

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

ModBERTBr: A ModernBERT-based Model for Brazilian Portuguese

Wallace Ben Teng Lin Wu

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia da Computação

Orientador
Prof. Dr. Luis Paulo Faina Garcia

Brasília
2025

CIP - Catalogação na Publicação

Ben Teng Lin Wu, Wallace.
B959m ModBERTBr: A ModernBERT-based Model for Brazilian Portuguese / Wallace Ben Teng Lin Wu;

Orientador: Luis Paulo Faina Garcia. Brasília, 2025.
21 f.

Trabalho de Conclusão de Curso (Graduação - Engenharia de Computação) Universidade de Brasília, 2025.

1. BERT. 2. ModernBERT. 3. Brazilian Portuguese. 4. Foundation Models. 5. Large Language Models. I. Faina Garcia, Luis Paulo, orient. II. Título.



Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

ModBERTBr: A ModernBERT-based Model for Brazilian Portuguese

Wallace Ben Teng Lin Wu

Monografia apresentada como requisito parcial
para conclusão do Curso de Engenharia da Computação

Prof. Dr. Luis Paulo Faina Garcia (Orientador)
CIC/UnB

Prof. Dr. Marcelo Keese Albertini
FACOM/UFU

Prof. Dr. Li Weigang
CIC/UnB

Prof. Dr. Guilherme Novaes Ramos
Coordenador do Curso de Engenharia da Computação

Brasília, 9 de Julho de 2025

Dedicatória

Primeiramente, gostaria de expressar minha mais sincera gratidão aos meus pais:

Lin Xiuping 林秀坪 e *Wu Zhaoxiao* 吴昭孝.

Sou eternamente grato por todo o amor, o cuidado, a dedicação e os sacrifícios que foram realizados para que eu pudesse estar aqui hoje. Na trajetória da minha vida, os primeiros e mais desafiadores passos já foram concluídos pelos meus pais. Desse modo, eu apenas precisei dar continuidade a esse percurso e espero estar trilhando um caminho que os deixe orgulhosos.

Gostaria de expressar minha profunda gratidão também aos demais membros da minha família. Adoro minha irmã, meus sobrinhos e meus primos. E espero que, um dia no futuro, quando meus sobrinhos crescerem, eles possam ler este trabalho e, quem sabe, se sintam inspirados.

Finalmente, quero agradecer aos meus amigos, os quais me guiam precisamente, me apoiam continuamente e constituem minha base nos momentos mais desafiadores. Agradeço por ouvirem meus desabafos sempre que precisava; pelos conselhos certeiros quando estava perdido; pela imensa paciência em meus momentos mais insuportáveis; e pela brutal honestidade, mesmo quando eu não queria ouvir.

Dessa forma, dedico este trabalho a todas essas pessoas fundamentais e importantíssimas na minha vida.

Agradecimentos

Primeiramente, sou imensamente grato pelo apoio e suporte oferecidos pelo *professor Dr. Luis Paulo Faina Garcia*. Agradeço não apenas pelos vastos conhecimentos técnicos e orientações acadêmicas, mas também pelo encorajamento e pela paciência dedicados à realização deste trabalho, além dos valiosos conselhos para minha vida pessoal e profissional.

Gostaria de expressar minha sincera gratidão a todos os professores e profissionais que contribuíram para a minha trajetória. Se hoje possuo algum conhecimento, devo isso a todos aqueles que me ensinaram. Em outras palavras, "Se eu vi mais longe, foi por estar sobre os ombros de gigantes", como disse Sir Isaac Newton.

Notoriamente, gostaria de expressar minha gratidão ao *professor Dr. Guilherme Novaes Ramos*, que me apresentou ao fascinante mundo da pesquisa acadêmica e às competições de programação, pelas quais me apaixonei. Agradeço também ao *professor Dr. Vinicius Ruela Pereira Borges*, que foi um dos pilares fundamentais para as maratonas de programação e pela incrível comunidade que se consolidou delas. E ao *professor Dr. Li Weigang*, que foi um mentor inestimável, acompanhando-me desde o início da universidade até o final com ensinamentos precisos.

Agradeço imensamente todo o patrocínio e a infraestrutura que a Universidade de Brasília me proporcionou. É uma instituição incrível, que nunca deixou a desejar e permitiu que eu desenvolvesse ao máximo minhas pesquisas e habilidades.

Sou também grato pelo patrocínio do projeto HPC Fund da AMD, que disponibilizou o equipamento computacional fundamental para a execução deste trabalho de conclusão.

Finalmente, é importante mencionar que o presente trabalho foi realizado com o apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), por meio do acesso ao Portal de Periódicos.

