Universidade Estadual de Campinas
Instituto de Computação

Gian Franco Joel Condori Luna

Analysis of self-supervised approaches for fine-tuning language models for Portuguese tasks

Análise de abordagens auto-supervisionadas para ajuste fino de modelos de linguagem para tarefas em português

CAMPINAS
2024

Gian Franco Joel Condori Luna

Analysis of self-supervised approaches for fine-tuning language models for Portuguese tasks

Análise de abordagens auto-supervisionadas para ajuste fino de modelos de linguagem para tarefas em português

Dissertação apresentada ao Instituto de Computação da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Dissertation presented to the Institute of Computing of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Computer Science.

Supervisor/Orientador: Prof. Dr. Marcelo da Silva Reis
Co-supervisor/Coorientador: Prof. Dr. Didier Augusto Vega-Oliveros

Este exemplar corresponde à versão final da Dissertação defendida por Gian Franco Joel Condori Luna e orientada pelo Prof. Dr. Marcelo da Silva Reis.

CAMPINAS
2024

Ficha catalográfica
Universidade Estadual de Campinas (UNICAMP)
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

C754a

Condori Luna, Gian Franco Joel, 1989-
Analysis of self-supervised approaches for fine-tuning language models for Portuguese tasks / Gian Franco Joel Condori Luna. – Campinas, SP : [s.n.], 2024.

Orientador(es): Marcelo da Silva Reis.
Coorientador(es): Didier Augusto Vega Oliveros.
Dissertação (mestrado) – Universidade Estadual de Campinas (UNICAMP), Instituto de Computação.

1. Aprendizado auto supervisionado (Aprendizado de máquina). 2. Inteligência artificial. 3. Aprendizado por transferência. 4. Análise de sentimentos. 5. Processamento de linguagem natural (Computação). 6. Adaptação de domínio (Inteligência artificial). I. Reis, Marcelo da Silva, 1979-. II. Vega Oliveros, Didier Augusto, 1984-. III. Universidade Estadual de Campinas (UNICAMP). Instituto de Computação. IV. Título.

Informações complementares

Título em outro idioma: Análise de abordagens auto-supervisionadas para ajuste fino de modelos de linguagem para tarefas em português
Palavras-chave em inglês:
Self-supervised learning (Machine learning)
Artificial intelligence
Transfer learning
Sentiment analysis
Natural language processing (Computer science)
Domain adaptation (Artificial intelligence)
Área de concentração: Ciência da Computação
Titulação: Mestre em Ciência da Computação
Banca examinadora:
Marcelo da Silva Reis [Orientador]
Thiago Alexandre Salgueiro Pardo
André Santanchè
Data de defesa: 11-10-2024
Programa de Pós-Graduação: Ciência da Computação

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0009-0000-6249-5860
- Currículo Lattes do autor: http://lattes.cnpq.br/2367857527051895

Universidade Estadual de Campinas
Instituto de Computação

Gian Franco Joel Condori Luna

Analysis of self-supervised approaches for fine-tuning language models for Portuguese tasks

Análise de abordagens auto-supervisionadas para ajuste fino de modelos de linguagem para tarefas em português

Banca Examinadora:

- Prof. Dr. Marcelo da Silva Reis
  Instituto de Computação - UNICAMP
- Prof. Dr. Thiago Alexandre Salgueiro Pardo
  Instituto de Ciências Matemáticas e de Computação - USP
- Prof. Dr. Andre Santanche
  Instituto de Computação - UNICAMP

A ata da defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria do Programa da Unidade.

Campinas, 11 de outubro de 2024

Acknowledgements

As I conclude this stage of my life, I feel a deep gratitude towards the people who accompanied me on the arduous path to the completion of this dissertation.

First of all, I want to express my most sincere gratitude to my family, whose unconditional support has been the fundamental pillar in my personal and academic development. To my mother, Edith Luna Gamarra, who has been my greatest source of inspiration and motivation. Thanks to her love and dedication, I have learned that, as long as she is by my side, everything is possible.

To my son, Jhoed, for his patience and understanding throughout these years. Although this time of separation was not easy, I am convinced that this sacrifice taught us to value even more the moments we shared. The love of a child is an unstoppable engine, and thanks to it, I find the strength to move forward.

To my sister, Francy, for her trust and respect, always supporting me in every step I take.

To my brother, Cristian and my niece, Maricielo, who, although they may not know it, were a constant source of encouragement in times of doubt. Thinking about them encouraged me to continue and not give up.

To Leydy Llerena, my girlfriend, for being my closest support. Her patience, understanding, and support at every moment of this constant struggle have allowed me to remain firm in my goals. Thank you for always being there, listening to me and motivating me to keep going.

Finally, I thank my advisors, Professor Marcelo da Silva Reis and Professor Didier Vega Oliveros, for their invaluable guidance, dedication, and teachings throughout this entire process. Their feedback, always constructive and demanding, has been fundamental to achieving this goal. Without their academic support, this dissertation would not have been possible.

This project was supported by the Ministry of Science, Technology, and Innovation of Brazil, with resources granted by the Federal Law 8.248 of October 23, 1991, under the PPI-Softex. The project was coordinated by Softex and published as Intelligent agents for mobile platforms based on Cognitive Architecture technology [01245.003479/2024-10].

This study was also financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and a DEAPE (*Diretoria Executiva de Apoio e Permanência Estudantil*) scholarship sponsored by Match<IT>.

Resumo

As organizações muitas vezes enfrentam a limitação de ter uma pequena quantidade de dados rotulados para calibrar e refinar os seus modelos de linguagem (LM, do inglês *language models*) em contextos específicos. Esta escassez de dados anotados traduz-se num desafio significativo para o desenvolvimento e melhoria do LM, uma vez que a qualidade e a quantidade dos dados são fatores críticos no desempenho e generalização do modelo. Por outro lado, a aquisição ou criação de dados rotulados caracteriza-se pela sua elevada exigência em termos de tempo e recursos financeiros; este processo complicado e caro pode representar uma barreira significativa para as organizações, limitando a sua capacidade de implementar soluções eficazes de aprendizagem de máquina adaptadas às suas necessidades específicas. A literatura demonstra que problemas semelhantes foram resolvidos por meio de ajuste fino auto-supervisionado, utilizando diferentes abordagens de pré-treinamento. Todavia, até o nosso conhecimento, inexistia a descrição e a avaliação de protocolos desse tipo de treinamento para LMs em português. Dessa forma, nesta dissertação propomos como adaptar o protocolo de pré-treinamento do LM em português BERTimbau para um procedimento de ajuste fino auto-supervisionado, acompanhado de uma avaliação de como este procedimento pode afetar a generalização e tarefas downstream quando se tem dados não rotulados. Realizamos vários experimentos com três conjuntos de dados de diferentes contextos, nos quais descongelamos diferentes números de camadas no modelo e utilizamos diferentes ajustes na taxa de aprendizagem, determinando assim um regime de treinamento ideal para o protocolo de ajuste fino auto-supervisionado. Os resultados utilizando análise de sentimentos como tarefa downstream, com dados rotulados dos mesmos conjuntos de dados, indicaram que descongelar apenas a última camada já traz bons resultados, o que permitiria usuários com recursos computacionais limitados obterem ótimos resultados com o método. Além disso, foi destacada a eficácia do ajuste fino auto-supervisionado em conjuntos de dados maiores, sugerindo o seu potencial para pesquisas futuras em LMs pré-treinados mais avançados.

Abstract

Organizations often face the limitation of having a small amount of labeled data to calibrate and refine their language models (LMs) in specific contexts. This scarcity of annotated data translates into a significant challenge for the development and improvement of LMs, since the quality and quantity of data are critical factors in the performance and generalization of the model. On the other hand, the acquisition or creation of labeled data is characterized by its high demand in terms of time and financial resources; this complicated and expensive process can represent a significant barrier for organizations, limiting their ability to implement effective machine learning solutions tailored to their specific needs. The literature shows that similar problems have been solved through self-supervised fine-tuning using different pre-training approaches. However, to our knowledge, there was no description and evaluation of such training protocols for LMs in Portuguese. Thus, in this dissertation, we propose how to adapt the BERTimbau Portuguese LM pre-training protocol to a self-supervised fine-tuning procedure, accompanied by an evaluation of how this procedure can affect generalization and downstream tasks when using unlabeled data. We performed several experiments with three datasets from different contexts, in which we unfroze different numbers of layers in the model and used different learning rate settings, thus determining an optimal training regime for the self-supervised fine-tuning protocol. The results using sentiment analysis as a downstream task, with labeled data from the same datasets, indicated that unfreezing only the last layer already yields good results, which allows users with limited computational resources to obtain excellent results with the method. Furthermore, the effectiveness of self-supervised fine-tuning on larger datasets was highlighted, suggesting its potential for future research in more advanced pre-trained LMs.

Keywords: Self-supervised learning, Fine-tuning, Artificial intelligence, Sentiment analysis, Natural language processing, Transfer learning, Domain adaptation.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

In recent decades, the artificial intelligence (AI) field has witnessed remarkable technological advancements that influence numerous facets of modern life. AI is now deeply integrated into everyday applications, ranging from recommendation systems on streaming platforms to the development of autonomous vehicles, fundamentally reshaping interactions with both the digital and physical world. Central to many of these advancements is machine learning (ML), a subset of AI that enables systems to autonomously identify patterns and execute tasks without explicit programming. ML has demonstrated effectiveness across various domains, including email classification for spam detection [41] and multi-task transfer learning for predicting entity modifiers in clinical text processing [1].

Supervised learning (SL) has long been a fundamental tool for training models in specific domains or tasks within the ML paradigm. In SL, models are trained using a labeled data set, where each training example is associated with a label indicating the desired output. This technique has proven to be highly effective in solving classification and regression problems, allowing models to learn to generalize patterns from labeled examples. For instance, in sentiment analysis, supervised learning can successfully classify simple cases, such as identifying a positive sentiment in a review like "I loved the product!". However, it struggles with more complex cases, such as detecting sarcasm in a review like "Oh great, another feature that doesn't work!" or interpreting mixed sentiments within the same sentence, as in "The design is stunning, but the battery life is disappointing".

However, despite its successes, SL faces a fundamental challenge: the scarcity and cost of acquiring labeled data. In real-world applications, having a few labeled data is the only resource available. This is related to the limited availability of experts, a lack of data sets with wide diversity, or situations where large amounts of unlabeled data are generated but manual labeling is impractical [54]. For example, labeling sentences for sarcasm detection in sentiment analysis requires human annotators with advanced linguistic skills, which increases both the time and cost of data preparation.

The scarcity of labeled data has led to the development of transfer learning, a technique that allows leveraging limited labeled data and unlabeled data to improve model performance. Regarding language models, the transfer learning is often achieved in two ways: first, a pretrained language model trained on a large generic unlabeled dataset is used in a self-supervised approach to learn useful feature representations from the data. In the second stage, the labeled data is used to fine tune the pretrained model. The SL

step exposes the model to the domain of the specific context of the subsequent task being addressed, often improving the model's performance in solving the problem.

Although traditional transfer learning often works well even when there are only a few labeled examples [83], the improvements in language model performance on the downstream task usually rely on collecting more labeled examples, so we can say that the more labeled data we have, the better the result. Traditional transfer learning will not give optimal results for a problem with a small amount of labeled data for the downstream task. However, it is often costly for companies to obtain and label data.

Therefore, an alternative approach involves the use of transfer learning. This strategy entails leveraging a language model pre-trained on vast amounts of generic, unlabeled data, followed by self-supervised fine-tuning with domain-specific unlabeled data. The objective is to enable the model to specialize in the domain of interest. Subsequently, the model can be applied to downstream tasks using the limited available labeled data from the same domain, hence achieving better results [43, 29, 79]. This approach offers several key advantages, including reduced dependence on labeled data, which lowers data acquisition costs. Furthermore, by utilizing a larger volume of unlabeled data, models can learn richer and more generalizable representations, often resulting in improved performance across various tasks.

Although there are currently a number of self-supervised fine-tuning techniques proposed in the literature [25, 29, 43], to the best of our knowledge there is no systematic investigation of the application of a self-supervised fune tuning protocol for Portuguese language for domain adaptation in areas such as information technology or e-commerce. For example, in the context of e-commerce, this approach could help models analyze customer feedback, from straightforward comments like "Fast delivery and great service!" to more nuanced and challenging cases, such as detecting sarcastic remarks like "Oh, another 'premium' product that breaks in a week". Therefore, to address such study would be a timely endeavor.

1.1 The Proposed Approach

Our proposed approach to addressing the domain specialization problem with few labeled data by exploring unstructured textual data, for specific domains such as e-commerce in Portuguese language, is to perform self-supervised fine-tuning of a pretrained language model on general Portuguese textual data to give it domain-specific knowledge, also in Portuguese, and then test its performance on a downstream task with a small amount of labeled data. To this end, we use BERTimbau [78], a language model based on BERT [15], trained on a Portuguese corpus. BERTimbau is the model to which we perform a self-supervised fine-tuning using the B2W [65], UTLC-apps and UTLC-movies [77] datasets; which are datasets from different domains in Brazilian Portuguese.

Self-supervised fine-tuning involves retraining a pre-trained model with data from a specific domain using pretext tasks; in the case of BERTimbau, it was used masked language modeling (MLM) and next sentence prediction (NSP). The MLM task masks a percentage of words in a sentence and requires the model to predict them, which helps

the model better understand the sentence's context. On the other hand, the NSP task takes two sentences as input and determines whether they are related, teaching the model to understand coherence and logical sequence in sentences. This self-supervised fine-tuning stage would be essential for adapting the model to its general knowledge to a more specialized context, thereby enhancing its generalization capacity and accuracy in tasks specific to that domain [27]. Once self-supervised fine-tuning is complete, the model is ready to be applied to specific downstream tasks. This type of task allows to explore and validate the improvements obtained during training, demonstrating how adapting the model to a specialized domain can bring significant gains in practical applications.

As the downstream task of our proposal, we defined to use multiclass sentiment analysis. This task significantly extends traditional sentiment analysis by allowing the classification of texts into more than two categories. This capability is crucial in many contexts, such as the entertainment industry, where understanding the diversity of opinions expressed about a film may require identifying specific emotions beyond the merely positive or negative ones. Similarly, understanding the full range of opinions on a topic in social network analysis is critical to informed decision-making. For this reason, this downstream task not only allows for a more detailed and accurate assessment of emotions in texts, but also highlights the importance of capturing the complexity and subtlety of human opinions in diverse contexts.

By applying the retrained model on a downstream task with labeled data from the same domain, we would expect an increase in task performance due to the improvement obtained from the previous stages. More specifically, having been fine-tuned on domain-specific data, the model would be better prepared to handle the complexities and nuances of the task, resulting in a substantial improvement in performance and accuracy. Therefore, this approach would ensure that the model not only has an understanding of general language, but also exceptional adaptability to domain specificity, thus achieving optimal results with a relatively smaller amount of labeled data.

1.2 Research Questions

The research questions (RQs) that we aim to answer are:

RQ 1 How can the pretraining protocol of BERTimbau be adapted for a self-supervised fine-tuning procedure on this model?

RQ 2 How does using unlabeled data in a specific domain in the proposed self-supervised fine-tuning approach impact BERTimbau's downstream performance and its generalization ability?

RQ 3 In which scenarios (*e.g.*, data availability, type of domain) is it suitable to use that self-supervised fine-tuning approach?

1.3 Outline of this Dissertation

After this Introduction, the remainder of this Masters' dissertation is organized as follows:

In Chapter 2, we present the fundamental concepts necessary for a comprehensive understanding of the topics addressed in this dissertation. This includes definitions and conceptual frameworks that form the basis of our research.

In Chapter 3, we conduct a comprehensive analysis of the current state of the literature in relation to our research proposal. This review critically examines existing studies, identifying gaps and opportunities that our work seeks to address.

In Chapter 4, we explain in detail the methodology adopted to answer the proposed research questions. We describe the research design, data collection, and analysis techniques used, ensuring a rigorous approach in our research.

In Chapter 5, we analyze the results obtained from our experiments. We interpret these findings in the context of the research questions by assessing the validity and implications of the results and their possible limitations.

And finally in Chapter 6, we present our conclusions and discuss possible directions for future works. This chapter summarizes the main contributions of our research and suggests areas where further research could be fruitful.

Chapter 2

Fundamental Concepts

This chapter presents a comprehensive theoretical framework, intended to establish the context necessary for understanding the following chapters. We begin by providing an overview of the fundamental principles of artificial intelligence (AI) and its subfields relevant to this work, including machine learning, natural language processing, and how transformer-based models learn in a self-supervised manner. In the sequence, we will present some fundamentals of natural language processing (NLP), including sentiment analysis and language models. Finally, we will present some concepts of learning a transformer model, describing transfer learning, fine-tuning and self-supervised fine-tuning.

2.1 Machine Learning

Machine learning (ML) is a subfield of Artificial Intelligence (AI) that focuses on developing algorithms and models that allow computers to learn patterns and perform specific tasks without requiring explicit programming. According to Mitchell (1997) [50], machine learning is defined as "the study of algorithms that automatically improve their performance through experience". This definition highlights the fundamental aspect of machine learning: the ability of systems to learn and adapt from data, rather than relying exclusively on predefined rules.

One of the most successful approaches in ML is the Deep Neural Networks. As noted by Goodfellow *et al.*(2016) [20], deep neural networks are machine learning models composed of multiple layers of interconnected nodes, which can learn hierarchical representations of input data. These models have performed exceptionally in various tasks, from image recognition to machine translation. We will detail them further in the following section.

2.1.1 Deep Neural Networks (Deep Learning)

Neural networks are computational models inspired by the functioning of the human brain. They have proven to be powerful tools for AI applications. Over time, neural networks have evolved from their early simple forms to more complex and sophisticated models driven by advanced algorithms and access to large data sets.

