in this regard). The authors developed a Twitter API to extract tweets using the key term search: "coronavirus". The OPCovid-BR were annotated by three annotators, with concordance among annotators equal to 82,77%. It is annotated with the binary document polarity (positive and negative) and fine-grained opinion (explicit aspects) for each tweet. There are 300 tweets with positive labels and 300 tweets with negative labels. For simplicity, in this paper, we refer to this dataset as ***covidbr***. This dataset is available in the transformers datasets library at https://huggingface.co/datasets/melll-uff/opcovidbr.

[1] https://www.statista.com/statistics/262946/share-of-the-most-common-languages-on-the-internet/

[2] https://huggingface.co/melll-uff/

## A.2 TweetSentBR

The dataset TweetSentBR was created in 2017 by (BRUM; GRAÇAS VOLPE NUNES, 2018) and is a corpus of tweets in Brazilian Portuguese in the domain of TV shows. This is a reasonably unbalanced dataset that contains the shortest tweets on average (approximately 11 words in each), and more than 40% of the rows have less than 10 words. Virtually all tweets have at least one occurrence of hashtags. TweetSentBR is also the dataset with the highest frequency of emojis (22.56%). The dataset was labeled by several annotators following steps established in the literature to improve the reliability of the Sentiment Analysis task and each tweet was annotated as either positive or negative. Several annotators labeled each tweet as positive or negative following an annotation process of eight steps. The final label for each tweet was determined based on a major voting strategy. While some tweets were labeled by only one annotator, others were annotated by three or seven. For simplicity, we refer to this dataset as ***sentbr*** in this manuscript. This dataset is available in the transformers datasets library at https://huggingface.co/datasets/melll-uff/tweetsentbr.

## A.3 FIAT-UFMG

FIAT-UFMG is composed of tweets related to the "FIAT" brand, manually labeled as positive or negative. This is the second-largest dataset in our collection. Tweets were extracted by (MARTINS; PEREIRA; BENEVENUTO, 2015) using the Twitter API filtering the year 2012 and messages related to the "Fiat" brand. For simplicity, we refer to this dataset as ***fiat*** in this study. This dataset is available in the transformers datasets library at https://huggingface.co/datasets/melll-uff/fiat-ufmg.

## A.4 narr-PT

The dataset narr-PT contains tweets that have been human-annotated with sentiment labels by three Mechanical Turk workers with the aim to create a multilingual sentiment dataset for the languages English, German, French, and Portuguese. Originally, in this dataset, there are 12,597 positive, neutral, and negative tweets. Particularly for Portuguese, there are 772 tweets. This balanced dataset is the smallest one in the collection we used in this study among the three-classes datasets. For simplicity, we refer to this dataset as ***narrpt*** in this manuscript. This dataset is available in the transformers

datasets library at https://huggingface.co/datasets/melll-uff/narrpt.

## A.5 MiningBR

This dataset contains tweets of the companies with the most number of complaints in the Brazilian Consumer Protection and Defense Program agency (PROCON). At least two annotators manually labeled the tweets from this collection as neutral, positive, or negative according to their sentiment polarity. Because of the nature of its source, this is a highly unbalanced dataset with nearly 65% of the tweets belonging to the negative class and only 9% being positive. For simplicity, we refer to this dataset as ***mining*** in this manuscript. This dataset is available in the transformers datasets library at https://huggingface.co/datasets/melll-uff/miningbr.

## A.6 Computer-BR

This corpus consists of 2,281 tweets extracted in the period from January to September 2015. To build it, (MORAES et al., 2016) used keywords related to computers, such as notebook, analysis, and testing, among others. In the annotation process, four human annotators have defined the polarity of the tweets, three of them participating in the whole process and the fourth deciding the final polarity in cases of disagreement only. It is worth mentioning that three annotators were from the Computer Science area and one of them was from the Linguistics area. To reduce noise, the dataset was normalized prior to publication: they removed (or treated) special characters and hashtags, turned emoticons and hyperlinks into text, and replaced abbreviations and slang with usual expressions, such as "vc" into "você" (you) and "novis" into "novidade" (news). This dataset is also highly unbalanced, having its dominant class *neutral* with approximately 75% of all tweets. For simplicity, we refer to this dataset as ***comput*** in this manuscript. This dataset is available in the transformers datasets library at https://huggingface.co/datasets/melll-uff/computerbr.

## A.7 TweetsMG

TweetsMG is a multiclass dataset collected and labeled by the IT staff of Prodemge MG. It belongs to the domain of Education and Politics in the context of the Brazilian State of

Minas Gerais. This dataset contains lots of duplicated tweets (retweets). For simplicity, we refer to this dataset as ***tweemg*** in this manuscript. This dataset is available in transformers datasets library at https://huggingface.co/datasets/melll-uff/tweetsmg.

## A.8 UniLex

The dataset UniLex, created by (SOUZA; PEREIRA; DALIP, 2017), contains most of the tweets belonging to the domain of politics. Labeling was performed by four annotators, each one labeling approximately 3,500 tweets. Tweets with dates, user mentions, and hashtags were considered neutral. This is the largest dataset of the collection used in this study. For simplicity, we refer to this dataset as ***unilex*** in this manuscript. This dataset is available in the transformers datasets library at https://huggingface.co/datasets/melll-uff/unilex.