Resumo

Avanços recentes em Inteligência Artificial (IA) têm impulsionado a rápida adoção dessas tecnologias em diversos setores e aspectos da vida cotidiana, impulsionados significativamente pela evolução dos Large Language Models (LLMs). Um modelo fundamental nesse domínio é o Bidirectional Encoder Representations from Transformers (BERT), cuja variante de estado-da-arte, o ModernBERT, se destaca em eficiência e desempenho. No entanto, o ModernBERT é limitado ao Inglês. Além disso, existe uma lacuna significativa na disponibilidade de modelos especificamente projetados para o Português Brasileiro, com as opções existentes sendo desatualizadas em relação às metodologias e tecnologias contemporâneas. Este artigo aborda essa lacuna ao introduzir o ModBERTBr, um novo modelo pré-treinado baseado na arquitetura do ModernBERT, especificamente adaptado para o Português Brasileiro e incorporando pesquisas e tecnologias de ponta. O termo *mod* refere-se a "Moderno", enquanto *Br* representa o "Português Brasileiro". O ModBERTBr foi pré-treinado do zero em conjuntos de dados meticulosamente tratados, obtidos a partir da Wikipedia e do BrWac, garantindo alta qualidade dos dados. Além de incorporar inovações do ModernBERT, como um mecanismo de atenção alternada, embeddings posicionais rotativos (rotary positional embeddings) e Flash Attention 2, também foi desenvolvido um tokenizador personalizado que utiliza a abordagem do modelo unigram. O ModBERTBr foi avaliado em comparação com múltiplas referências, incluindo o BERTimbau e o BERTugues, por meio de avaliações intrínsecas, como na tarefa de Masked Language Modeling (MLM), e tarefas mais específicas, como a de Semantic Textual Similarity (STS), Recognizing Textual Entailment (RTE) e Named Entity Recognition (NER). Os resultados destacam a eficácia do ModBERTBr, demonstrando melhorias consistentes nas tarefas de STS e NER e desempenho competitivo em RTE em comparação com os modelos predecessores.

Palavras-chave: BERT, ModernBERT e Português Brasileiro

Abstract

Recent advancements in Artificial Intelligence (AI) have spurred rapid adoption across various sectors and aspects of daily life, significantly propelled by the evolution of Large Language Models (LLMs). A pivotal model in this domain is the Bidirectional Encoder Representations from Transformers (BERT), whose state-of-the-art variant, ModernBERT, excels in efficiency and performance. However, ModernBERT is limited to the English language. Furthermore, a notable gap exists in the availability of models specifically designed for Brazilian Portuguese, with the existing options being outdated regarding contemporary methodologies and technologies. This paper addresses this notable gap by introducing ModBERTBr, a novel pre-trained model based on the ModernBERT architecture, explicitly tailored for Brazilian Portuguese and incorporating cutting-edge research and technologies. The term *Mod* refers to Modern, while *Br* stands for Brazilian Portuguese. ModBERTBr was pre-trained from scratch on meticulously curated datasets composed of Wikipedia and BrWac, ensuring high data quality. In addition to incorporating key innovations from ModernBERT, such as an alternating attention mechanism, rotary positional embeddings, and Flash Attention 2, a custom tokenizer employing the unigram model approach was also developed. ModBERTBr was assessed against multiple baselines, including BERTimbau and BERTugues, through intrinsic evaluations such as Masked Language Modeling (MLM) and downstream tasks such as Semantic Textual Similarity (STS), Recognizing Textual Entailment (RTE) and Named Entity Recognition (NER). The results highlight ModBERTBr's effectiveness, demonstrating consistent improvements in STS and NER tasks and competitive performance in RTE compared to the predecessor models.

Keywords: BERT, ModernBERT and Brazilian Portuguese.

ModBERTBr: A ModernBERT-based Model for Brazilian Portuguese

Wallace Ben Teng Lin Wu[1], **Luis Paulo Faina Garcia**[1]

[1]Computer Science Department
University of Brasilia (UnB) – Brasilia – DF – Brazil

wallace.wu@aluno.unb.br, luis.garcia@unb.br

Abstract. *Recent advancements in Artificial Intelligence (AI) have spurred rapid adoption across various sectors and aspects of daily life, significantly propelled by the evolution of Large Language Models (LLMs). A pivotal model in this domain is the Bidirectional Encoder Representations from Transformers (BERT), whose state-of-the-art variant, ModernBERT, excels in efficiency and performance. However, ModernBERT is limited to the English language. Furthermore, a notable gap exists in the availability of models specifically designed for Brazilian Portuguese, with the existing options being outdated regarding contemporary methodologies and technologies. This paper addresses this notable gap by introducing ModBERTBr, a novel pre-trained model based on the ModernBERT architecture, explicitly tailored for Brazilian Portuguese and incorporating cutting-edge research and technologies. The term Mod refers to Modern, while Br stands for Brazilian Portuguese. ModBERTBr was pre-trained from scratch on meticulously curated datasets composed of Wikipedia and BrWac, ensuring high data quality. In addition to incorporating key innovations from ModernBERT, such as an alternating attention mechanism, rotary positional embeddings, and Flash Attention 2, a custom tokenizer employing the unigram model approach was also developed. ModBERTBr was assessed against multiple baselines, including BERTimbau and BERTugues, through intrinsic evaluations such as Masked Language Modeling (MLM) and downstream tasks such as Semantic Textual Similarity (STS), Recognizing Textual Entailment (RTE) and Named Entity Recognition (NER). The results highlight ModBERTBr's effectiveness, demonstrating consistent improvements in STS and NER tasks and competitive performance in RTE compared to the predecessor models.*