One of the early milestones in the development of neural networks is the perceptron, proposed by Rosenblatt (1958) [69]. The perceptron is a supervised learning model that

can be used for binary classification, and its ability to learn from data made it one of the first machine learning models.

Over time, neural networks have evolved into more complex models such as Multilayer Perceptron [71], Hopfield Networks [26], Time Delay Neural Networks [82], and Echo State Networks [30], among others. More recently, neural networks have experienced a renaissance with the advent of deep neural networks. As noted by Goodfellow *et al.*(2016) [20], deep neural networks are capable of learning hierarchical representations of data, making them ideal for natural language processing tasks, speech recognition, and many other applications.

Deep Learning is then a subfield of ML that focuses on training deep neural networks to learn hierarchical representations of data [20]. Unlike traditional ML models, deep neural networks comprise multiple layers of interconnected nodes, allowing them to capture and learn complex patterns and abstractions from input data.

We have, for instance, the Convolutional neural networks (CNN) and recurrent neural networks (RNN). According to LeCun *et al.* (2015) [40], CNNs have proven to be especially effective in computer vision tasks such as object recognition and image segmentation. On the other hand, RNNs, as Hochreiter and Schmidhuber (1997) [24] point out, are ideal for processing data sequences, such as text or time series.

One of the distinguishing features of Deep Learning is its ability to automatically learn relevant data features from large data sets. As noted by LeCun *et al.* (2015) [40], "Deep Learning has proven to be especially effective in tasks where large amounts of data are available for training". This has led to significant advances in various areas, including image recognition, natural language processing, and computer vision.

2.1.2 Transformers

Transformers are a neural network architecture designed to handle sequential and contextual data, introduced by Vaswani *et al.* (2017) [80] in the seminal paper "Attention is All You Need". This architecture revolutionized the field of natural language processing (NLP) by overcoming the limitations of previous recurrent models, such as recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), by implementing the self-attention mechanism, allowing to capture long-range dependencies in data sequences more efficiently [80].

The main innovation of Transformers lies in their attention mechanism, which allows the model to focus on different parts of the input sequence dynamically. This approach eliminates the need to process sequential data in a strictly ordered manner, as is the case with RNNs and LSTMs, allowing for more efficient parallelism and significantly reducing training time. In particular, the multi-head self-attention mechanism allows the model to capture different types of contextual relationships at multiple levels of granularity [80].

The Transformers architecture consists of an encoder and a decoder, each one with a series of layers. Each encoding layer takes an input sequence and applies self-attention mechanisms and fully connected feed-forward layers. In contrast, each decoding layer incorporates attention mechanisms to both the input and the output generated up to that point, facilitating the translation of sequences [80]. This modular and scalable structure

has proven to be extremely effective in various NLP tasks. The figure of the architecture of a Transformer can be observed in Figure 2.1.

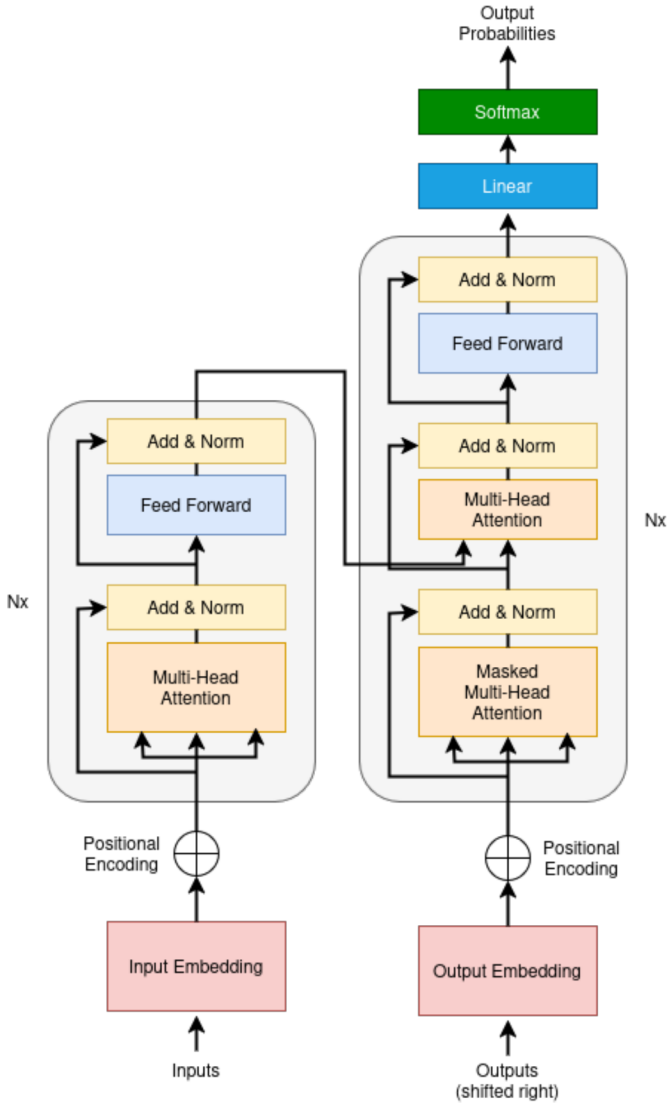


Figure 2.1: Architecture of a Transformer, adapted from Vaswani *et al.* (2017) [80].

Transformers have been the basis of numerous advanced NLP models. Pretrained models such as BERT [15], RoBERTa [46], and GPT [62] all rely on variants of the Transformer architecture, tailored and optimized for specific tasks. These models have set new performance standards in multiple NLP benchmarks, demonstrating the flexibility and power of Transformers to effectively capture and represent linguistic context [15, 46].

Transformers have fundamentally changed the way stream processing tasks are approached in machine learning. Their ability to handle long-term dependencies efficiently, combined with their parallelizable and modular structure, has enabled significant advances in multiple fields, cementing their position as an essential tool in the contemporary machine-learning toolbox.

2.1.3 BERT (Bidirectional Encoder Representations from Transformers)

BERT, or Bidirectional Encoder Representations from Transformers, is a pretrained language model developed by Devlin *et al.* (2018) [15] that has marked a significant milestone in the field of natural language processing (NLP). BERT is distinguished by its ability to understand the context of a word in both directions, forward and backward, within a text sequence. This bidirectionality is one of the key innovations that allows BERT to capture complex contextual relationships and significantly improve performance on various NLP tasks [15].

The BERT model is based on the Transformer architecture, specifically on the encoder part. Unlike unidirectional language models, which only consider the context of a word from one direction, BERT uses a bidirectional attention approach that allows for a deeper and more complete understanding of the context [80]. This capability is achieved through pretraining on masked language modeling (MLM) and next-sentence prediction (NSP) tasks. BERT pretraining is performed on large unlabeled text corpora, such as Wikipedia and the BookCorpus, allowing the model to learn general linguistic representations. Once pretrained, BERT can be fine tuned for specific tasks, such as text classification, named entity extraction, question answering, among others. Such fine-tuning can be achieved by adding a final task-specific layer and performing additional training on labeled data [15].

Since its introduction, BERT has inspired a number of spin-off models and improvements, such as RoBERTa, ALBERT [39], DistilBERT [72], BERTimbau [78], among others, which seek to optimize its efficiency and generalization capabilities further. These models have adopted various strategies to reduce computational requirements and improve performance on specific tasks while maintaining BERT's fundamental architecture. BERT represents a revolutionary advance in the field of NLP, providing a powerful and flexible framework for understanding and generating natural language. Its ability to capture bi-directional contexts and its success in various applications have cemented it as an essential tool in the research and development of natural language technologies.

2.2 Natural Language Processing

Natural Language Processing (NLP) is an interdisciplinary field that sits at the intersection of computational linguistics, artificial intelligence, and data science. Its primary goal is to enable computers to understand, interpret, and generate human language effectively [32]. NLP has evolved significantly since its inception, driven by advances in algorithms, machine learning models, and the availability of large data sets.

The origin of NLP dates back to the early days of computing, with the first attempts to develop programs that could understand and generate human language. One of the early milestones was the ELIZA program, developed by Joseph Weizenbaum in the 1960s [85], which simulates a therapeutic conversation using simple word-matching patterns. Although rudimentary compared to today's systems, ELIZA laid the groundwork for future research in the field of NLP. Since then, NLP has seen significant advances in various areas, driven by the exponential growth of digital data and the development of machine

learning techniques.

NLP encompasses various tasks and applications that allow machines to interact with human language in diverse ways. These tasks include sentiment analysis, text classification, machine translation, text summarization, named entity recognition, and natural language generation. Sentiment analysis, in particular, has evolved to include advanced techniques such as multiclass sentiment analysis, which goes beyond simple categories of positive, negative, or neutral, allowing for the identification of a broader range of emotions, like anger, happiness, sadness, fear, excitement, and others [66]. This capability is crucial for applications in areas such as customer service, where it is important to understand emotional nuances to improve the user experience, or in social media analysis, where understanding the diversity of opinions can influence strategic decision-making.

In summary, NLP has come a long way from its rudimentary beginnings, evolving into a robust and multifaceted field that takes advantage of advances in artificial intelligence and machine learning. As technology advances and digital data expands, NLP is likely to play an increasingly crucial role in various applications, from human-computer interaction to automating complex text analysis tasks. The ability to understand and generate human language not only improves operational efficiency, but also opens up new possibilities for innovation across multiple disciplines.

2.2.1 Sentiment Analysis

Sentiment analysis or opinion mining is the computational study of people's opinions, feelings, emotions, and attitudes toward entities such as products, services, issues, events, topics, and their attributes; as such, sentiment analysis can capture public sentiment about a particular entity to create actionable knowledge through it [44]. This field has gained increasing importance with the proliferation of digital platforms that generate vast volumes of textual data on a daily basis, such as product reviews, online forums, and social media [60].

Traditionally, sentiment analysis has been approached using a binary framework, such as that by Pang and Lee (2005) [56], who developed a binary classification approach to categorize reviews into positive or negative opinions. This method is notable for its simplicity and effectiveness in identifying polarities in short, highly subjective texts and was one of the first to apply machine learning techniques to differentiate between two classes of sentiments.

Furthermore, studies have been conducted on social media data, such as the work of Go *et al.* (2009) [18], who used emoticons to automatically label tweets as positive or negative, thus training a binary sentiment analysis model. This methodology allowed leveraging large volumes of unlabeled data, resulting in robust performance in predicting binary polarities on microblogging platforms.

Binary sentiment analysis has also been applied in domains such as restaurant reviews. Zahoor *et al.* (2022) conducted a comparative study of various classification techniques to identify positive and negative polarities in customer reviews, showing that the use of machine learning-based approaches offers a high degree of accuracy [90]. This study highlights the effectiveness of supervised classifiers in capturing binary polarity in texts

containing direct consumer opinions.

While the binary approach to sentiment analysis is suitable in many cases, it is often insufficient to capture the full complexity of human emotions. This methodology may not adequately reflect emotional diversity in contexts where opinions are more nuanced. This has led to the development of more advanced techniques, such as multiclass sentiment analysis.

Multiclass sentiment analysis provides a richer representation of emotions by classifying opinions on a scale that can include multiple categories, such as negative, neutral, and positive, as discussed by Zhang *et al.* (2018) in their article, where they provide an extensive review of sentiment analysis techniques, focusing on the use of deep learning showing state-of-the-art results for various sentiment analysis tasks, highlighting models that allow classifying opinions into more than two classes, such as positive, negative, neutral, among others [91].

This multiclass approach provides a more nuanced assessment of the sentiments expressed in texts, which is especially useful in applications that require a detailed understanding of user opinions. For example, in the realm of product reviews, a scale of 1 to 5 not only indicates whether a review is positive or negative, but also captures the intensity of the sentiment, ranging from very negative to very positive. A similar analysis is presented by Keith *et al.* (2019) [51] in their paper, where they make a significant contribution to multiclass sentiment analysis in the context of scientific article reviews. Keith facilitates the identification of inconsistencies between written evaluations and scores given by reviewers, using a 5-point scale ("very negative", "negative", "neutral", "positive" and "very positive") and supervised classification methods.

Despite its advantages, multiclass sentiment analysis faces several challenges. One of the most significant problems is the inherent ambiguity of natural language, where context and language subtleties can make it difficult to categorize sentiments accurately [57]. Emotional expressions in texts can be subtle and contextual, confusing the task of classification into multiple categories [48]. Furthermore, labeling textual data for training machine learning models is an expensive and laborious task, and human-labeled data can introduce biases that affect model quality.

Multiclass sentiment analysis represents a crucial advancement in the field of NLP, providing more detailed and accurate tools for understanding human emotions expressed in texts. Despite the challenges inherent to this task, recent advances in language models and machine learning techniques offer promising solutions to improve the accuracy of multiclass sentiment analysis and expand its applications in diverse contexts.

2.2.2 Language Models

Language models are machine learning-based systems designed to understand, generate, and manipulate natural language text [80]. These models learn a language's structure and statistical properties from large amounts of textual data, using advanced artificial intelligence techniques such as deep neural networks. The main purpose of these models is to predict the probability of a sequence of words given its preceding history, thus allowing the generation of coherent and contextually relevant text [88].

The evolution of language models has been significant, from early approaches such as n-gram models, which use strings of n-words to predict the next word in a sequence [32], to more sophisticated techniques based on recurrent neural networks (RNNs) [24] and convolutional neural networks (CNNs) [35]. A notable advance in the field of language models has been the introduction of attention networks, which have revolutionized the way long-term dependencies in text are handled [33]. Attention mechanisms allow the model to assess the importance of different text parts when generating or interpreting a sequence, improving the context modeling ability and the coherence of the generated text [80].

In addition, pre-training on large text corpora and fine-tuning on specific tasks have significantly improved the effectiveness of language models. This approach allows models to generate contextually rich representations of words and phrases, improving accuracy and performance in various practical natural language processing applications such as machine translation, text summarization, sentiment analysis, and question answering [27].

In summary, modern language models represent a dynamic and rapidly evolving area of research, with applications ranging from text understanding and generation to the automation of complex text analysis tasks. The combination of large amounts of data, advanced machine learning techniques, and innovative architectures has allowed these models to reach unprecedented levels of performance, positioning them as essential tools in the field of natural language processing.

2.3 Learning a Transformer-Based Model

2.3.1 Self-Supervised Learning

Self-supervised learning (SSL) is an approach in which models are trained explicitly with automatically generated labels. This method allows for the use of large-scale datasets without human annotation, resulting in effective feature learning that can be transferred to multiple tasks in diverse areas [31], including natural language processing, audio analysis, and other machine learning applications.

Gui *et al.* (2023) mention in their article the most commonly addressed pipeline in SSL, in which a pretext task is defined that the model must solve. During this phase, pseudo labels are automatically generated from the unlabeled data, allowing the model to learn useful representations without human annotation [22]. Once the model has been trained on this pretext task, the learned features can be transferred to specific natural language processing (NLP) tasks, such as text classification, sentiment analysis or machine translation, which are commonly called downstream tasks. This transfer process is carried out by fine-tuning a labeled dataset.

One of the earliest applications of SSL is in data representation, where models learn to extract useful features from data without the need for explicit labels. As noted by Chen *et al.*(2020), self-supervised learning has proven to be especially effective in creating text and image representations, where models can learn relevant semantic and visual features from large amounts of unlabeled data [10].

Self-supervised learning has also been applied in recommender systems, as highlighted

in the study by Yu *et al.* (2022), where the progress of research in self-supervised recommendations is analyzed, achieving key findings in self-supervised signal selection demonstrating significant improvements in recommendation quality [89]. It is also applied to graph data [47], video processing [73], and adversarial pretraining of self-supervised deep networks [61], among others.

Another approach to SSL is the use of pretrained language models. According to Brown *et al.*(2020) [6], pretrained language models, such as BERT [15] and GPT [62], have demonstrated good performance on a variety of natural language processing tasks, such as machine translation, text generation, and sentiment analysis. These models are trained on large corpora of unlabeled text, allowing them to capture the structure and context of language effectively.

More recent models have also demonstrated remarkable advances in text generation and language understanding without relying on extensive manual annotations. For example, GPT-4 has scaled text generation capabilities to even higher levels, overcoming limitations seen in earlier versions such as GPT-3 [6]. These types of models have paved the way for advanced applications in machine translation, text summarization, and content generation, standing out for their ability to handle complex tasks effectively and efficiently.

Self-supervised learning with language models has revolutionized the field of natural language processing by allowing machines to learn from large amounts of unlabeled textual data autonomously. This approach has democratized access to advanced linguistic models, significantly improving accuracy and robustness in various practical applications. As research advances, language models are expected to continue to evolve to address more complex challenges and provide innovative solutions in the realm of human language and artificial intelligence.

2.3.2 Fine Tuning

Fine-tuning is a process in which a pre-trained model, which has already been trained on a general task or with unlabeled data, is fine-tuned to a specific task using a labeled dataset. This fine-tuning process involves modifying the model's weights through supervised or self-supervised learning on the target task, allowing the model to generalize better to that specific task. Unlike training a model from scratch, fine-tuning starts from already trained parameters, taking advantage of the richness of previously learned features and patterns, reducing the time and resources required to achieve good performance [19].

One of the first uses of fine-tuning in natural language processing was documented by Howard and Ruder (2018), who showed that fine-tuning pre-trained language models has become established as a standard practice to customize these models to specific tasks, such as text classification or information extraction, showing its effectiveness in a wide variety of domains [27].

Over time, fine-tuning has evolved into more up-to-date approaches, such as incremental fine-tuning and layered fine-tuning. According to Raffel *et al.* (2019), incremental fine-tuning involves initially training only the top layers of a pretrained model before gradually fine-tuning all layers of the model on the specific task [64]. This approach can

be useful to avoid overfitting on small datasets.