Keywords. *BERT · ModernBERT · Brazilian Portuguese.*

1. Introduction

Advancements in Artificial Intelligence have accelerated rapidly, leading to significant investments from both industry and academia. Several factors contribute to this surge, one of the most significant being the widespread adoption of LLMs, such as ChatGPT, DeepSeek, Claude, and LLaMA, by regular users worldwide. These remarkable generative models are constructed on decoder-only architecture and are characterized by their exceptional capabilities to tackle a broad spectrum of tasks [Gao et al. 2025].

Equally important are the encoder-only models, which tend to be less resource-intensive, faster, and more portable. An eminent example is BERT [Devlin et al. 2018],

the foundational architecture for numerous multimodal AI systems and platforms. BERT plays a critical role in addressing several functions integral to the widely used tools available. It is highly esteemed for its versatility and exceptional performance across various tasks, including information retrieval, regression, classification, and entity recognition [Koroteev 2021]. The base model was released in 2018, and even after several years, it remains one of the most widely used and researched models. This perdurance is particularly noteworthy given the substantial advancements in the current state-of-the-art.

ModernBERT [Warner et al. 2024] emerged as a direct replacement for BERT, revealing substantial Pareto improvements over its predecessor while ensuring full backward compatibility. ModernBERT incorporates several state-of-the-art techniques that enhance the training model's efficiency and expand the model's architecture capabilities, yielding an overall performance improvement. Although ModernBERT introduces significant advances, it is trained exclusively in English, which limits its effectiveness in other languages. This language hegemony is prevalent in the domain of LLMs due to the scarcity of resources in different languages despite the substantial demand for them.

In particular, the context of Brazilian Portuguese reveals a notable deficiency: while models like BERTimbau [Souza et al. 2020] and BERTugues [Zago and Pedotti 2024] have made admirable contributions to the field, these few existing models have yet to incorporate the latest architectural advancements and innovations from the industry and academia. Considering the limited resources for training a new model and the scarcity of existing models in this language, one might ponder why not use multilingual models. However, research indicates that multilingual models, which are generalist by nature and trained across multiple languages, often perform worse than their monolingual counterparts in language-specific contexts, as shown by [Souza et al. 2020].

This paper proposes a solution to the identified challenges and compiles the necessary resources and tools for developing a key model. The outcome is the novel BERT-type model, ModBERTBr, a modern version trained from scratch and optimized explicitly for Brazilian Portuguese. Its primary architectural reference was ModernBERT, incorporating many innovative features proposed therein, including the deep and narrow architecture with the transformer++ block and a revised training strategy that used Flash Attention 2. Due to the grammatical differences in the Brazilian Portuguese linguistic scope, a custom tokenizer has been developed and specifically trained for ModBERTBr to enhance its performance further.

Advancements in architectural design, innovative training methodologies, and the integration of extensive high-quality data in Portuguese have enabled ModBERTBr to demonstrate outstanding performance compared to several baseline models. For benchmarking purposes, two monolingual BERT models specifically trained on Portuguese, BERTimbau and BERTugues, were evaluated alongside other reference models such as ModernBERT and mBERT [Devlin et al. 2018], the multilingual variant of base BERT model. These comparisons encompass a range of tasks, including MLM, which serves as the primary objective for pre-training the model, and diverse downstream tasks, specifically STS, RTE, and NER.

2. Background and Related Works

The seminal paper *Attention Is All You Need* [Vaswani et al. 2017] significantly revolutionized the landscape of AI by introducing the transformer architecture. The primary contribution of this research lies in popularizing the attention mechanism, which effectively computes and retains the influence of each token on its neighboring ones. The primary advantage of the attention mechanism is its ability to facilitate parallelism for model training and inference, a pivotal betterment over past sequential models. This mathematical framework revolutionized the field, providing a robust and effective method for representing and understanding languages. The original paper introduced two categories of components: encoder and decoder blocks, each more suitable for distinct tasks. These blocks have since become the foundational components of most state-of-the-art LLMs.

BERT [Devlin et al. 2018] employs a series of encoder blocks to enhance contextual understanding by capturing bidirectional context, in contrast to decoder-only models that rely solely on left context. To accomplish this, the model introduces two pre-training tasks: Masked Language Modeling (MLM) and Next-Sentence Prediction (NSP). In each layer, multi-attention heads and feed-forward networks compute the relation among tokens for refined language understanding, enabling the retention of learned information within the model's parameters. Each token is represented as a 768-dimensional embedding vector, and the model processes input sentences with a maximum context size of 512 tokens, utilizing positional embeddings to account for token position. This architecture accommodates various output heads, providing versatility for multiple tasks, including classification, regression, and question-answering. The original paper introduced BERT with various sub-models, including base, large, and multilingual variants (mBERT).