More recently, fine-tuning has seen significant advances with the development of continuous adaptive learning techniques. As highlighted by Liu *et al.* (2019), continuous adaptive learning enables the constant updating of a pretrained model as new data is received, making it easier to maintain model performance in changing and real-time environments [45].

Furthermore, the application of self-supervised fine-tuning, an advanced variant of this technique that focuses on fine-tuning pretrained models using large unlabeled data corpora, is observed. This approach allows models to absorb general linguistic and contextual knowledge during initial pretraining and then adapt more efficiently to specific tasks with smaller data sets or particular domains. In summary, fine-tuning has evolved from its initial application in natural language processing to become a fundamental tool in machine learning. This approach has not only facilitated significant advances in model accuracy, but has also paved the way for future innovations in the personalization and continuous optimization of language-based artificial intelligence systems.

2.3.3 Transfer Learning

Transfer learning is an advanced technique in the field of machine learning that allows a pretrained model to be reused in a task or domain different from the one for which it was originally developed. This approach is based on the premise that knowledge acquired during training on a specific task can be transferred and leveraged to improve performance on another related task, especially when the data available for the new task is limited [55].

One of the most prominent applications of transfer learning is in natural language processing. In this context, pretrained language models, such as BERT and GPT, are initially trained on large corpora of unlabeled text to learn rich and generalized linguistic representations, as shown in Figure 2.2. These models are then fine-tuned for specific tasks, such as text classification, sentiment analysis, machine translation, and text generation, thereby achieving significant performance improvements compared to models trained from scratch [27].

Transfer learning can be classified into several categories, depending on the relationship between the tasks and the domains involved. Among the most common are inductive transfer learning, transductive transfer learning, and self-supervised transfer learning. Inductive transfer learning is applied when the source and target tasks are different but related, while transductive transfer learning is used when the tasks are the same but the domains differ. On the other hand, self-supervised transfer learning refers to the transfer of knowledge in contexts where labels for the target tasks are not available [84].

The benefits of transfer learning are numerous. First, it allows the use of large amounts of unlabeled data, facilitating the training of more robust and generalizable models. Second, it reduces the time and computational resources required for training since it starts from a pretrained model that has already captured many linguistic patterns and structures. Finally, transfer learning has proven to be especially useful in scenarios where labeled data is scarce or expensive to obtain, allowing high performance levels to be achieved with reduced data sets [70].

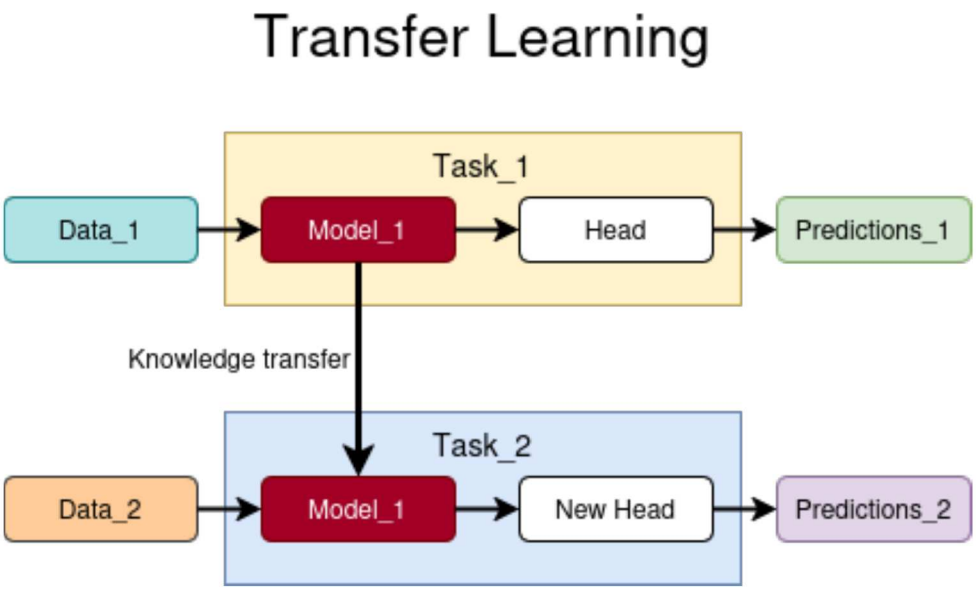


Figure 2.2: Transfer learning allows you to reuse the knowledge acquired when solving a specific problem to address another related problem.
Source:Adapted from Bhavsar (2019) [4].

Transfer learning represents a significant machine learning advancement, providing an efficient and effective methodology for leveraging pretrained models in new tasks and domains. Its ability to improve the performance of models with less data and resources has consolidated its importance in various applications, particularly in natural language processing and other areas of artificial intelligence.

Chapter 3

Related Works

In this chapter, we present and discuss works reported in the literature that are related to this Master's dissertation. Firstly, we detail the methodology adopted for the selection of the most relevant and appropriate scientific articles for this dissertation. Adopting a good methodology not only ensures the quality and relevance of the sources used but also provides a structured framework for the critical review of the existing literature. Subsequently, a comparison of the most important language models in Portuguese will be presented, highlighting their main characteristics. Finally, the various ways in which self-supervised fine-tuning has been implemented in the literature will be explored, providing a detailed analysis of its effectiveness and usefulness in different contexts.

3.1 Systematic Review Method

To carry out the literature review, we used the PRISMA methodology (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) which brings many benefits to this dissertation [53]. PRISMA provides a structured and transparent approach to the selection, evaluation and synthesis of relevant studies, ensuring the thoroughness and rigor of the review process. By following the PRISMA guidelines, the reproducibility and credibility of the research is improved, making it easier for other researchers to replicate the study and verify its findings. In addition, the use of the PRISMA flowchart allows each stage of the paper selection process to be clearly and concisely documented, from the identification of studies to final inclusion, which adds an additional level of transparency and accuracy. These aspects are crucial for a dissertation, as they strengthen the validity and reliability of the research, contributing significantly to the academic quality and impact of the work.

PRISMA guidelines were followed to conduct the literature review, and the corresponding flowchart is presented in Figure 3.1. The search was carried out in the indexers Scopus, ACM Digital Library and Elsevier Science Direct.

First, based on our research questions, we compiled a list of possible sets of keywords that could help us identify the most relevant and representative papers for our dissertation. The sets of keywords are as follows:

- **Domain adaptation:** We selected this keyword because of the need to adapt the

domain to a specific field.

- **Low-resource languages:** These keywords were chosen because the Brazilian Portuguese language has a limited amount of resources compared to other languages.
- **Downstream task:** This was chosen because our dissertation needs to be evaluated on a specific task in order to compare metrics.
- **Unsupervised OR self-supervised:** We selected these keywords because both are used interchangeably in the literature to refer to the same methodology.
- **Text mining:** These keywords were selected because we aim to extract valuable information from large volumes of unstructured textual data for further analysis.
- **Sentiment analysis OR recommender systems:** These keywords were chosen because they are the tasks in which we plan to perform the analysis of the obtained model.

To ensure the timeliness and relevance of our dissertation, we limited our search to papers published between the years 2019 and 2024. Combining our keywords, we generated the following query (example for Scopus):

- **("domain adaptation" OR "low-resource languages" OR "downstream task") AND ("unsupervised" OR "self-supervised") AND "text mining" AND ("sentiment analysis" OR "recommender systems") AND PUBYEAR $>$ 2018 AND PUBYEAR $<$ 2025**

By entering our set of keywords into the selected indexers, we collected a total of 853 papers: 575 from Scopus, 125 from ACM Digital Library, and 153 from Elsevier Science Direct. We then compiled a list of all papers titles to identify and remove duplicates, resulting in the removal of 127 duplicate papers.

We then thoroughly reviewed the titles of the remaining papers, paying particular attention to those that were relevant to our dissertation, which allowed us to exclude an additional 384 papers. This reduced the set to 342 papers. We then evaluated the abstracts and conclusions of these 342 papers, excluding 189 that did not meet our relevance criteria, leaving a total of 153 papers.

Of these 153 articles, we conducted a thorough reading of the content to determine their specific usefulness in the context of our dissertation, excluding 99 articles and leaving us with 54 articles. During the detailed review we identified and added some additional articles that had not been initially mapped but were found to be relevant, adding 21 articles. At the end of the entire process we were left with a total of 75 articles that form the basis of our literature review.

3.2 Language Models in Portuguese

In the field of natural language processing (NLP), there is a notable disparity in the availability and development of language models for different languages. In particular,

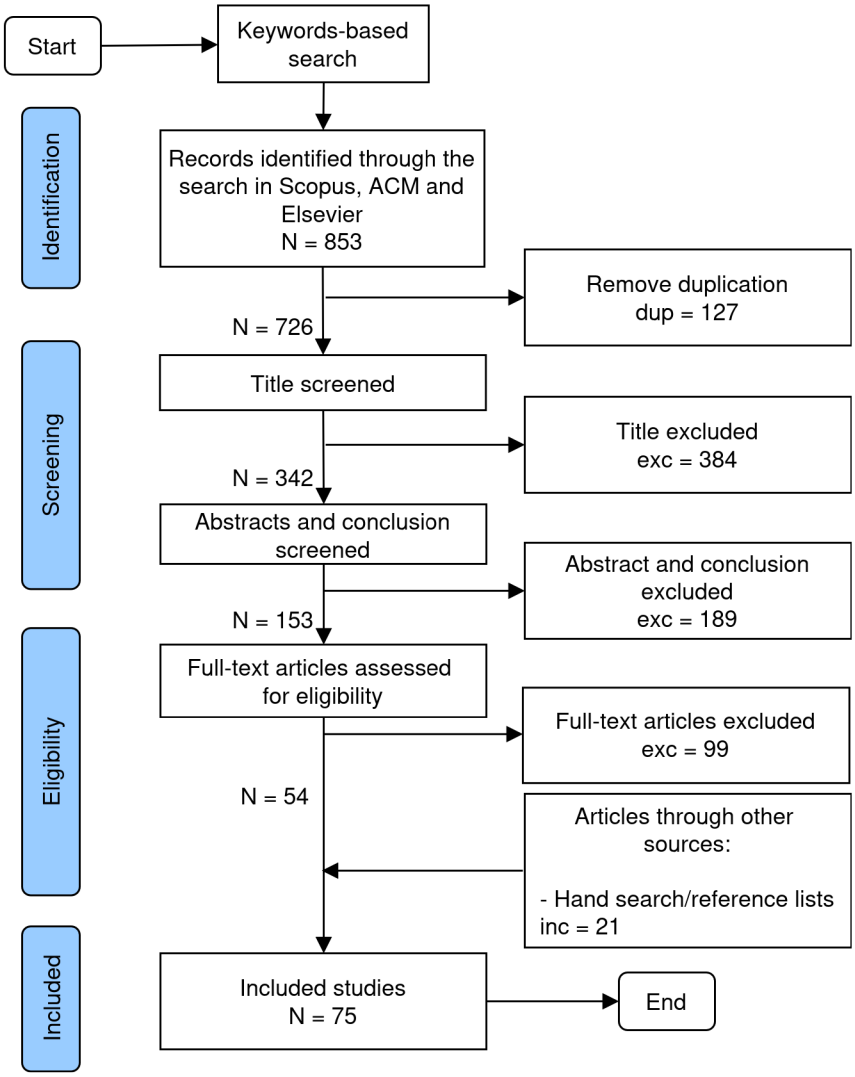


Figure 3.1: PRISMA flowchart of papers identified, excluded and included in the dissertation.

Portuguese has received less attention than English, Chinese, and Spanish. This disparity is manifested in the quantity and quality of language models available and in the linguistic resources and textual corpora used for their training.

In recent years, there have been efforts to develop Portuguese language models. Two examples are BioBERTpt [13] and GPT-2-Bio-Pt [74], which are trained using existing BERT [15] and GPT-2 [63] models and then specialized using data from clinical narratives and biomedical scientific articles in Brazilian Portuguese. BioBERTpt is effective for tasks requiring an understanding of relationships within text, thanks to its ability to capture nuances and complex connections between biomedical terms. On the other hand, GPT-2-Bio-Pt excels at generating fluent and coherent text, essential for applications requiring the creation of natural language content. These models show that starting with a pre-trained transformer model is suitable for tasks demanding a detailed analysis of the context of words in a text.

Another relevant model is PTT5 [7], derived from the T5 [64] model and trained on data from BrWac [17], a collection of Brazilian Portuguese web pages. PTT5 is capable

of performing tasks such as summary generation, abstract question answering, and text translation. In contrast, the BERTimbau model, an adaptation of BERT, is trained on the same dataset of PTT5, but is a more suitable model for tasks that require text understanding and information extraction, such as sentiment analysis, document classification, and information retrieval [78]. While PTT5 focuses on text generation, BERTimbau specializes in textual understanding and analysis, providing valuable tools for various applications in Portuguese natural language processing.

From our basic comparative analysis carried out in the previous paragraphs on Portuguese models derived from BERT, GPT and T5, we can conclude that a BERT-based model is shown to be the most suitable for the task of sentiment analysis. This is due to BERT's ability to capture contextual relationships in text sequences, which is essential to correctly interpret emotions and opinions expressed in natural language. The GPT and T5 models, although powerful in various applications, particularly excel in areas such as text generation and translation.

PetroBERT is a BERT-based model adapted to the oil and gas exploration domain in Brazilian Portuguese [68]. In this article, Rodrigues *et al.* (2022) make comparisons between the monolingual pretrained model BERTimbau, and the multilingual pretrained model mBERT [59], adapted to a specific domain. Two PetroBERT models were trained, one starting from the weights of the BERTimbau model and the other starting from the weights of the mBERT model. The results showed that PetroBERT, initialized from BERTimbau, outperformed the model initialized from mBERT on several specific tasks. It can be seen from the paper that it is better to initialize the weights from a monolingual pretrained model compared to a multilingual one. This observation has been supported in other studies [14, 49, 81, 2, 8], which have also found advantages in using monolingual models over multilingual ones for specific natural language processing tasks [68].

There is also the LegalBERT model, specialized in the legal domain of Brazilian Portuguese [75]. This model trains two variants: one from scratch and another starting from the pretrained weights of BERTimbau. Both variants achieve superior results compared to general models such as the BERTimbau Base and show similar performance to each other. From these results it can be inferred that, in certain scenarios, starting with a general pretrained model in the same language of the specific domain can offer results as optimal as a model trained from scratch in that domain. This suggests that the reuse of pre-trained models in the same language is not only feasible, but also efficient, avoiding the need for a complete training from scratch and taking advantage of the wealth of knowledge already present in the pre-trained models.

It is worth remembering that our proposal was formulated in mid-2022, and since then new language models specific to the Portuguese language have emerged. However, we have continuously monitored these updates to ensure the relevance and timeliness of our research. Among recent models, BERTabaporu stands out, trained specifically for the Twitter domain, managing to outperform the results of BERTimbau in this specific context of Brazilian Portuguese social networks [12]. Written texts on Twitter have different characteristics compared to texts from more elaborate opinions or comments. These differences can be seen mainly in the use of a more informal and abbreviated language, as well as in the presence of hashtags, mentions and emojis.

Another notable model is ALBERTina, which improves the results of BERTimbau Large in several tasks [67]. ALBERTina incorporates improvements in deinterlaced attention and the mask decoder, features inherited from DeBERTa [23], which optimizes its performance in certain applications. However, ALBERTina is only available in a Large version, which implies a greater demand for parameters and computational resources for its training.

In addition, language models with significantly more parameters, designed for general domains in Brazilian Portuguese, have emerged, such as Sabiá [58] and TeenyTinyLlama [11], based on the LLaMA architecture. These models represent significant advances in the processing and understanding of natural language in Portuguese, offering new opportunities for the development of more sophisticated and accurate applications.

In Table 3.1, we provide a summary of the language models in Portuguese discussed in this chapter.

Table 3.1: A number of language models in Portuguese that exist in the literature; we highlight the model that best fits our proposal.

	Come from BERT	General domain	Less than 110M
BioBERTpt PetroBERT LegalBERT	✓	×	✓
GPT-2-Bio-Pt	×	×	✓
PTT5 small	×	✓	✓
PTT5 base PTT5 large	×	✓	×
BERTimbau Base	✓	✓	✓
BERTimbau Large ALBERTina	✓	✓	×
BERTabaporu Base	✓	×	✓
BERTabaporu Large	✓	×	×
Sabiá 6B Sabiá 7B Sabiá 65B TTL-160m TTL-460m	×	✓	×

3.3 Self-Supervised Fine-Tuning Approaches

The use of pretrained models for domain-specific specialization using a variety of unlabeled data has been widely documented in the literature. In particular, the BERT pretrained

model has been a model of considerable interest and has been the subject of numerous studies due to its innovative approach to address the domain adaptation challenge [15]. The domain adaptation solution largely relies on two essential components: the vocabulary and the pretrained model.

The vocabulary, which constitutes the set of tokens used by the model to process and understand text, plays a crucial role in BERT's ability to handle terms specific to a particular domain. Adapting the vocabulary to include tokens that are frequent and relevant in the new domain can significantly improve the model's performance in specific tasks, allowing for a more precise understanding of the context and semantics of the domain [3].

On the other hand, the pretrained model provides general knowledge derived from its initial training, reflected in the values of its weights. However, this benefit is only exploitable if the same vocabulary is used. Changing the vocabulary or making variations to it may result in incompatibilities, which could require additional components or, alternatively, a new training from scratch.