However, as the pioneer of BERT-type architectures, its design and training methodologies were suboptimal, leaving substantial potential for future enhancements. A Robustly Optimized BERT Pretraining Approach (RoBERTa) [Liu et al. 2019] presented an extensive analysis of the original BERT model. A noteworthy recommendation involved removing the NSP task regarded as an overhead that does not improve results [Izsak et al. 2021]. Another observation was the use of dynamic masking, which involves masking different tokens as the training epochs progress, a feature inaccurately overlooked by the original paper.

Subsequently, some research concentrated on the encoder landscape for the Portuguese language. BERTimbau [Souza et al. 2020] emerged as the pioneering model explicitly designed for this language based on BERT. This study demonstrated that monolingual models outperform multilingual ones, such as mBERT, in language-specific tasks, achieving state-of-the-art results in Semantic Textual Similarity (STS) and Recognizing Textual Entailment (RTE) for the ASSIN2 dataset. Later, another BERT model tailored for Portuguese, BERTugues [Zago and Pedotti 2024], emerged. This one builds on BERTimbau by retraining the reference model with targeted optimizations, resulting in performance improvements across specific tasks. Key enhancements include optimizing the tokenizer and refining the cleaning filter criteria for training data. However, both models adopted the original BERT architecture, which is becoming increasingly outdated, particularly given the numerous advancements in the field over recent years.

Following groundbreaking innovations, ModernBERT [Warner et al. 2024] synthesizes numerous contributions within the realm of LLMs and consolidates significant

advancements into a unified framework. This development introduced substantial Pareto improvements, enhancing computational speed and overall performance. In the architectural field, ModernBERT utilizes a reformulated version of the original transformer block, referred to as transformer++, introduced by Mamba [Gu and Dao 2024]. This adoption implies that the original positional embedding is replaced by rotary positional embeddings (ROPE), as presented by ROFOMER [Su et al. 2024], which improves the capture of relative distance between tokens while maintaining absolute position information. Furthermore, it also incorporates a normalization function between layers that stabilizes training [Xiong et al. 2020].

At an equivalent parameter count, models characterized by deep and narrow architectures exhibit distinct learning patterns compared to those with shallow and wide configurations [Nguyen et al. 2020]. Recent studies suggest that deeper and narrower language models consistently demonstrate enhanced performance on downstream tasks compared to their shallower and wider counterparts [Tay et al. 2021]. Therefore, compared to the original BERT model, ModernBERT was designed to incorporate 22 layers instead of 12 while maintaining the embedding size of 768. To balance the number of parameters, ModernBERT reduced the intermediate size of the linear layers from 3072 to 1152 and eliminated the bias term, considered an inefficient use of parameters [Xuan et al. 2020].

As for the training methodology, ModernBERT also introduced several enhancements. Improvements include utilizing Flash Attention 2 [Dao 2023], a framework optimized for computing the attention mechanism; adopting the Warmup-Stable-Decay (WSD) learning rate scheduler, a more robust approach for long pre-training sessions [Hägele et al. 2024]; and employing an alternating attention mechanism, in which global attention is applied to all 512 tokens every three layers. In comparison, local attention attends to the nearest 128 tokens for other layers, improving efficiency while proficiently addressing both short-range and long-range contextual dependencies [Warner et al. 2024].

For pre-training, it focused exclusively on the MLM task with a dynamic masking rate of 30%, as the original 15% rate was deemed suboptimal [Wettig et al. 2023]. In this manner, ModernBERT achieves state-of-the-art results across multiple tasks in the English linguistic domain while maintaining a moderate number of parameters, comparable with the past reference models. Finally, another noteworthy advantage of ModernBERT is that it is the most memory-efficient model overall when compared to recent rival models, resulting in faster training and inference times while diminishing hardware consumption and optimizing cost efficiency [Warner et al. 2024].

3. ModBERTBr

The novel ModBERTBr model was developed after a detailed analysis of the literature. This model, based mainly on ModernBERT, has been refined throughout the entire pipeline, redesigning major components, including the model architecture, training strategy, and tokenizer. In addition to innovations suggested by ModernBERT, it integrates enhancements introduced by other works while remaining completely retro-compatible with older BERT models. Moreover, ModBERTBr extends the advancements made by BERTimbau and BERTugues within the Brazilian Portuguese context, utilizing the same data collection and further enhancing the dataset filtering stage. Consequently, an innovative model has been successfully developed, inheriting the significant advantages of the

referenced models. ModBERTBr exhibits remarkable performance in various tasks, including STS, RTE, and Named Entity Recognition (NER), with faster training time and more efficient hardware usage.