Taking into account the above-mentioned information, the literature presents several domain-specific pretrained models that create a new specific vocabulary and train BERT from scratch, such as SciBERT [3] for the scientific domain, PubMedBERT [21] for the biomedical domain, FinBERT [87] for the financial domain, Legal-BERT [9] for the legal domain, and IndoBERT [37] for the Indonesian language. Self-supervised training allows these models to better capture domain particularities and to obtain better results in specific tasks, demonstrating the effectiveness of a specialized vocabulary combined with from-scratch pre-training. However, the main drawback of these models is the high computational cost and the large amount of data required for their training. Beltagy *et al.* (2019) [3] conclude that, although it is useful to have one's own vocabulary, the model benefits more from training with the domain-specific corpus.

An alternative approach in the literature is represented in the exBERT [79] model, which, like IndoBERTweet [36], generates a new vocabulary for a specific domain. However, unlike other approaches, they only select the new words and add them to the original vocabulary of the pre-trained model, avoiding inconsistencies in the integration of the new vocabulary with the model architecture; thus combining the robustness of the original vocabulary with the specificity of the new vocabulary [38]. Hong *et al.* (2021) [25] describes in their scientific article a method to achieve this increase in vocabulary, which is used in some studies by other authors. These models are not trained from scratch, but continue the pre-training from the already established weights, achieving an efficient and effective integration of new terminologies without sacrificing the stability of the base model. The disadvantage of this approache is that increasing the vocabulary size impacts performance and increases the time to convergence. As pointed out by Yang *et al.* (2023) [86], these models not only demand large volumes of domain-specific data to achieve optimal performance, but also require considerable computational power during the pre-training phase. Furthermore, this approach may lead to redundancy in memory usage, which adds an additional layer of complexity in managing computing resources.

Other approaches mentioned in the literature include the BioBERT [43] models for the biomedical domain and ClinicalBERT [29] for clinical annotations. These models use

the weights and vocabulary of the base model, BERT, to perform self-supervised fine-tuning in a specific domain. This allows them to benefit from the knowledge already learned by the base model, while retaining their flexibility and versatility in general tasks. By not requiring vocabulary expansion, they can handle new words from the domain by splitting them into existing subwords. During fine-tuning, the model adjusts its weights to improve the internal representations of these combinations in the context of the new domain, resulting in greater efficiency in terms of training time and use of computational resources. This will be the approach used in our proposal due to the characteristics mentioned.

So far, none of the mentioned models have implemented a layer-freezing approach. However, Lee et al. [42] demonstrated that by freezing only a quarter of the final layers of a transforming model, it is possible to achieve up to 90% of the quality of the original model. Inspired by these findings, our proposal aims to empirically evaluate the applicability of this approach in the context of our specific problem. To do so, we will perform a detailed analysis of the progressive unfreezing of each of the layers of the model, identifying those that are most relevant to the target task. These selected layers will then be used to develop the analyses that support our approach.

In Table 3.2, we provide a summary of the approaches for model domain adaptation that were discussed in this chapter.

Based on the analysis performed, BERT-based models have proven to be outstanding tools for sentiment analysis, particularly when using pre-trained monolingual Portuguese variants, such as BERTimbau. These solutions offer significant advantages over multilingual models, both in accuracy and adaptability to the linguistic domain. Although recent models have introduced improvements in general tasks, their high computational requirements make it difficult to align them with the efficiency goals raised in this research. Inspired by the benefits of selectively freezing layers during fine-tuning, our proposal is based on a pragmatic and efficient approach. This includes not only the use of BERTimbau with self-supervised fine-tuning, but also the empirical evaluation of progressive layer freezing and unfreezing strategies to optimize the balance between accuracy and efficiency. This strategy leverages the pre-existing knowledge of the model, identifies the most relevant layers for the target task, and minimizes costs associated with training from scratch, thus achieving an optimal balance aligned with the current challenges of natural language processing in Portuguese.

Table 3.2: The different approaches available in the literature for adapting models to specific domains are presented, as well as which approach we are applying in our proposal, adding specific requirements of our problem and also our contribution.

	Vocabulary	Pretrained model weights	Portuguese language	Unfreeze only some of the layers
Legal-BERT PTT5 BERTimbau ALBERTina BERTabaporu Sabiá TTL	Own	✗	✓	✗
SciBERT PubMedBERT FinBERT IndoBERT KB-BERT	Own	✗	✗	✗
exBERT, IndoBERTweet	Inherits + Extended	✓	✗	✗
PetroBERT, BioBERTpt, GPT-2-Bio-Pt	Inherits	✓	✓	✗
ClinicalBERT, Clinical KB-BERT	Inherits	✓	✗	✗
Our proposal	Inherits	✓	✓	✓

Chapter 4

Methodology

The establishment of the retraining protocol for self-supervised fine-tuning of the BERTimbau model, proposed in this Master's project, was based on a detailed analysis of the hyperparameters, focusing on the learning rate and the number of unfrozen layers, depending on the size and context of the data sets.

Starting from the BERTimbau Base model, trained on a general domain in Brazilian Portuguese, the domain adaptation was carried out using three data sets of different sizes and contexts. These sets were mostly divided for retraining using self-supervised fine-tuning, which requires unlabeled data, and a smaller portion was used for the downstream task that uses labeled data. During the self-supervised fine-tuning process, various configurations were tested, adjusting the number of unfrozen layers and the learning rate. Finally, the fine-tuned model was evaluated on a multiclass sentiment analysis downstream task using cross-validation, to validate the effectiveness of the implemented adaptations.

This approach aims to show how effective the self-supervised fine-tuning technique is. It also illustrates the model's ability to adapt and generalize across different sizes and domains, based on the number of unfrozen layers and the learning rate used.

The pipeline for evaluating the proposed strategy is presented in Figure 4.1. In the following, we describe the data sets, the adopted pre-trained language model, the details of the self-supervised fine-tuning procedure, the downstream task and the evaluation metrics.

4.1 Data Sets

An exhaustive search was conducted on the Internet for public data sets in Brazilian Portuguese, covering various domains and sizes, containing labeled data for the task of multiclass sentiment analysis. This search was motivated by the lack of labeled data to be evaluated, in the final stage of our proposal, and seeking to compare our results to other existing models.

We chose to collect data sets from different contexts in order to demonstrate the versatility and effectiveness of our methodology in varied contexts. The diversity of contexts ensures that our proposal is not limited to a single field of application, but can be adapted and still provide optimal results in a different context. This strategy is essential to support

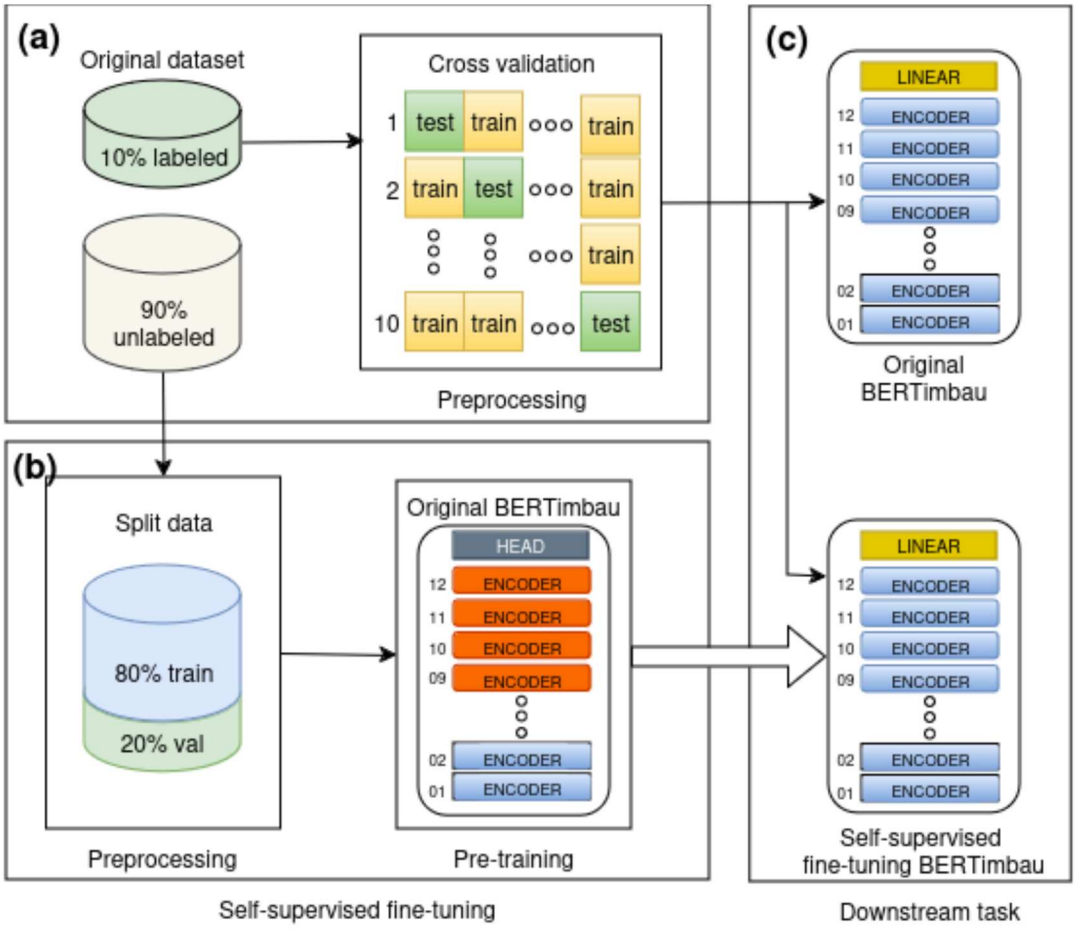


Figure 4.1: Pipeline of the proposed methodology. **(a)** The split of the labeled dataset - 90% of data have their labels discarded and used for self-supervised fine-tuning, the remaining labeled data is kept for the downstream task; **(b)** Self-supervised fine-tuning of BERTimbau, where some encoder layers are unfrozen (orange) for further adjustment with unlabeled data from the downstream context; **(c)** Supervised fine-tuning of the BERTimbau model with self-supervised fine-tuning and the original BERTimbau model.

the validity and applicability of our self-supervised fine-tuning approach in specializing language models in different contexts in which the Portuguese language is used.

The data sets that met our requirements were carefully selected to ensure their relevance and suitability to our research. Listed below are the chosen data sets, which possess the necessary characteristics for the task of multiclass sentiment analysis in Brazilian Portuguese and represent a variety of sizes and contexts, providing a solid basis for the validation of our proposal:

- **B2W** [65]: It is an open corpus that deals with product reviews of a Brazilian company. It has a size of 67.03 MB, and contains more than 130,000 customer reviews collected from a retailer website between January and May 2018. It divides opinion of the customer into five levels, five being a very favorable opinion and one a very unfavorable opinion towards a certain product.

- **UTL-Corpus** [77] : It is a corpus in Portuguese obtained through crawlers; it was

created especially for the polarity classification of online reviews, using the "likes" counter on each review, which allowed them to infer a usefulness option. The authors previously anonymized and preprocessed the datasets. There are two UTL-Corpus datasets, the **utlc-apps** (285.1 MB) for technology reviews and the **utlc-movies** for movies reviews (1.1 GB).

The main properties of those two data sets are shown in Tables 4.1 and 4.2.

Table 4.1: Context, size and characteristics of each data set that we use in this dissertation.

	Context	Size	Number of records	Number of words	Target features	Classes
B2W	e-commerce	67.03MB	132 k	3,5 M	rating	5
UTLC-apps	smartphone applications	285.1MB	1040 k	14,9 M	rating	5
UTLC-movies	movies	1.1 GB	1488 k	62,5 M	rating	5

The datasets underwent a rigorous preprocessing process. Initially, records with empty or undefined data (e.g. NaN) were removed, ensuring that only complete and useful entries were retained for analysis. In addition, duplicate records were identified and removed to avoid redundancies and ensure data integrity. This step was crucial to ensure data quality and consistency, thus allowing a solid and reliable basis for training and evaluating the models.

Table 4.2: Preprocessing performed on the data sets we used and the number of records remaining at the end of this process.

	Initial number of records	empty or undefined records	duplicate records	Final number of record
B2W	132.373	3.275	2.305	126,793
UTLCapps	1.039.535	0	221.359	818.176
UTLCmovies	1.487.449	0	80.120	1.407.329

For the purposes of our investigation, we proceeded to split each of the datasets in a strategic manner. We allocated 90% of the total data to carry out self-supervised fine-tuning, ensuring that the model is trained exhaustively and robustly on a large sample of data. The remaining 10% was reserved for the downstream task, in order to evaluate the model's performance in a practical and realistic context (Figure 4.1(a)). This split allows us to validate the effectiveness of our proposal, ensuring that the model not only fits the training data well, but also generalizes adequately to unseen data, thus demonstrating its applicability and efficiency in the multiclass classification task.

4.2 Pre-trained Language Model

In the realm of natural language processing, BERT has proven to be a crucial tool for transfer learning. Its ability to bidirectionally understand context has allowed models

pre-trained on large corpora of unlabeled data to efficiently adapt to specific tasks with smaller, labeled datasets. This feature is particularly relevant in scenarios where the availability of labeled data is limited, allowing researchers to leverage large volumes of unlabeled text to pre-train robust and versatile models. BERT's pre-training methodology, followed by fine-tuning on specific tasks, has demonstrated exceptional results, setting a new standard in the field.

In the Brazilian Portuguese domain, BERTimbau emerges as a specialized adaptation of BERT. BERTimbau was trained using a large corpus of Brazilian Portuguese text, allowing it to effectively capture the specific linguistic peculiarities and structures of this language. This adaptation is essential to improve performance in Portuguese natural language processing (NLP) applications, given that models originally trained in other languages may not adequately capture the particularities of Brazilian Portuguese [81].

Using BERTimbau on natural language processing tasks for Brazilian Portuguese has shown that models pre-trained in the same language of the specific domain deliver significantly better results compared to multilingual models. This methodology of specializing the pre-trained model through self-supervised transfer learning allows for a more accurate understanding of the text, which is crucial for applications such as sentiment analysis, document classification, and information retrieval in Portuguese.

4.3 Self-Supervised Fine-Tuning Approach

The goal at this stage is to perform self-supervised fine-tuning starting from the weights of the pre-trained language model BERTimbau Base. The choice to perform self-supervised fine-tuning was carefully substantiated based on the scope defined for this dissertation and the tools available at the start of the project (May 2022). During this period, emerging options such as LoRA (Low-Rank Adaptation) [28] were not yet widely disseminated in the literature or in mature implementations, limiting their consideration as a viable alternative at that time. On the other hand, contextual learning-based approaches (such as prompt-tuning or in-context learning) were also considered, but did not fully align with the goal of this dissertation. Furthermore, large-scale language model (LLM) architectures, although promising, present significantly higher computational and implementation costs, which exceeded the resources and scope of this research.

To initiate the self-supervised fine-tuning approach, it was necessary to define a dataset partitioning strategy, dividing the unlabeled data into subsets for training and validation. As shown in Figure 4.1(b), the input 90% of unlabeled data was divided as follows: 80% for training and 20% of the data for validation step. This division was determined by pilots studies, which evalutated the impact of different split ratios on the B2W data set (Appendix A, Tables A.1, A.2 and A.3).

To perform self-supervised fine-tuning on the pre-trained model, prediction heads are added using the `BertForpre-training` class of the Hugging Face `transformers` library, which are responsible for performing the two pretext tasks: masked language model (MLM) and next sentence prediction (NSP). In the MLM task, a probability of 0.15 was used to randomly mask words within the text, allowing the model to learn to predict

the masked words based on the surrounding context. For the NSP task, a probability of 0.50 was applied for the next sentence to be the real one, meaning that when sentence pairs are created, there is a 50% chance that the second sentence in the pair is the sentence that actually follows the first one; in the other 50% of the cases, the second sentence is randomly selected from somewhere else in the corpus, and has no logical or sequential relationship with the first sentence. This serves to challenge the model to understand coherence and continuity between sentences. This is data taken from the training performed by BERT.

We used the BERTimbau tokenizer, which contains a vocabulary of 29,794 tokens, through the `BertTokenizer` class. The maximum number of input tokens was set to 512 to ensure that the model can handle long text sequences without losing crucial information. The batch size was set to 64 for both training and validation, balancing the computational load and efficiency of the training process.

To optimize the training process, the "Early Stopping" technique was implemented with a patience of 5 and a tolerance of 0.001, allowing a maximum of 40 epochs. This technique is vital to prevent overfitting of the model by stopping training when validation performance stops improving significantly. Furthermore, all experiments were carried out with a fixed seed of 42 to ensure reproducibility and consistency of results.

At this stage, experiments were conducted by varying the learning rate (1e-4, 1e-5, 1e-6) with a decay rate of 0.01 and a thorough empirical analysis of the progressive unfreezing of the model layers was carried out, with the aim of identifying configurations that would optimize the results for our task. As a result of this process, the four most relevant unfreezing strategies were selected, based on their positive impact on the model performance: unfreezing all layers, the last four layers, the last two layers, and only the last layer. These adjustments were made to assess the impact of each factor on model performance, providing essential insights for optimization.

4.4 Downstream Task

We chose multiclass sentiment analysis (Figure 4.1(c)) due to its higher complexity compared to binary classification, presenting a significant challenge to assess the robustness of our model. To address this task, we performed data balancing using undersampling, which involves reducing the amount of data in the largest classes until it equals the number of instances in the smallest class. This process was performed randomly to select the data to preserve, and then we shuffled all the data to ensure an equal distribution between the classes.

We used the model obtained in the self-supervised fine-tuning stage, freezing all its layers to keep the weights intact and avoid changes during the training of the downstream task. To this model we added a classification layer with five outputs, corresponding to the number of classes in our dataset.

For the multiclass sentiment analysis, we employed the AdamW optimizer, known for its efficiency and ability to handle large models. The loss function used was the multinomial cross entropy, which is suitable for multiclass classification tasks. The maximum

number of input tokens was set to 512 to ensure that the model could process long text sequences. The learning rate was set to 1e-4, and the batch size was 16, balancing the computational load and training efficiency. The model was trained for 10 epochs, with a fixed seed of 42 to ensure reproducibility of the results.

In addition, cross validation was implemented with 10-fold. This technique is crucial to evaluate the model performance more robustly, by splitting the dataset into 10 parts and using each of them as a validation set while training the others. This not only helps mitigate the risk of overfitting, but also provides a more accurate and generalizable assessment of the model performance on different data subsets.

4.5 Evaluation Metrics

The evaluation of models is critical to ensure their effectiveness and generalization. Selecting appropriate metrics is essential, especially when working with multiclass classification tasks and employing cross-validation techniques to obtain a robust evaluation.

In this research, we used three main metrics to evaluate the performance of our models: *"Weighted Accuracy"*, *"Balanced Accuracy"* and *"Weighted F1"*. These metrics were selected for their ability to offer a comprehensive and detailed view of the model's behavior, addressing different aspects of accuracy and balance in classification.

The choice of these metrics allows us to obtain a more complete and accurate evaluation of the model's performance, considering both the distribution of the classes and the model's ability to generalize in different scenarios. In the following sections, each of these metrics will be detailed, highlighting their definition, calculation and relevance in the context of this research. It will be discussed how each metric contributes to a complete evaluation of the model's performance, allowing a more detailed understanding of its behavior in the task of multiclass sentiment analysis addressed in this study.

4.5.1 Balanced Accuracy

Balanced Accuracy is a key evaluation metric in multiclass sentiment analysis problems, especially useful when faced with imbalanced data sets. Unlike simple accuracy, which can be biased towards the most prevalent classes, balanced accuracy provides a fairer and more representative measure of model performance by considering performance across all classes equally.

For multiclass classification problems, balanced accuracy is defined as the average of the sensitivities (recalls) of each class. The general formula for balanced accuracy in the multiclass context is:

$$\text{Balanced Accuracy} = \frac{1}{N} \sum_{i=1}^{N} \frac{TP_i}{TP_i + FN_i}, \tag{4.1}$$

where N is the number of classes, TP_i is the number of true positives for class i, and FN_i is the number of false negatives for class i. Each class is considered positive in turn, while the rest of the classes are considered negative.

Balanced accuracy is particularly relevant in scenarios where classes are imbalanced. In such cases, a model may show high simple accuracy in mostly predicting the dominant class, but fail to correctly identify instances of the minority classes. Balanced accuracy addresses this problem by evaluating the model equally across all classes, better reflecting its actual performance.

In the literature, balanced accuracy has been widely recommended for evaluating models on imbalanced datasets. For example, Brodersen *et al.* (2010) argue that balanced accuracy is a suitable metric for comparing classifiers in the presence of class imbalance and cross-validation, providing an unbiased measure of model performance [5]. Likewise, the work of Kelleher *et al.* (2015) on the evaluation of machine learning algorithms highlights the importance of using metrics that consider class imbalance, such as balanced accuracy, to obtain a more accurate evaluation of the model [34].

In the context of our research, we have chosen to use balanced accuracy due to its ability to provide a fair and equitable evaluation of the model's performance across all classes, regardless of their distribution. Since we work with multiclass classification tasks and our classes are balanced, balanced accuracy ensures that the model's performance is assessed uniformly across all classes. This choice is crucial to validate the effectiveness of our approach and ensure that our model is robust and generalizable to different class distributions.

4.5.2 Weighted F1

The Weighted F1 is a widely used evaluation metric for multiclass classification problems. This metric is especially useful when a balanced measure of model performance is desired, considering both the precision and recall of each class. Unlike the macro F1 score, which treats all classes equally, the Weighted F1 weights the F1 score of each class based on the proportion of instances of that class in the dataset. This ensures that more frequent classes have a greater impact on the Weighted F1 score, more accurately reflecting the model's performance in scenarios with balanced classes.

The F1 score for a class is defined as the harmonic mean of the precision and recall of that class. Precision is the fraction of relevant instances among the retrieved instances, while recall is the fraction of relevant instances that have been retrieved out of the total number of relevant instances. Mathematically, the F1 score for class i is expressed as:

$$F1_i = 2 \cdot \frac{\text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i}, \tag{4.2}$$

where Precision_i is the precision for class i and Recall_i is the recall for class i. Precision and Recall are calculated as follows:

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \tag{4.3a}$$

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i}. \tag{4.3b}$$

Here, TP_i is the number of true positives, FP_i is the number of false positives, and FN_i

is the number of false negatives for class i.

The Weighted F1 score is calculated as the weighted average of the F1 scores for each class, where the weights are the proportions of instances of each class in the dataset. This is expressed mathematically as:

$$\text{Weighted F1} = \sum_{i=1}^{N} w_i \cdot F1_i, \tag{4.4}$$

where w_i is the proportion of class i in the dataset, defined as:

$$w_i = \frac{N_i}{N}, \tag{4.5}$$

where N_i is the number of instances of class i and N is the total number of instances in the dataset.

In the context of our research, we have decided to use the Weighted F1 metric due to its ability to provide a balanced and fair assessment of model performance across classes. Even though our classes are balanced, the Weighted F1 allows us to accurately measure model performance in multiclass classification problems, where each class has equal importance. Furthermore, this metric is especially relevant in cross-validation scenarios, where a consistent assessment across different folds of the dataset is needed.

The academic literature supports using the Weighted F1 in multiclass classification problems. In the study by Opitz and Burst (2019) titled "Macro F1 and Macro F1" they emphasize how weighted metrics can provide a more balanced and representative view of model performance on multiclass datasets [52].

Implementing the Weighted F1 score involves correctly calculating the precision and recall for each class, and then averaging the F1 scores of each class by weighting them by the proportion of instances of that class in the dataset. In our approach, we have ensured that this calculation is performed accurately and consistently across all cross-validation folds, thus ensuring a robust and reliable model evaluation.

4.5.3 Weighted Accuracy

Weighted Accuracy is a crucial metric in evaluating multiclass classification models, especially in scenarios where classes are imbalanced. This metric considers the proportion of each class in the dataset, providing a more representative measure of model performance than simple accuracy, which can be biased towards more prevalent classes. Weighted Accuracy is calculated by assigning a weight to the accuracy of each class, proportional to the number of instances of that class in the dataset. The general formula for weighted accuracy is:

$$\text{Weighted Accuracy} = \sum_{i=1}^{N} w_i \cdot \text{Accuracy}_i, \tag{4.6}$$

where N is the number of classes, w_i is the weight of class i (usually the proportion of class i in the dataset), and Accuracy_i is the accuracy of class i.

The weighted accuracy metric is widely used in research and practical applications. In

the field of natural language processing (NLP), for example, the study by Sokolova and Lapalme (2009) highlights the importance of using weighted metrics to evaluate classifiers on imbalanced datasets [76]. Likewise, the work by Fernández-Delgado *et al.* (2014) shows that weighted accuracy is essential to compare the performance of different classifiers on multiple datasets with varying class distributions [16].

In the context of our research, we have decided to use weighted accuracy due to the multiclass nature of our tasks and the balance of classes in our data sets. Although we took the trouble to balance our classes, weighted accuracy is still valuable because it ensures that the model's performance is evaluated fairly and comprehensively across all classes. This choice allows us to avoid potential bias if some classes, although balanced in number, present different difficulty levels for the model. Furthermore, by using weighted accuracy, we can provide a more robust and detailed assessment of our model's performance on each individual class, which is crucial to ensure the validity and generalizability of our results.

When implementing weighted accuracy, it is crucial to ensure that the weights are correctly calculated based on the distribution of classes in the dataset. Furthermore, it is important to maintain consistency in weighting across the different folds in cross-validation scenarios to ensure a fair and representative model evaluation.

4.6 Computational Resources

For the execution of these experiments, a server equipped with an NVIDIA A100 GPU and 80 GB of memory was used, provided by the Artificial Intelligence Laboratory (Recod) of the Institute of Computing, University of Campinas. That advanced infrastructure was essential to handle the intensive computational demands of model training, ensuring efficient and accurate results and allowing effective exploration of complex configurations.

The experiments were conducted in a Jupyter Notebook environment, using Python 3.12 as the primary programming language. Various libraries and tools were employed to facilitate data handling, model training, evaluation, and visualization. The main libraries used include:

- **Torch**: For implementing and training deep learning models.
- **Transformers**: To utilize and fine-tune pre-trained language models such as BERTimbau.
- **Datasets**: For efficient dataset loading and preprocessing.
- **Scikit-learn**: For implementing machine learning utilities and evaluation metrics.
- **Permetrics**: For computing advanced metrics during the evaluation phase.
- **Pandas**: For data manipulation and analysis.
- **Matplotlib**: To generate visualizations for exploratory data analysis and result presentation.

The server ran on a Linux-based operating system, which ensured compatibility and stability for executing the experiments and managing the computational resources effectively. This combination of hardware and software allowed the experiments to be conducted efficiently, addressing the challenges of training computationally intensive models while maintaining reproducibility and scalability.

4.7 Source Codes

All the source code, scripts, and resources developed for this research are available in an open-source repository hosted on GitHub. This repository contains the implementation of the self-supervised fine-tuning protocol, data processing scripts, and experimental configurations used throughout this dissertation.

The repository can be accessed at the following link: `https://github.com/pantro/self-supervised-fine-tuning-for-portuguese-language-models`

The structure of the repository is as follows:

- `data/`: Instructions and links to datasets used in this research.
- `notebooks/`: Jupyter notebooks for exploratory data analysis, fine-tuning, and evaluation.

The repository is released under the **MIT License**, which allows others to reuse, modify, and distribute the code in accordance with the terms specified in the license.

We encourage researchers and practitioners to explore and build upon this work to further advance Portuguese language modeling.

Chapter 5

Results and Discussion

The results of this Master's dissertation were obtained through the rigorous evaluation of several tests designed to address our research questions. First, we performed a general experiment to understand the performance of self-supervised fine-tuning in various scenarios. For this purpose, three data sets belonging to different domains and of different sizes were used, a percentage of which were used for the self-supervised fine-tuning process and another percentage for the downstream task. The results obtained in this experiment were compared with those of a general Portuguese model, such as BERTimbau Base, which served as our baseline.

To carry out self-supervised fine-tuning, we decided to vary the number of unfrozen model layers and the learning rate used in training. This way, we evaluated the influence of these hyperparameters on the final results. As for the unfrozen layers, four different configurations were tested: all layers, the last four, the last two, and the last layer unfrozen. Three different values were tested for the learning rate: 1e-4, 1e-5, and 1e-6.

This comprehensive approach ensures that the model is tested in multiple contexts, providing a more holistic view of the classification results. In addition, comparing the results with the BERTimbau Base model enables a more precise evaluation of the effectiveness of domain adaptation and the impact of fine-tuning hyperparameters on model performance. This methodological approach is crucial for comprehensively assessing the model's capabilities in specific natural language processing tasks. The variation in domains and dataset sizes used in the experiment provides a robust and detailed evaluation, ensuring that the results are both representative and generalizable across different contexts and applications [78].

On Table 5.1, we present results from learning rate exploration obtained for all datasets. Our main findings in the B2W data set can be summarized as follows:

- The best results were obtained by unfreezing all layers and using the learning rate 1e-5. We can also see that for this dataset, regardless of how many layers we unfreeze in the model, we get the best results with the learning rate equal to 1e-5;
- When using the learning rate of 1e-4, regardless of the number of layers unfrozen, there is not much variation of the metrics compared to the baseline;
- We observe that when using the smallest LR 1e-6, the results get worse as fewer

Table 5.1: Results obtained when performing the downstream task of multiclass sentiment analysis using the various models that were subjected to self-supervised fine-tuning. The tested configurations included different combinations of unfrozen layers and different learning rate values. The datasets used were B2W, UTLC-apss, and UTLC-movies, which cover a variety of domains and sizes. The balanced accuracy (B_Acc), weighted F1 (W_F1) , and weighted accuracy (W_Acc) metrics were used to evaluate the models' performance. In addition, a column with the results of the BERTimbau Base model, subjected to the same downstream task, was included as a baseline.

	Multiclass sentiment analysis (10% data of total)												
	BERT imbau (baseline)	SSL with ALL layers unfreeze			SSL with 4 layers unfreeze			SSL with 2 layers unfreeze			SSL with 1 layers unfreeze		
		1e-4	1e-5	1e-6	1e-4	1e-5	1e-6	1e-4	1e-5	1e-6	1e-4	1e-5	1e-6
B2W													
B_Acc	0.432	0.429	**0.476**	0.458	0.412	**0.457**	0.450	0.435	**0.451**	0.408	0.427	**0.466**	0.389
W_F1	0.422	0.419	**0.459**	0.443	0.391	**0.437**	0.431	0.422	**0.431**	**0.431**	0.407	**0.447**	0.372
W_Acc	0.773	0.772	**0.790**	0.783	0.765	**0.783**	0.780	0.774	**0.781**	0.763	0.771	**0.786**	0.756
UTLC-apps													
B_Acc	0.409	0.409	0.442	**0.452**	0.385	**0.444**	0.424	0.429	**0.430**	0.397	**0.434**	0.433	0.395
W_F1	0.395	0.383	0.426	**0.436**	0.364	**0.427**	0.406	**0.412**	0.406	0.378	**0.421**	0.412	0.376
W_Acc	0.764	0.764	0.777	**0.781**	0.754	**0.778**	0.770	0.771	**0.772**	0.759	**0.774**	0.773	0.758
UTLC-movies													
B_Acc	0.364	0.385	0.409	**0.419**	0.378	**0.400**	0.398	0.402	**0.424**	0.383	0.409	**0.414**	0.364
W_F1	0.350	0.363	0.394	**0.404**	0.361	**0.386**	0.383	0.388	**0.407**	0.367	0.398	**0.399**	0.349
W_Acc	0.746	0.754	0.763	**0.768**	0.751	**0.760**	0.759	0.761	**0.770**	0.753	0.764	**0.766**	0.745

layers of the model are unfrozen.

On the other hand, our main findings for the UTLC-apps dataset (Table 5.1) are:

- The best results are obtained by unfreezing all the layers and using the LR 1e-6;
- Similar to the previous dataset (B2W) if we choose a very small LR (1e-6) the results get worse as we unfreeze fewer layers of the model. For instance, with LR equal to 1e-6 and unfreezing all the layers, we get the best result, but the worst result is if we only unfreeze the last layer;
- With LR 1e-5 in any of the cases, regardless of the number of layers unfrozen, we get better results than the baseline;
- Despite not being the best result in this dataset, unfreezing only the last layer with the higher learning rate 1e-4 returns higher results than the baseline.

Finally, our main findings for the UTLC-movies data set (Table 5.1) are:

- The best results for this dataset are when the last 2 layers are unfrozen and the learning rate is 1e-5;
- All the cases outperformed the baseline, except for the model that unfrozen the last layer, and the LR is 1e-6. This behavior did not occur in previous datasets;

- The LR 1e-5 is again the most suitable choice, when unfrozen some of the layers but not all. The best result, when unfrozen all is with the smallest LR, equal to 1e-6;
- It is true that the fewer layers you unfreeze, for a LR 1e-6, the worse results you get. In other words, the smaller the LR, the better to thaw as many layers as possible.

Looking at Table 5.1 as a whole, we can convey the following observations:

- We observe that we can reach higher results than the baseline by unfreezing only the last layer in the datasets and with an adequate learning rate;
- We can observe that using a learning rate (LR) of 1e-5 for self-supervised fine-tuning yields better results than the baseline, regardless of the number of unfrozen model layers;
- Reviewing the table, we can see that as the dataset size increases, the difference in percentage points between the baseline results and the best results of the self-supervised fine-tuning model also increases.
- We can observe that for all data sets, it is true that if we have fewer unfreezing layers and use a small LR such as 1e-6, the results get worse.

In the sequence, we focused on the best hyperparameter choices (number of unfreeze layers, learning rate) for each of the three datasets. In Table 5.2, we can observe only the best results of combining the number of unfreeze layers and the learning rate for those three data sets used; each value is with its standard deviation to see how dispersed these data are to the mean. On the other hand, Figures 5.1, 5.2, and 5.3 show how all the models that underwent self-supervised fine-tuning with the three data sets used outperformed our general domain baseline. In the upper right corner of each figure there is an enlarged box that highlights in greater detail the differences between the results. This box allows a more precise visualization of the improvements achieved, facilitating comparison and highlighting the advantages of our proposal.

Table 5.2: Summary of results taking into account the best hyperparameters for each data set. The best results are obtained by combining the number of unfreeze layers of the model and a certain learning rate for each data set. The standard deviation highlights the variability of the evaluated metrics for the different data sets and hyperparameters applied.