3.1. Dataset

The data employed in this study focuses on Brazilian Portuguese and was obtained from Wikipedia [1], which retains valuable information across various topics and comprises 1.11 million pages within the Portuguese subset. Given the scale of ModBERTBr, the BrWAC dataset [2] [Wagner et al. 2018] was additionally incorporated. It consists of multiple pages from the Internet and contains 3.53 million documents. The Wikipedia dataset was already clean, whereas the BrWAC dataset required a thorough pre-processing procedure that included rectifying Mojibake issues and removing HTML tags as conducted by [Souza et al. 2020, Zago and Pedotti 2024].

After concatenating the datasets, the texts were split into individual paragraphs. This segmentation process also involved filtering out sentences with a single word and empty documents, ultimately yielding 157 million usable documents. As suggested by [Rae et al. 2021], other criteria were then incorporated in the data pre-processing stage. First, restrictions on document length were imposed, requiring it to be within a specified word count range. This criterion ensures that texts do not exceed a size that could lead to truncation while also preventing excessively short texts, which would result in inefficient model usage. Second, the condition of having a minimum number of stop words assures that the documents have semantic and syntactic coherence. Finally, filtering texts based on the mean word length ensures that words are not incorrectly spaced (e.g., *s p l i t*) with low average values or improperly combined (e.g., *wordsconcatenated*) when the average values are overly high.

The last step was to divide the documents into three groups for the training process: Datasets 1, 2, and 3. Dataset 1 was constructed using a broader range of word counts and more lenient criteria for the other metrics. Meanwhile, Dataset 2 was created from shorter documents that adhered to more stringent criteria regarding stop words and average character count. Dataset 3 is the combination of the previously mentioned datasets. Finally, all datasets were partitioned into training and testing subsets to mitigate bias in the intrinsic evaluation process, applying a 90% / 10% split ratio with random shuffling. Table 1 shows the details regarding the complete filtering criteria and final dataset splits.

Table 1. Filtering criteria and datasets sizes. $[L, R]$ refers to the closed interval from L to R.

Filtering criteria & datasets	Dataset 1	Dataset 2	Dataset 3
Word count	[20, 512]	[10, 19]	[10, 512]
Minimum stopword count	1	2	mixed
Mean word length	[2, 15]	[3, 10]	mixed
Train split (90%)	56.7M	42.2M	98.9M
Test split (10%)	6.3M	4.7M	11.0M
Total dataset size	63.0M	46.8M	109.9M

[1] https://huggingface.co/datasets/wikimedia/wikipedia
[2] https://huggingface.co/datasets/UFRGS/brwac

3.2. Tokenizer

According to [Ali et al. 2024], the unigram algorithm generally outperforms other models when applied to Romance languages, such as Portuguese. This effectiveness stems from the unigram model's ability to capture the lexical structure inherent in these languages effectively. With this tokenizer, root words and their derivations tend to share the same tokens, simplifying ModBERTBr's process of grammar mapping for Brazilian Portuguese. The Unigram algorithm, introduced by [Kudo 2018], assumes that each subword occurs independently. Hence, the probability of a subword sequence is expressed as the product of the individual probabilities of each subword's occurrence. Accordingly, ModBERTBr utilizes a custom fast tokenizer based on the unigram algorithm, which is trained on the training split of Dataset 1 to acquire the Brazilian Portuguese vocabulary. The vocabulary size for this tokenizer was set to 32 768 tokens, and the same set of special tokens and context size of 512 from the original BERT was kept to ensure retro compatibility.

A tokenizer constitutes a pipeline comprised of several distinct components. First, this custom tokenizer employs NFC Unicode normalization to ensure uniformity and converts all characters to lowercase. Then, the pre-tokenizer separates words after identifying whitespaces, punctuation, and digits. Following this, it incorporates a metaspace (the character $U + 2581$ before all words to represent that the associated token marks the beginning of a word, analogous to ALBERT's tokenizer [Lan et al. 2019]. Sequentially, the post-tokenizer consistently appends a [CLS] token at the beginning and a [SEP] token at the end. It also appends [SEP] tokens between sentences when multiple sentences are provided as input. Finally, the decoder component retrieves the previously employed metaspaces and recombines the tokens originally belonging to the same word.

3.3. Model Architecture

ModBERTBr's significant architectural modifications include implementing transformer++ blocks, a deep and narrow configuration of 22 layers with a hidden size of 1152, normalization layers, an alternating attention mechanism, and the removal of the bias term. Table 2 presents the comprehensive list of parameters for the ModBERTBr architecture.

Table 2. Parameters of ModBERTBr architecture.

ModBERTBr	
Parameters	136 120 832
Activation function	GELU
Embedding size	768
Intermediate size	1 152
Attention heads	12
Number of hidden layers	22
Local attention size	128
Context size	512
Vocabulary size	32 768

3.4. Training Strategy

In addition to the development of more advanced models, there has been a concurrent evolution of sophisticated libraries designed for training these models over the years. In

this context, ModBERTBr utilizes Flash Attention 2 [Dao 2023]. This framework significantly enhances training efficiency on hardware by leveraging the asymmetric GPU memory hierarchy, thereby breaking down the computationally intensive process of computing attention into simpler and faster sub-processes.

The complete pre-training process was systematically divided into two distinct phases: the first phase (i) focused on achieving a general understanding of the language with Dataset 1, and the second phase (ii) involved a more refined approach that utilized the expanded Dataset 3 alongside more advanced training augmentations. Some hyperparameters shared by both phases include using AdamW optimizer with Betas (0.9, 0.999) and 1×10^{-6} Epsilon, the batch size of 32, and the f16 mixed precision. After each phase, the last checkpoint was loaded.

For (i), it utilized a linear decay schedule, reducing the learning rate (LR) from 5×10^{-5} to 0 with a constant slope. As for (ii), it included weight decay to regularize model parameters, and the base LR was lowered to 1×10^{-5}, considering that the model had already established a robust foundation in language comprehension during the first phase [Shen et al. 2024]. Furthermore, the WSD LR scheduler was introduced for (ii), consisting of three distinct stages: Warmup, which is particularly advantageous for minimizing instabilities while loading previously trained model parameters; Stable, which promotes steady and continuous training with a high LR for most of the procedure; and Decay, which facilitates further fine-tuning and enhances the model's convergence at the end of training [Hägele et al. 2024]. The total pre-training procedure lasted over 200 hours on an AMD-powered cluster with four MI250 accelerators (eight GPUs). Table 3 presents the complete list of hyperparameters for each phase.

Table 3. Hyperparameters for the pre-training of ModBERTBr

Hyperparameter	**Phase (i)**	**Phase (ii)**
Dataset size	56.7M	98.9M
Weight decay	0	1e-4
Learning rate	5e-5	1e-5
LR scheduler	Linear Decay	WSD
Warmup steps	0	40k
Stable steps	0	240k
Decay steps	500k	120k
LR decay function	Linear	Cosine
Total steps	500k	400k
Epochs	2.2	1.0

4. Methodology

The evaluation of ModBERTBr was conducted in two ways: intrinsically and extrinsically. For the intrinsic assessment, the tokenizer was evaluated using the fertility metric, which quantifies the efficiency of token generation and directly affects the model's downstream performance [Ali et al. 2024]. Additionally, ModBERTBr underwent pre-training on the MLM task, which has proven effective and sufficient for comprehending the underlying linguistic structures [Devlin et al. 2018]. Thereafter, considering the test split

from Dataset 3, ModBERTBr was evaluated in the MLM task using the cross-entropy loss metric, which represents the model's confidence when predicting the masked tokens.

On the other hand, extrinsic evaluation involves fine-tuning the base model for downstream tasks to assess its performance in diverse applications. The tasks considered were Semantic Textual Similarity (STS), Recognizing Textual Entailment (RTE), and Named Entity Recognition (NER), which enables the assessment of performance at token and sentence levels. First, STS is a regression problem characterized by two input sentences and a corresponding numerical output. It aims to quantify the degree of similarity between the two given sentences. Then, RTE is a binary classification task involving two ordered sentences and a binary output. The objective is to determine whether one sentence, the premise, entails or logically implies another sentence, the hypothesis. Finally, NER is a multi-label classification task with the objective of categorizing all tokens in the input sentence into named entity labels.

The evaluation metrics utilized for STS were the Mean Squared Error (MSE) and the Pearson correlation coefficient. As for the RTE task, the employed metrics included accuracy and the F1 (Macro) score. Finally, metrics for NER comprised the F1 score, Recall, and Precision for each class, as well as aggregate metrics for all entities. The ASSIN2 [3] corpus, constituted of sentence pairs in Brazilian Portuguese manually annotated for entailment and semantic similarity, was used for STS and RTE tasks. For NER, The LeNER-Br [4] [Luz de Araujo et al. 2018] dataset was utilized. It is designed in Portuguese and consists entirely of manually annotated legal documents obtained from various Brazilian Courts.

As the architectural reference model, ModernBERT is included as a baseline model, allowing for a comparison of its limited English specialty with the effects of evaluating in another language. Moreover, as the evaluation focuses on tasks related to Portuguese, the comparative analysis also utilized BERTimbau, BERTugues, and mBERT. For the aforementioned downstream tasks, each baseline model and ModBERTBr underwent a fine-tuning process, which further trained the models on their respective datasets and tasks. During the training stage, distinguishing environments and hyperparameter settings were employed for the BERT models (mBERT, BERTimbau, and BERTugues) and the ModernBERT models (ModernBERT and ModBERTBr) to optimize results for each task. However, some similarities were shared, such as using the AdamW optimizer with Betas (0.9, 0.999), Epsilon 1×10^{-6}, and avoiding f16 mixed precision. Table 4 outlines other hyperparameters used in fine-tuning.