	Multiclass sentiment analysis				
	BERTimbau (baseline)	**SSL with ALL layers unfreeze**	**SSL with last 4 layers unfreeze**	**SSL with last 2 layers unfreeze**	**SSL with last layer unfreeze**
		B2W			
		(LR=1e-5)	(LR=1e-5)	(LR=1e-5)	(LR=1e-5)
B_Acc	0.432 ± 017	**0.476 ± 0.024**	0.457 ± 0.020	0.451 ± 0.016	0.466 ± 0.014
W_F1	0.422 ± 018	**0.459 ± 0.026**	0.437 ± 0.021	0.431 ± 0.017	0.447 ± 0.014
W_Acc	0.773 ± 007	**0.790 ± 0.010**	0.783 ± 0.008	0.781 ± 0.007	0.786 ± 0.006
		UTLC-apps			
		(LR=1e-6)	(LR=1e-5)	(LR=1e-5)	(LR=1e-4)
B_Acc	0.409 ± 0.009	**0.452 ± 0.010**	0.444 ± 0.005	0.430 ± 0.010	0.434 ± 0.012
W_F1	0.395 ± 0.010	**0.436 ± 0.010**	0.427 ± 0.004	0.406 ± 0.011	0.421 ± 0.012
W_Acc	0.764 ± 0.004	**0.781 ± 0.004**	0.778 ± 0.002	0.772 ± 0.004	0.774 ± 0.005
		UTLC-movies			
		(LR=1e-5)	(LR=1e-5)	(LR=1e-5)	(LR=1e-5)
B_Acc	0.364 ± 0.004	0.419 ± 0.008	0.400 ± 0.010	**0.424 ± 0.012**	0.414 ± 0.009
W_F1	0.350 ± 0.004	0.404 ± 0.009	0.386 ± 0.010	**0.407 ± 0.013**	0.399 ± 0.010
W_Acc	0.746 ± 0.001	0.768 ± 0.003	0.760 ± 0.004	**0.770 ± 0.005**	0.766 ± 0.004

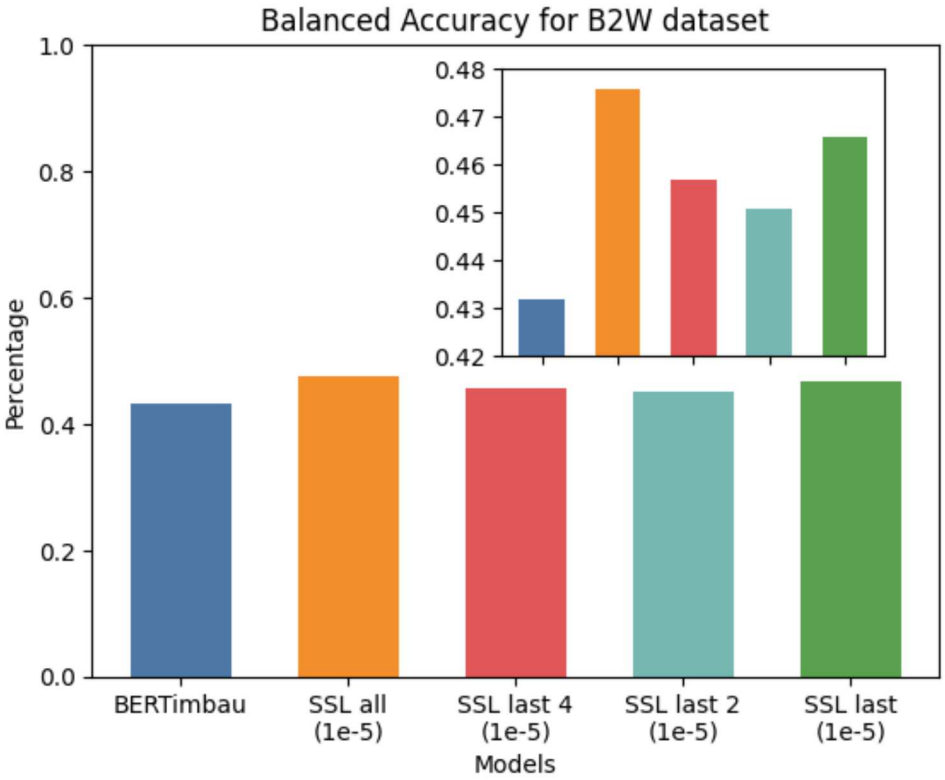


Figure 5.1: Difference, in terms of balanced accuracy, among the best self-supervised fine-tuning models and the baseline, for the B2W dataset.

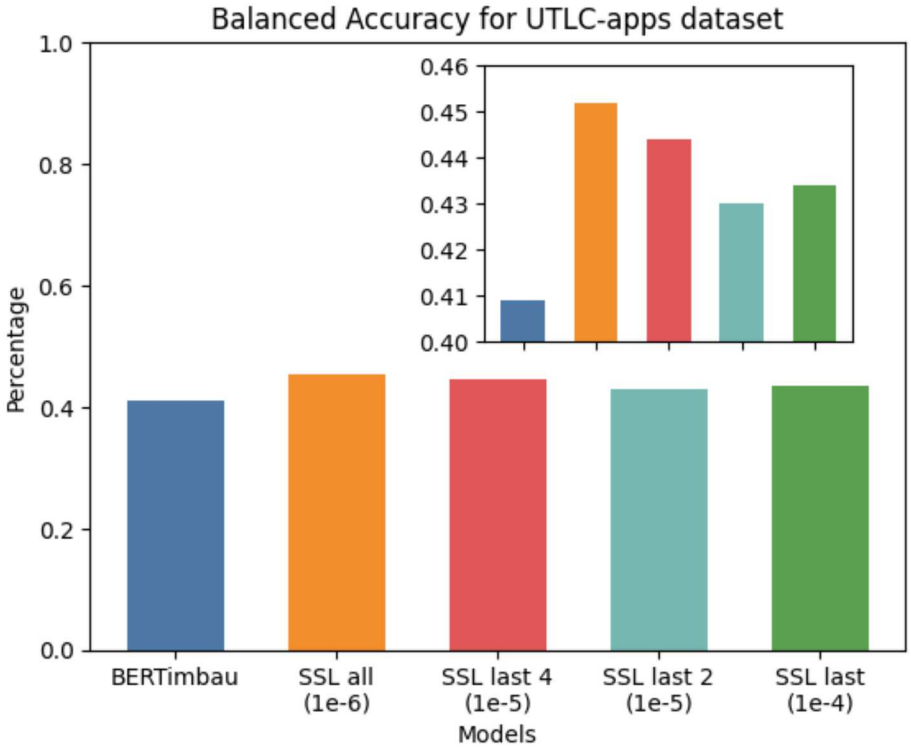


Figure 5.2: Difference, in terms of balanced accuracy, among the best self-supervised fine-tuning models and the baseline for the UTLC-apps dataset.

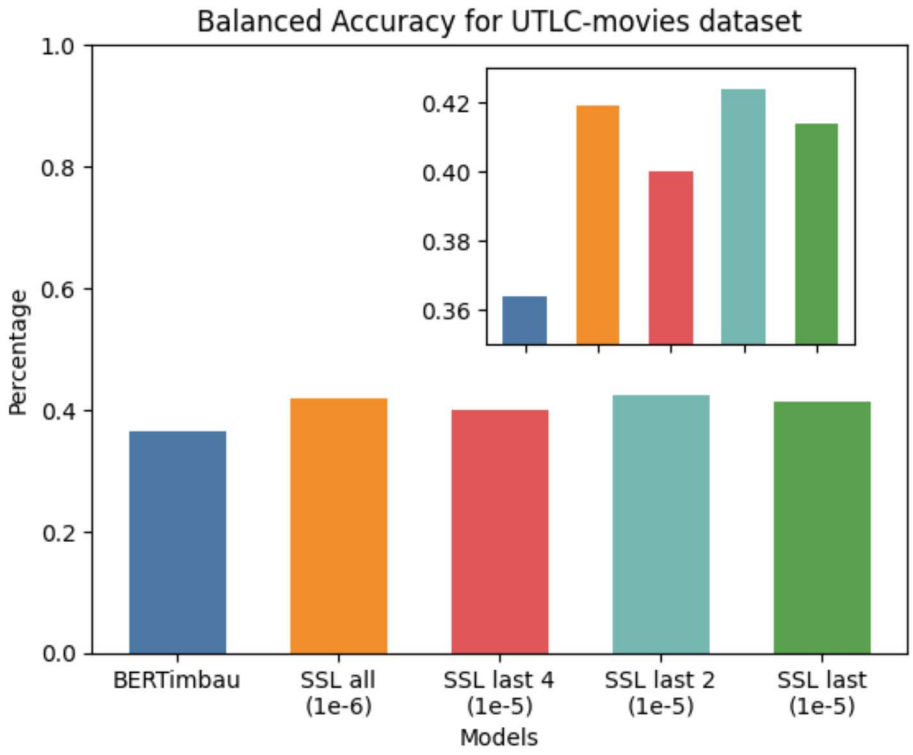
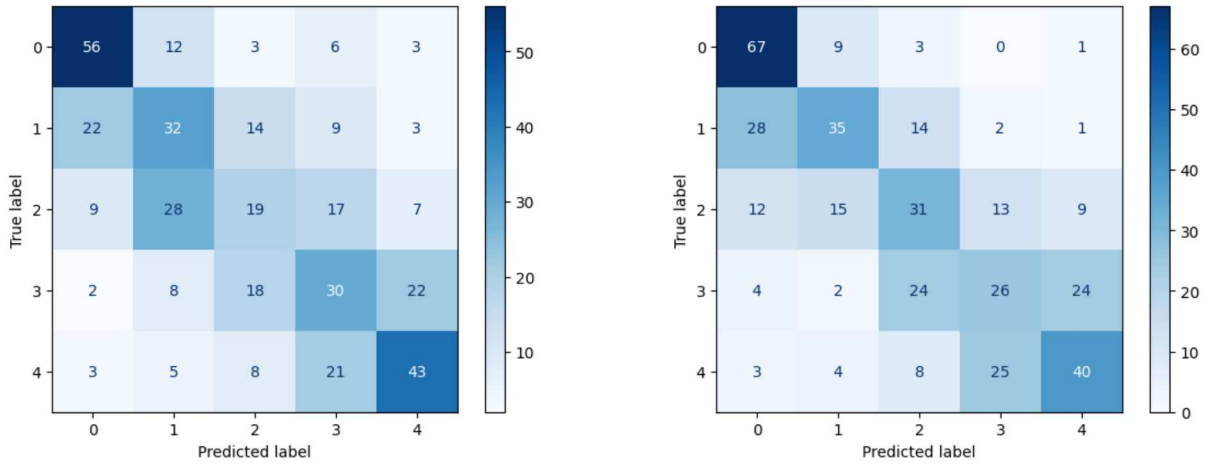


Figure 5.3: Difference, in terms of balanced accuracy, among the best self-supervised fine-tuning models and the baseline, for the UTLC-movies dataset.

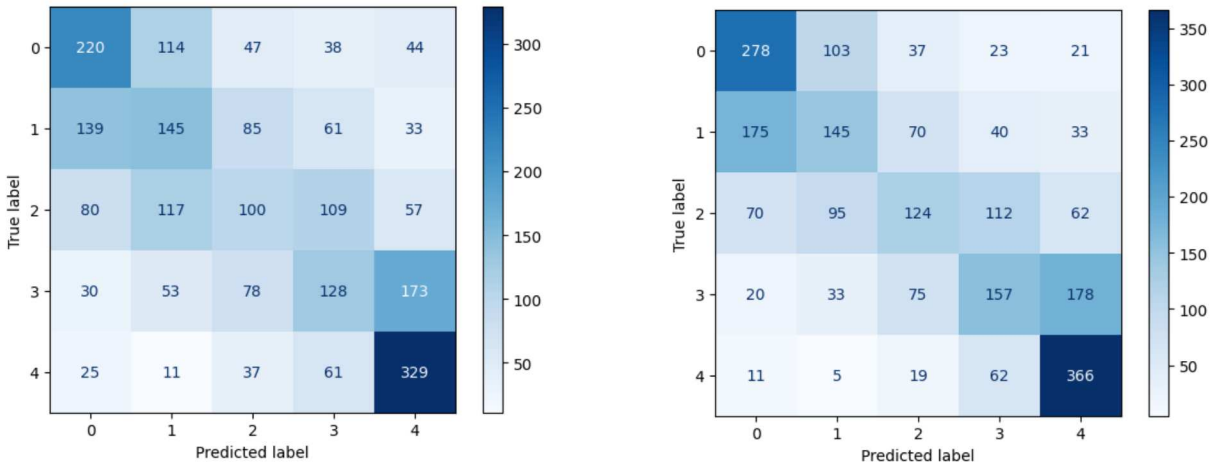
Next, we analyze, in confusion matrices depicted in Figures 5.4, 5.5, and 5.6, the results showed in Table 5.2. Noticeably, a higher dispersion of the data can be observed in the baseline model than in the models that have been subjected to self-supervised fine-tuning. The data are dispersed outside the main diagonal in the baseline, indicating more incorrect predictions. On the other hand, in the models adjusted using self-supervised fine-tuning, the data tend to cluster closer to the main diagonal, suggesting greater accuracy in the predictions.



(a) BERTimbau Base (baseline)

(b) Self-supervised fine-tuning model with all layers unfreeze and LR=1e-5.

Figure 5.4: Confusion matrices obtained by performing the downstream multiclass sentiment analysis task with both the BERTimbau Base model and the self-supervised fine-tuning Model when all layers are unfreeze and the learning rate is equal to 1e-5 for the B2W dataset.

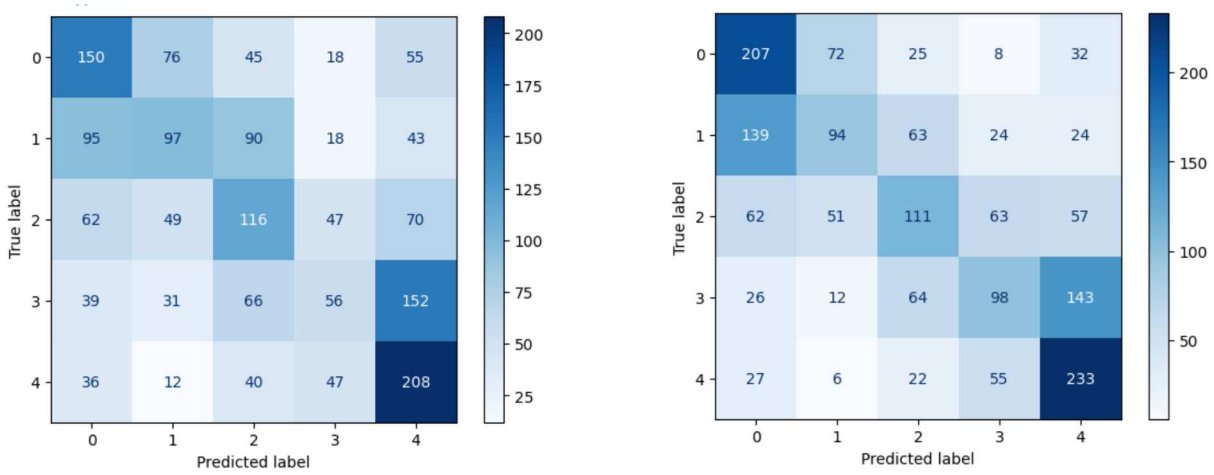


(a) BERTimbau Base (baseline)

(b) Self-supervised fine-tuning model with all layers unfreeze and LR=1e-6.

Figure 5.5: Confusion matrices obtained by performing the downstream multiclass sentiment analysis task with both the BERTimbau Base model and the self-supervised fine-tuning Model when all layers are unfreeze and the learning rate is equal to 1e-6 for the UTLC-apss dataset.

Furthermore, the adjusted models identify three distinguishable groups: the first group is composed of classes 0 and 1, the second group by class 2, and the third group by classes 3 and 4. This differentiation of groups improves the prediction, which is reflected in the results compared with the baseline model.



(a) BERTimbau Base (baseline)

(b) Self-supervised fine-tuning model with last 2 layers unfreeze and LR=1e-5.

Figure 5.6: Confusion matrices obtained by performing the downstream multiclass sentiment analysis task with both the BERTimbau Base model and the self-supervised fine-tuning Model when last 2 layers are unfreeze and the learning rate is equal to 1e-5 for the UTLC-movies dataset.

A Student's t-test was conducted, a widely used statistical method for comparing means and assessing the significance of differences between groups, to support the hypothesis that using only the last unfrozen layer with an appropriate learning rate in a self-supervised fine-tuning model yields significantly better results compared to a general model. This test is particularly suitable for evaluating the robustness of our results, as it provides a rigorous framework to determine whether the observed differences are likely due to random variation or represent a genuine improvement. Furthermore, when the resulting p-value is less than the established significance level (typically 0.05), it supports the alternative hypothesis, indicating that the differences are not due to chance but reflect a statistically significant effect. This test was applied to the results of the weighted F1 metric obtained by cross validation to compare the general BERTimbau model (model A) with the results of the best models tuned by self-supervised fine-tuning (model B), this for each data set.

Performing this statistical test is crucial in evaluating and comparing models, as it allows for determining whether the observed differences in performance are statistically significant. Specifically, we seek to demonstrate that fine-tuning the model in a particular context using the self-supervised fine-tuning approach provides a significant improvement compared to the general model. This analysis not only reinforces the validity of our methodology but also provides a solid basis to claim that the proposed approach is more effective for the specific task of multiclass sentiment analysis, ensuring that the results obtained are not due to random variability but to the effectiveness of the tuning process implemented.

Statistical significance test for the B2W dataset:

- Model A: BERTimbau Base for the multiclass sentiment analysis task;

- Model B: Self-supervised fine-tuning model with only the last layer unfreeze and LR 1e-5 applied to the multiclass sentiment analysis task;
- Null hypothesis (H_0): There is no significant difference in the weighted F1 metric between model A and model B;
- Alternative hypothesis (H_a): There is a significant difference in the weighted F1 metric between model A and model B;
- Significance level (α): 0.05.