The experiments were conducted on two different environments: an AMD-powered cluster with four MI250 accelerators (eight GPUs) running Rocky Linux 9.4, Python 3.9.14, Pytorch 2.5.1+rocm6.2 with Flash Attention 2; and a Nvidia-powered node, with two Tesla V100, running Ubuntu 24.04 LTS, Python 3.12.8, and Pytorch 2.5.1. Both platforms used Transformers 4.48.1 for pre-training and fine-tuning. The AMD cluster was used for pre-training ModBERTBr and fine-tuning ModernBERT architecture models. Meanwhile, BERT architecture models used the NVIDIA node for the fine-tuning process.

[3] https://huggingface.co/datasets/nilc-nlp/assin2

[4] https://huggingface.co/datasets/peluz/lener_br

Table 4. Hyperparameters used for fine-tuning the models

Hyperparameters	Bert Models			ModernBERT Models		
	STS	RTE	NER	STS	RTE	NER
Batch size	16	16	16	4	4	4
Learning rate	4e-5	2e-5	4e-5	2e-5	1e-5	2e-5
Weight decay	1e-2	1e-2	1e-2	1e-4	1e-4	1e-4
LR scheduler	WSD	Linear	WSD	WSD	Linear	WSD
Warmup steps	300	0	300	500	0	300
Stable steps	600	0	900	1 000	0	900
Decay steps	2100	3 000	1 800	3 500	10 000	1 800
LR decay function	Cosine	Linear	Cosine	Cosine	Linear	Cosine
Total steps	3 000	3 000	3 000	5 000	10 000	3 000
Epochs	15	15	13	25	50	13

The complete project, including code, data, and training configurations, is available in a Git repository [5]. For convenience, ModBERTBr and its weights were also made publicly available at the Hugging Face Hub [6].

5. Experiments and Results

This section presents the results of ModBERTBr, encompassing intrinsic assessments for both the tokenizer and the model and evaluations for downstream tasks compared to the baseline models. Subsequent analysis of these results reveals several intriguing observations.

5.1. Intrinsic Evaluation

Fertility measures the number of tokens a tokenizer generates for a given sequence of words, and studies have shown that lower fertility correlates with downstream model performance success [Goldman et al. 2024]. The value obtained by ModBERTBr is 1.5357, signifying an excellent compression rate while effectively generating significant tokens to represent Brazilian words and grammatical structures.

For the pre-training evaluation, logs were systematically saved and analyzed throughout the training process for both phases, as shown in Figures 1–6. The LR curves reflect the schedulers' impact, illustrating their influence on the gradient normalization and loss curves. Upon analyzing the gradient curve, the values indicate that it did not increase exponentially, suggesting that the updates remained stable and facilitated continued learning by the model. Additionally, the absence of gradient vanishing implies that the weights continued to be updated, preventing excessively slow or halted learning processes. Consequently, it is observable that the loss curves continued to decrease steadily. Finally, the results demonstrate that the model did not reach saturation and maintains the potential for further enhancement while already producing impressive outcomes.

For the MLM task, the final values of cross-entropy loss of ModBERTBr in the training and testing stages are 1.4632 and 1.4971, respectively. These results suggest

[5] https://github.com/wallacelw/ModBERTBr

[6] https://huggingface.co/wallacelw/ModBERTBr

that, on average, the model has approximately 25% and 22% confidence when predicting tokens. This outcome is commendable considering the extensive number of tokens involved and shows that the model is effectively learning the language and developing an understanding of it.

5.2. Extrinsic Evaluation

The results generated by each model for each task are presented in Table 5. Except for BERTimbau values in STS and RTE, which are documented in [Souza et al. 2020], all results were obtained from the fine-tuning experiments conducted by this paper.

Table 5. Evaluation of models for STS, RTE and NER tasks. (Best values in bold).

Model	STS (ASSIN 2)		RTE (ASSIN 2)		NER (LeNER-Br)		
	MSE	Pearson	F1(Macro)	Accuracy	F1(Micro)	Recall	Precision
mBERT	0.597	0.801	84.45%	84.52%	88.50%	90.45%	86.63%
BERTimbau	0.580	**0.836**	**89.2%**	**89.2%**	**90.48%**	91.64%	**89.35%**
BERTugues	0.583	0.823	86.27%	86.40%	89.56%	90.59%	88.55%
ModernBERT	0.514	0.790	81.09%	81.17%	75.76%	78.03%	73.61%
ModBERTBr	**0.509**	0.812	85.28%	85.42%	90.08%	**92.40%**	87.88%

For the STS task, ModBERTBr achieves the lowest value for the MSE metric, indicating the best numerical accuracy in terms of the magnitude of the predictions. However, this model is less effective in capturing the underlying trends when compared to BERTimbau, which exhibits the highest value for the Pearson Correlation Coefficient. Nevertheless, ModBERTBr does not significantly lag in correlation and has respectable linearity between predictions and targets. An interesting observation is that ModernBERT performs surprisingly well in the STS task despite not being trained in Portuguese data. This result suggests that the advanced architecture alone has contributed substantially to the successful fine-tuning process for STS, a regression task.