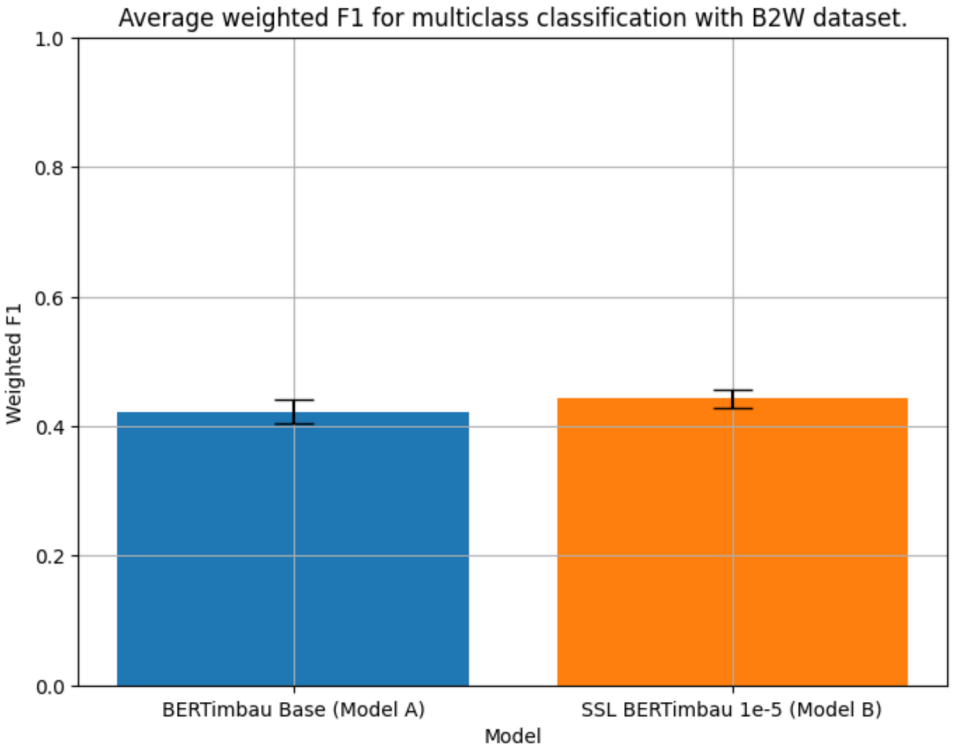


Figure 5.7: Average of the weighted F1 metric with its respective standard deviation when using both models for multiclass sentiment analysis on the B2W dataset.

The results obtained for the B2W data set (Figure 5.7) when comparing the BERTimbau Base model and the model with self-supervised fine-tuning, using only the last unfrozen layer and a learning rate of 1e-5, revealed a p-value of 0.0327. This significantly low value allows us to reject the null hypothesis, indicating that there is a statistically significant difference in performance between both models. By indicating that the null hypothesis is rejected, it tells us that model B, which in our case is the model that underwent self-supervised fine-tuning, does have a significant difference and therefore indicates that it is significantly better than model A, which is the BERTimbau baseline model.

Statistical significance test for the UTLC-apps dataset:

- Model A: BERTimbau Base for the multiclass sentiment analysis task;

- Model B: Self-supervised fine-tuning model with only the last layer unfreeze and LR 1e-4 applied to the multiclass sentiment analysis task;
- Null hypothesis (H_0): There is no significant difference in the weighted F1 metric between model A and model;
- Alternative hypothesis (H_a): There is a significant difference in the weighted F1 metric between model A and model B;
- Significance level (α): 0.05.

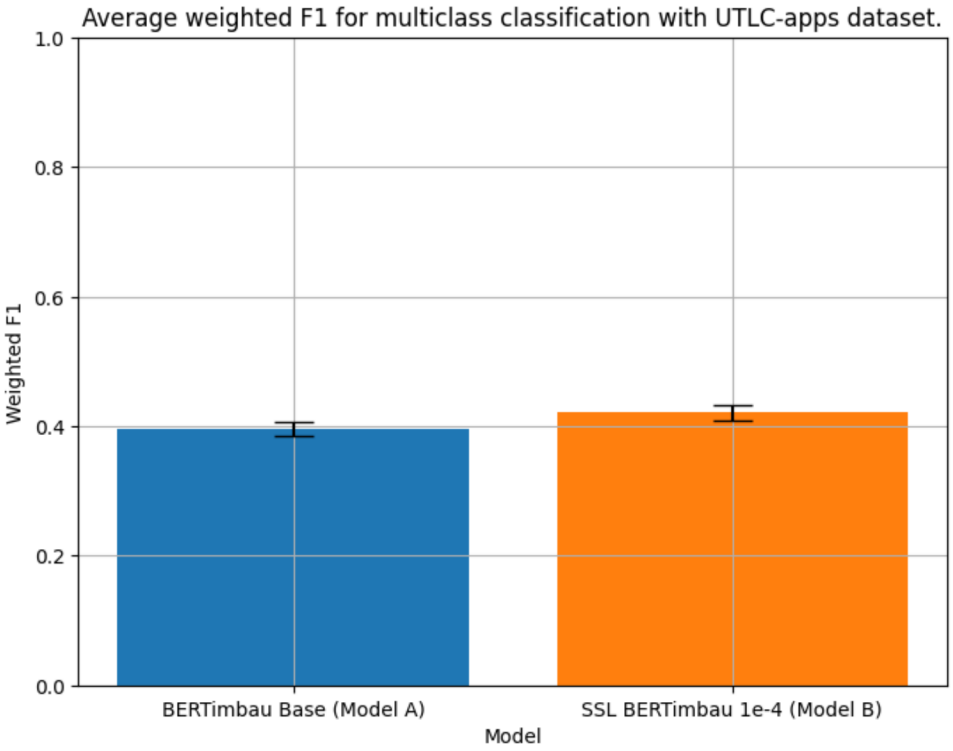


Figure 5.8: Average of the weighted F1 metric with its respective standard deviation when using both models for multiclass sentiment analysis on the UTLC-apps dataset.

The results obtained for the UTLC-apps data set (Figure 5.8) when comparing the BERTimbau Base model and the model with self-supervised fine-tuning, using only the last unfrozen layer and a learning rate of 1e-4, revealed a p-value of 0.0001. This significantly low value allows us to reject the null hypothesis, indicating that there is a statistically significant difference in performance between both models.

Finally, the statistical significance test setup for the UTLC-movies dataset was:

- Model A: BERTimbau Base for the multiclass sentiment analysis task;
- Model B: Self-supervised fine-tuning model with only the last layer unfreeze and LR 1e-5 applied to the multiclass sentiment analysis task;

- Null hypothesis (H_0): There is no significant difference in the weighted F1 metric between model A and model B;
- Alternative hypothesis (H_a): There is a significant difference in the weighted F1 metric between model A and model B;
- Significance level (α): 0.05.

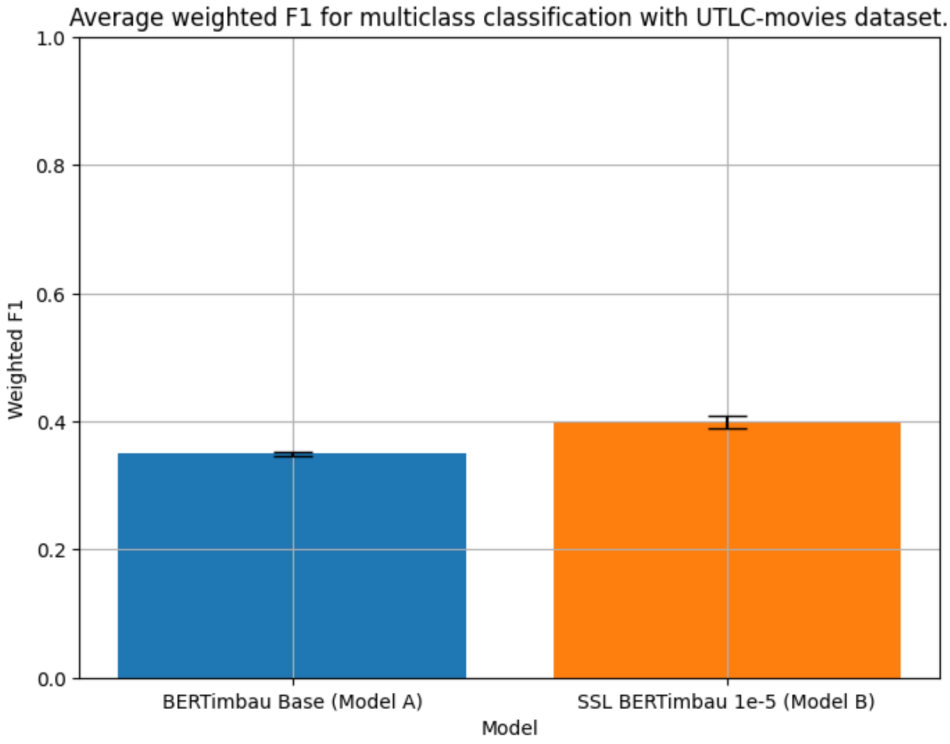


Figure 5.9: The figure shows the average of the weighted F1 metric with its respective standard deviation when using both models for multiclass sentiment analysis on the UTLC-movies dataset.

The results obtained for the UTLC-movies data set (Figure 5.9) when comparing the BERTimbau baseline model and the model with self-supervised fine-tuning, using only the last unfrozen layer and a learning rate of 1e-4, revealed a p-value of 1.6×10^{-7}. This significantly low value allows us to reject the null hypothesis, indicating that there is a statistically significant difference in performance between both models.

Chapter 6

Conclusions

This chapter covers this Master's dissertation main contributions, limitations, and potential future work. We aimed to expand the existing knowledge base and offer practical and applicable approaches for researchers and practitioners in natural language processing. This will help to encourage ongoing development and innovation in this important field.

6.1 Main Contributions of this Work

1. Using compact models: An important contribution of this work lies in the choice and use of a relatively small model such as BERTimbau to address the problem of multiclass sentiment analysis in Brazilian Portuguese. Compact models offer significant advantages in terms of efficiency and adaptability, making them particularly suitable for specific tasks where the complexity of larger models is not required. In this sense, choosing a smaller model facilitates adaptation to specific domains, improves interpretability, and reduces the risk of overfitting when working with more limited or highly specialized datasets. Furthermore, these models represent a more scalable and sustainable solution, especially in scenarios where efficiency in inference time and the possibility of deployment on resource-limited devices are key factors. This research highlights the relevance of compact models as an effective strategy to address specific linguistic problems, underlining their potential as a fundamental tool in the development of lightweight and specialized solutions in natural language processing (NLP).

2. Expanding the literature on self-supervised fine-tuning: This dissertation contributed by exploring the self-supervised fine-tuning approach in depth. This technique has received less attention than the more commonly used supervised fine-tuning approach. This research opens up new avenues for future exploration and applications in natural language processing by providing a detailed and thorough analysis of this technique for data sets in a low-resource language (Portuguese);

3. Demonstrating effectiveness in specific contexts: This research showed the efficacy of self-supervised fine-tuning in particular contexts, focusing on sentiment analysis in Brazilian Portuguese. The experiments showed that this approach can significantly

improve the performance of language models in specific contexts, offering a viable and effective alternative to traditional techniques;

4. Comparative analysis of self-supervised fine-tuning configurations: This work includes a detailed comparative study of different fine-tuning configurations, including varying unfreeze layers and using various learning rates. These analyses help to identify best practices and optimal parameters to maximize the performance of language models on specific tasks;

5. Release of source code for self-supervised fine-tuning: A relevant contribution of this project is releasing the source code used for self-supervised fine-tuning of the BERTimbau language model. By making the code available, we facilitate the replication of the experiments and provide a valuable tool for other researchers. This transparency and accessibility promote collaboration and advancement in the field, allowing others to apply these techniques to similar problems and develop new solutions based on the findings of this work.

6.2 Answers to the Research Questions

After accomplishing this research, we revisit the research questions (RQs) to answer them:

RQ 1 How to adapt the pretraining protocol of BERTimbau for a self-supervised fine-tuning procedure on that model?

Answer: From the literature review, we identified several approaches to adapt the pre-training protocol for a self-supervised fine-tuning procedure. In this discussion, we choose to use the vocabulary already established for the Portuguese language (provided by the BERTimbau model), because this choice avoids the increase in computational complexity that would result from the expansion of the vocabulary. This decision translates into greater efficiency, both in terms of training time and non-use of computing resources. Additionally, the use of a pre-trained model, such as BERTimbau, allows us to take advantage of the previously acquired knowledge of the base model, preserving its flexibility and generalization capacity for a wide range of tasks, without the need to initiate zero training. In our adaptation, we perform self-supervised fine tuning by unfreezing different numbers of litters and varying learner taxa (LR). The results indicate that defrosting only the last litter is not enough to achieve satisfactory performance, or that it is particularly relevant in environments with limited computing resources.

RQ 2 How does using unlabeled data in a specific domain in the proposed self-supervised fine-tuning approach impact BERTimbau's downstream performance and its generalization ability?

Answer: The experiments carried out will allow us to verify that the use of unlabeled data in a specific domain, within the self-supervised fine-tuning procedure, contributes significantly to improving performance in downstream tasks. In the

specific case of this dissertation, we apply to the downstream task of multiclass sentiment analysis. When we analyzed not only the validation metrics, but also the confusion matrices, it was possible to observe a clear improvement in the results. Specifically, we note that, as the use of self-supervised fine tuning, the main diagonal of the confusion matrix – which reflects the successes of the model – presents a greater concentration of values, in comparison with the scenario in which the self-supervised training does not I was used. This demonstrates that this approach does not only improve the precision of the model in subsequent tasks, but also positively impacts its generalization capacity. The form and use of unlabeled data in a specific domain showed an effective strategy to enhance the performance of the model, standing out as a promising solution for environments with a shortage of labeled data.

RQ 3 In which scenarios (e.g., data availability, type of domain) is it suitable to use that self-supervised fine-tuning approach?

Answer:

- Amount of data: We have observed that the more unlabeled data available, the better the results obtained. However, even with a moderate amount of unlabeled data, self-supervised fine-tuning can provide significant improvements in model performance.
- Computational capacity: Although the best results were obtained by unfreezing all layers of the model, we found that unfreezing just one layer during self-supervised fine-tuning already resulted in satisfactory performance. This strategy is particularly advantageous for organizations with limited computational resources, as it reduces the processing demand.
- Learning rate (LR): A critical point identified is the inversely proportional relationship between the number of unfrozen layers and the ideal value of the learning rate. When choosing to unfreeze only one layer, it is not recommended to use a very small learning rate, as this may harm the training.
- Recommended LR: The learning rate that showed best results in most experiments was 1e-5, being a robust choice for different scenarios.

6.3 Limitations

During the development of this study, we faced some limitations that impacted both the scope and depth of the experiments conducted:

- Constraints on computational resources and experimental repetitions: During the project's duration, one of the primary challenges was the significant computational cost associated with performing multiple experimental repetitions to ensure the robustness and validity of our results. The need for extensive experimentation,

combined with the time constraints inherent in the research timeline, required careful optimization of training pipelines and resource allocation to maximize efficiency without compromising the quality of the findings.

- Limitation in the availability of labeled datasets: Another important limitation was the lack of labeled datasets for multiclass sentiment analysis in Brazilian Portuguese. The availability of such data is crucial to assess the generalization and performance of our proposal in specific linguistic contexts.

6.4 Possibilities for Future Works

To strengthen and extend the findings of this research, we propose several directions for future work:

- Evaluation with larger and state-of-the-art (SOTA) models: Testing with larger, state-of-the-art models will allow us to verify whether our proposal for self-supervised fine-tuning and varying the number of unfreeze layers is equally effective on models of higher capacity and complexity, which could provide even better performance on various natural language processing tasks.

- Incorporating explainability techniques: Adding explainability methods to our proposal to understand how our model is learning and making decisions. Interpretability is crucial for applications where model transparency is essential and would help identify the reasons behind the model's predictions, allowing for better validation and confidence in its use.

- Application of the methodology to real-world problems: Initially, we intended to apply our methodology using data of a company that sponsored this dissertation. However, due to amount of data and time constraints, carrying out this activity was not possible until the end of this dissertation. Therefore, it is proposed as future work to develop this test, which will allow validation of the applicability of our methodology in a real business environment and will contribute to solving practical problems.

In summary, these future works will expand the scope and applicability of our proposal and contribute to the advancement of knowledge in machine learning and natural language processing.

6.5 Final Remarks

This research has focused on the development and evaluation of advanced self-supervised fine-tuning techniques applied to natural language models, specifically in the context of multiclass sentiment analysis in Brazilian Portuguese. Throughout this study, we have explored multiple configurations to improve the performance of pretrained models in specific domains.

First, it has been shown that the self-supervised fine-tuning technique can significantly improve the ability of language models to adapt to new domains, depending on the number of unfrozen layers and the appropriate learning rate (LR). This finding underlines the importance of carefully selecting these hyperparameters to maximize the effectiveness of the model in specific tasks.

A key finding from Table 5.1 is that unfreezing only the last layer of the model across the three datasets used in our experiments (B2W, UTLC-apps, UTLC-movies) and applying an appropriate learning rate yields better results than the baseline. This approach reduces the computational cost during training and enables researchers with limited resources to achieve robust performance without unfreezing all model layers. Additionally, the statistical test confirms that self-supervised fine-tuning with only the last layer unfrozen produces significantly better results compared to the general model. This outcome supports the hypothesis that self-supervised fine-tuning can be effectively tailored to specific domains, even when employing simplified model configurations.

We also observed that using a very small learning rate, like 1e-6, is not advisable when unfreezing only one layer. Our results show that this value is inversely proportional to the number of unfrozen layers, resulting in suboptimal performance. Regardless of the number of unfrozen layers, the learning rate of 1e-5 proved to be a robust choice, consistently providing better results than those obtained with a general model.

Table 5.1 also reveals that as the dataset size increases, the performance metrics of the general model tend to decrease. In contrast, the difference in performance between the self-supervised fine-tuning and baseline models increases. This finding suggests that self-supervised fine-tuning is particularly beneficial for larger datasets, although further testing would be needed to confirm this observation.

The confusion matrices of the Figures 5.4, 5.5, and 5.6 show that self-supervised fine-tuning models significantly improve data classification, concentrating more values on the main diagonal and showing better differentiation between classes. This evidences the effectiveness of the fine-tuning process in improving the model's performance in the multiclass sentiment analysis task.

Finally, this project has raised new directions for future research, including applying our methodology to larger and more modern models, integrating explainability techniques, and validating in real business environments. These future works will not only broaden the scope of our research but also contribute to the development of more accurate and efficient natural language models.

Bibliography

[1] Abdullateef I. Almudaifer, Whitney Covington, Ja Mor Hairston, Zachary Deitch, Ankit Anand, Caleb M. Carroll, Estera Crisan, William Bradford, Lauren A. Walter, Ellen F. Eaton, Sue S. Feldman, and John D. Osborne. Multi-task transfer learning for the prediction of entity modifiers in clinical text: application to opioid use disorder case detection. *Journal of biomedical semantics*, 15:11, 12 2024.

[2] Wissam Antoun, Fady Baly, and Hazem Hajj. Arabert: Transformer-based model for arabic language understanding, 2 2020.

[3] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, pages 3615–3620, 3 2019.

[4] Pratik Bhavsar. Transfer learning in nlp. let's not start from zero, 2019.

[5] Kay H. Brodersen, Cheng Soon Ong, Klaas E. Stephan, and Joachim M. Buhmann. The balanced accuracy and its posterior distribution. *Proceedings - International Conference on Pattern Recognition*, pages 3121–3124, 2010.

[6] Brown. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 2020-December, 5 2020.

[7] Diedre Carmo, Marcos Piau, Israel Campiotti, Rodrigo Nogueira, and Roberto Lotufo. Ptt5: Pretraining and validating the t5 model on brazilian portuguese data, 8 2020.

[8] José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. Spanish pre-trained bert model and evaluation month = 8, data, 2023.

[9] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. Legal-bert: The muppets straight out of law school. *Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020*, pages 2898–2904, 10 2020.

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *37th International Conference on Machine Learning, ICML 2020*, PartF168147-3:1575–1585, 2 2020.

[11] Nicholas Kluge Corrêa, Sophia Falk, Shiza Fatimah, Aniket Sen, and Nythamar de Oliveira. Teenytinyllama: open-source tiny language models trained in brazilian portuguese. *Machine Learning with Applications*, 16:100558, 1 2024.

[12] Pablo Da, Costa Matheus, Pavan Wesley, Dos Santos, Samuel Da, and Silva Ivandré Paraboni. Bertabaporu: Assessing a genre-specific language model for portuguese nlp, 2023.

[13] João Vitor Andrioli de Souza, Lucas Emanuel Silva e Oliveira Julien Knafou, Yohan Bonescki Gumiel Jenny Copara, Lucas Ferro Antunes de Oliveira, Emerson Cabrera Paraiso, Douglas Teodoro, and Cláudia Maria Cabral Moro Barra Elisa Terumi Rubel Schneider. Biobertpt -a portuguese neural language model for clinical named entity recognition | request pdf, 2020.

[14] Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. Bertje: A dutch bert model, 12 2019.

[15] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 1:4171–4186, 2018. doi: 10.48550/arxiv.1810.04805.

[16] Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, Dinani Amorim, and Amorim Fernández-Delgado. Do we need hundreds of classifiers to solve real world classification problems? *The Journal of Machine Learning Research*, 15:3133–3181, 1 2014.

[17] Jorge A Wagner Filho, Rodrigo Wilkens, Marco Idiart, and Aline Villavicencio. The brwac corpus: A new open resource for brazilian portuguese, 2022.

[18] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision, 2009.

[19] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[20] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016. `http://www.deeplearningbook.org`.

[21] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3:24, 7 2020.

[22] Jie Gui, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao. A survey on self-supervised learning: Algorithms, applications, and future trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14, 1 2023.

[23] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *ICLR 2021 - 9th International Conference on Learning Representations*, 6 2020.

[24] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9:1735–1780, 11 1997.

[25] Jimin Hong, Taehee Kim, Hyesu Lim, and Jaegul Choo. Avocado: Strategy for adapting vocabulary to downstream domain. *EMNLP 2021 - 2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, pages 4692–4700, 10 2021.

[26] John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

[27] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 1:328–339, 1 2018.

[28] Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ICLR 2022 - 10th International Conference on Learning Representations*, 6 2021.

[29] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission, 4 2019.

[30] Herbert Jaeger and Harald Haas. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667):78–80, 2004.

[31] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:4037–4058, 2 2019.

[32] Dan Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, Upper Saddle River, NJ, 2nd edition, 2009.

[33] Katikapalli Subramanyam Kalyan, Ajit Rajasekharan, and Sivanesan Sangeetha. Ammus : A survey of transformer-based pretrained models in natural language processing, 8 2021.

[34] John D. Kelleher, Brian. Mac Namee, and Aoife D'Arcy. *Fundamentals of machine learning for predictive data analytics : algorithms, worked examples, and case studies*. The MIT Press; 1st edition (July 24, 2015), 7 2015.

[35] Yoon Kim. Convolutional neural networks for sentence classification. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 1746–1751, 8 2014.

[36] Fajri Koto, Jey Han Lau, and Timothy Baldwin. Indobertweet: A pretrained language model for indonesian twitter with effective domain-specific vocabulary initialization. *EMNLP 2021 - 2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, pages 10660–10668, 9 2021.

[37] Fajri Koto, Afshin Rahimi, Jey Han Lau, and Timothy Baldwin. Indolem and indobert: A benchmark dataset and pre-trained language model for indonesian nlp. *COLING 2020 - 28th International Conference on Computational Linguistics, Proceedings of the Conference*, pages 757–770, 11 2020.

[38] Anastasios Lamproudis., Aron Henriksson., and Hercules Dalianis. Vocabulary modifications for domain-adaptive pretraining of clinical language models. In *Proceedings of the 15th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2022) - HEALTHINF*, pages 180–188. INSTICC, SciTePress, 2022.

[39] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *8th International Conference on Learning Representations, ICLR 2020*, 9 2019.

[40] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–444, 5 2015.

[41] Hwabin Lee, Sua Jeong, Seogyeong Cho, and Eunjung Choi. Visualization technology and deep-learning for multilingual spam message detection. *Electronics (Switzerland)*, 12, 2 2023.

[42] Jaejun Lee, Raphael Tang, Jimmy Lin, and David R Cheriton. What would elsa do? freezing layers during transformer fine-tuning. *1*, 11 2019.

[43] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, September 2019.

[44] Alexander Ligthart, Cagatay Catal, and Bedir Tekinerdogan. Systematic reviews in sentiment analysis: a tertiary study. *Artificial Intelligence Review 2021 54:7*, 54:4997–5053, 3 2021.

[45] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *8th International Conference on Learning Representations, ICLR 2020*, 8 2019.

[46] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv:1907.11692*, 2019.

[47] Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and Philip S. Yu. Graph self-supervised learning: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35:5879–5900, 2 2021.

[48] Yanying Mao, Qun Liu, and Yu Zhang. Sentiment analysis methods, applications, and challenges: A systematic literature review. *Journal of King Saud University - Computer and Information Sciences*, 36:102048, 4 2024.

[49] Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. Camembert: a tasty french language model. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, 11 2019.

[50] Tom M Mitchell. *Machine Learning*. McGraw-Hill Education; 1st edition (March 1, 1997), 1997.

[51] Brian Keith Norambuena, Exequiel Fuentes Lettura, and Claudio Meneses Villegas. Sentiment analysis and opinion mining applied to scientific paper reviews. *Intelligent Data Analysis*, 23:191–214, 2019.

[52] Juri Opitz and Sebastian Burst. Macro f1 and macro f1, 11 2019.

[53] Matthew J. Page, Joanne E. McKenzie, Patrick M. Bossuyt, Isabelle Boutron, Tammy C. Hoffmann, Cynthia D. Mulrow, Larissa Shamseer, Jennifer M. Tetzlaff, Elie A. Akl, Sue E. Brennan, Roger Chou, Julie Glanville, Jeremy M. Grimshaw, Asbjørn Hróbjartsson, Manoj M. Lalu, Tianjing Li, Elizabeth W. Loder, Evan Mayo-Wilson, Steve McDonald, Luke A. McGuinness, Lesley A. Stewart, James Thomas, Andrea C. Tricco, Vivian A. Welch, Penny Whiting, and David Moher. The prisma 2020 statement: An updated guideline for reporting systematic reviews. *The BMJ*, 372, 3 2021.

[54] Andrei Paleyes, Raoul-Gabriel Urma, and Neil D. Lawrence. Challenges in deploying machine learning: a survey of case studies. *ACM Computing Surveys*, 55, 2020. doi: 10.1145/3533378.

[55] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359, 2010.

[56] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *ACL-05 - 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 115–124, 6 2005.

[57] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2:1–135, 2008.

[58] Ramon Pires, Hugo Abonizio, Thales Sales Almeida, and Rodrigo Nogueira. Sabiá: Portuguese large language models. *Lecture Notes in Computer Science (including*

subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 14197 LNAI:226–240, 4 2023.

[59] Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual bert? *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 4996–5001, 6 2019.

[60] Federico Alberto Pozzi, Elisabetta Fersini, Enza Messina, and Bing Liu. Challenges of sentiment analysis in social networks: An overview. *Sentiment Analysis in Social Networks*, pages 1–11, 1 2017.

[61] Guo-Jun Qi and Mubarak Shah. Adversarial pretraining of self-supervised deep networks: Past, present and future, 10 2022.

[62] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training, 2018.

[63] Alec Radford, Jeff Wu, R. Child, D. Luan, Dario Amodei, and I. Sutskever. Language models are unsupervised multitask learners, 2019.

[64] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67, 10 2019.

[65] L. Real, M. Oshiro, and A. Mafra1. B2w-reviews01 an open product reviews corpus. In XII Symposium in Information and Human Language Technology. 200–208, 2019.

[66] Raghavendra Reddy and U. M. Ashwin Kumar. Multi-class sentiment analysis over social networking applications using text and emoji-based features. *Proceedings - International Conference on Applied Artificial Intelligence and Computing, ICAAIC 2022*, pages 829–834, 2022.

[67] João Rodrigues, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso, and Tomás Osório. Advancing neural encoding of portuguese with transformer albertina pt-*. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 14115 LNAI:441–453, 5 2023.

[68] Rafael B.M. Rodrigues, Pedro I.M. Privatto, Gustavo José de Sousa, Rafael P. Murari, Luis C.S. Afonso, João P. Papa, Daniel C.G. Pedronette, Ivan R. Guilherme, Stephan R. Perrout, and Aliel F. Riente. Petrobert: A domain adaptation language model for oil and gas applications in portuguese. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 13208 LNAI:101–109, 2022.

[69] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386–408, 11 1958.

[70] Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. Transfer learning in natural language processing. *Proceedings of the 2019 Conference of the North*, pages 15–18, 2019.

[71] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[72] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 10 2019.

[73] Madeline C. Schiappa, Yogesh S. Rawat, and Mubarak Shah. Self-supervised learning for videos: A survey. *ACM Computing Surveys*, 55, 6 2022.

[74] Elisa Terumi Rubel Schneider, Joao Vitor Andrioli De Souza, Yohan Bonescki Gumiel, Claudia Moro, and Emerson Cabrera Paraiso. A gpt-2 language model for biomedical texts in portuguese. *Proceedings - IEEE Symposium on Computer-Based Medical Systems*, 2021-June:474–479, 6 2021.

[75] Raquel Silveira, Caio Ponte, Vitor Almeida, Vládia Pinheiro, and Vasco Furtado. Legalbert-pt: A pretrained language model for the brazilian portuguese legal domain. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 14197 LNAI:268–282, 2023.

[76] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45:427–437, 7 2009.

[77] Rogério Figueredo De Sousa, Henrico Bertini Brum, Maria Das Graças, and Volpe Nunes. A bunch of helpfulness and sentiment corpora in brazilian portuguese. Symposium in Information and Human Language Technology - STI, 2019. url: repositorio.usp.br/item/002971317.

[78] Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. BERTimbau: Pretrained BERT Models for Brazilian Portuguese. In *Lecture Notes in Computer Science*, volume 12319 LNAI, pages 403–417. Springer Science and Business Media Deutschland GmbH, 2020.

[79] Wen Tai, H. T. Kung, Xin Dong, Marcus Comiter, and Chang Fu Kuo. exbert: Extending pre-trained models with domain-specific vocabulary under constrained training resources. *Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020*, pages 1433–1439, 2020.

[80] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017-December:5999–6009, 6 2017.

[81] Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. Multilingual is not enough: Bert for finnish, 12 2019.

[82] Alex Waibel, Toshiyuki Hanazawa, Geoffrey Hinton, Kiyohiro Shikano, and Kevin J. Lang. Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(3):328–339, 1989.

[83] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34, 2019. doi: 10.1145/3386252.

[84] Karl Weiss, Taghi M. Khoshgoftaar, and Ding Ding Wang. A survey of transfer learning. *Journal of Big Data*, 3:1–40, 12 2016.

[85] Joseph Weizenbaum. Eliza-a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9:36–45, 1 1966.

[86] Jian Yang, Xinyu Hu, Weichun Huang, Hao Yuan, Yulong Shen, and Gang Xiao. Advancing domain adaptation of bert by learning domain term semantics. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 14120 LNAI:12–24, 2023.

[87] Yi Yang, Mark Christopher, Siy Uy, and Allen Huang. Finbert: A pretrained language model for financial communications, 6 2020.

[88] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing, 8 2017.

[89] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Jundong Li, and Zi Huang. Self-supervised learning for recommender systems: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 36:335–355, 3 2022.

[90] Kanwal Zahoor, Narmeen Zakaria Bawany, and Soomaiya Hamid. Sentiment analysis and classification of restaurant reviews using machine learning. *Proceedings - 2020 21st International Arab Conference on Information Technology, ACIT 2020*, 11 2020.

[91] Lei Zhang, Shuai Wang, and Bing Liu. Deep learning for sentiment analysis : A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8, 1 2018.

Appendix A

Split Ratio Experiments

These additional experiments were performed in order to evaluate the impact of the amount of data on the final results and to obtain a rough reference on the amount of data needed to achieve good results. To do so, two partitions of the B2W dataset were used, corresponding to 50% and 20% of the total data used in the self-supervised training.

Table A.1: Results obtained when training the model with different configurations of unfrozen layers (all layers, last 4, last 2, and only the last one), varying the learning rate (1e-4, 1e-5, 1e-6), and using only 50% of the total data used in the initial self-supervised training with the B2W dataset. The model resulting from this process was applied to a downstream multiclass sentiment analysis task, evaluated by cross-validation, using the Balanced Accuracy, Weighted F1, and Weighted Accuracy metrics. Highlighting in green the column with the best results.

	Multiclass sentiment analysis for B2W												
		SSL with ALL layers defreeze			SSL with 4 layers defreeze			SSL with 2 layers defreeze			SSL with 1 layers defreeze		
	BERT imbau	**1e-4**	1e-5	1e-6	1e-4	**1e-5**	1e-6	1e-4	**1e-5**	1e-6	**1e-4**	1e-5	1e-6
B_Acc	0.432	**0.454**	0.447	0.441	0.435	**0.450**	0.445	0.387	**0.436**	0.393	**0.438**	0.436	0.381
W_F1	0.422	**0.438**	0.434	0.426	0.421	**0.432**	0.429	0.372	**0.410**	0.379	**0.418**	0.413	0.366
W_Acc	0.773	**0.781**	0.779	0.776	0.774	**0.780**	0.778	0.755	**0.774**	0.757	**0.775**	0.775	0.752

Table A.2: Results obtained when training the model with different configurations of unfrozen layers (all layers, last 4, last 2, and only the last one), varying the learning rate (1e-4, 1e-5, 1e-6), and using only 20% of the total data used in the initial self-supervised training with the B2W dataset. The model resulting from this process was applied to a downstream multiclass sentiment analysis task, evaluated by cross-validation, using the Balanced Accuracy, Weighted F1, and Weighted Accuracy metrics. Highlighting in green the column with the best results.

	Multiclass sentiment analysis for B2W												
		SSL with ALL layers defreeze			SSL with 4 layers defreeze			SSL with 2 layers defreeze			SSL with 1 layers defreeze		
	BERT imbau	**(1e-4)**	(1e-5)	(1e-6)	**(1e-4)**	(1e-5)	(1e-6)	(1e-4)	**(1e-5)**	(1e-6)	**(1e-4)**	(1e-5)	(1e-6)
B_Acc	0.432	**0.445**	0.442	0.441	**0.462**	0.440	0.437	0.400	**0.416**	0.372	**0.425**	0.420	0.362
W_F1	0.422	**0.433**	0.431	0.428	**0.450**	0.421	0.420	0.378	**0.396**	0.359	**0.407**	0.403	0.350
W_Acc	0.773	**0.778**	0.777	0.776	**0.785**	0.776	0.775	0.760	**0.766**	0.749	**0.770**	0.768	0.745

Table A.3: Results of the BERTimbau model (our baseline model), the best result obtained with 100% of the training data in self-supervised fine-tuning, the best result with 50% of the training data in self-supervised fine-tuning, and the best result with 20% of the training data in self-supervised fine-tuning. These results show that it is possible to obtain remarkable performance using only 50% of the data, while by reducing the amount to 20%, the results start to approach the baseline model's performance, indicating a decrease in improvement.

	Multiclass sentiment analysis for B2W			
	BERTimbau	100% dataset SSL with ALL layers defreeze 1e-5	50% dataset SSL with ALL layers defreeze 1e-4	20% dataset SSL with 4 layers defreeze 1e-4
B_Acc	0.432	0.456	0.454	0.435
W_F1	0.422	0.459	0.438	0.421
W_Acc	0.773	0.790	0.781	0.774