In contrast, an analysis of the RTE and NER tasks reveals that the model demonstrating the lowest performance is also ModernBERT, while mBERT shows subpar performance overall. This observation highlights the crucial role of pre-training in the appropriate language for specific tasks, notably classification tasks such as RTE and NER. For the RTE task, BERTimbau outperforms other models across both evaluation metrics, with BERTugues securing second place and ModBERTBr in third. Lastly, ModBERTBr performs best in the NER task when evaluated using the recall metric, indicating that it is more effective at identifying the real named entities. Further, it remains competitive in other metrics, placing close second for the F1 score (Micro) and third for precision.

6. Conclusion and Future Work

This research identifies BERT's shortcomings when applied to Brazilian Portuguese. ModBERTBr has been proposed to address these deficiencies, incorporating cutting-edge training techniques alongside architectural enhancements. The development of ModBERTBr was a meticulous process at every stage of the pipeline, including thorough data cleaning, the assembly of an explicitly designed custom tokenizer, a modern training paradigm, and fine-tuning multiple models for evaluation and comparison.

The results from both intrinsic and extrinsic evaluations demonstrate that the proposed model performs competitively compared to other specialized Portuguese models,

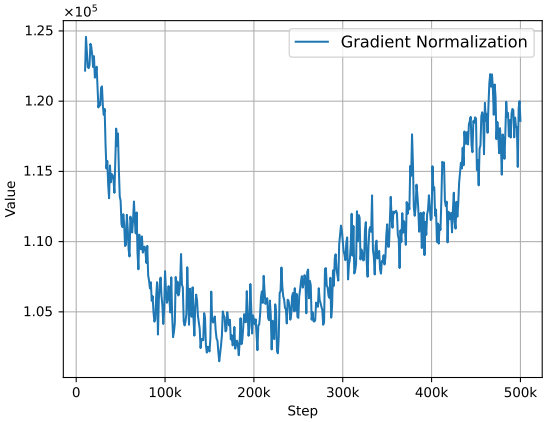


Figure 1. GradNorm for (i) with a 0.6 smoothing factor.

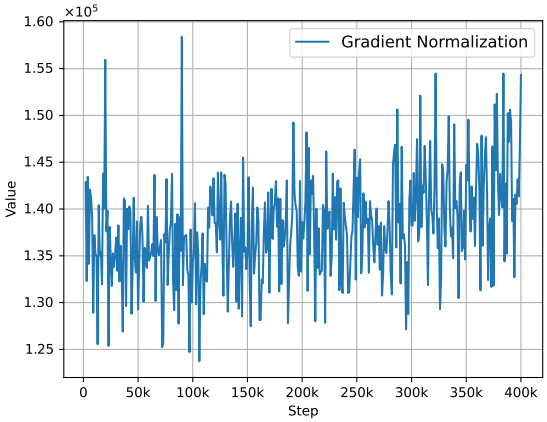


Figure 2. GradNorm for (ii) with a 0.6 smoothing factor.

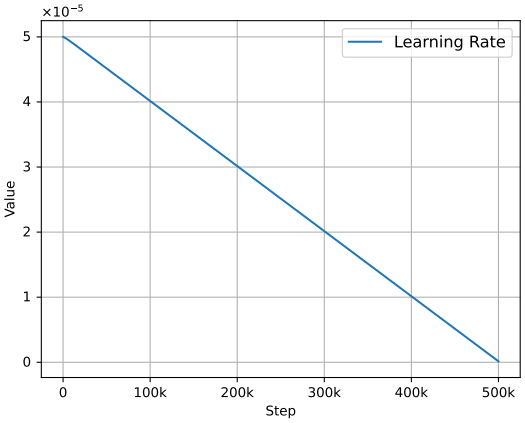


Figure 3. LR curve for (i).

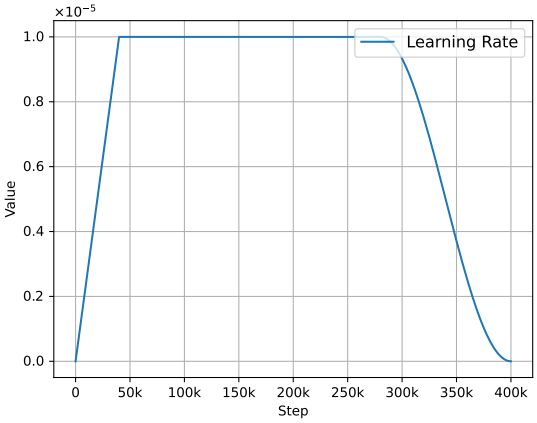


Figure 4. LR curve for (ii).

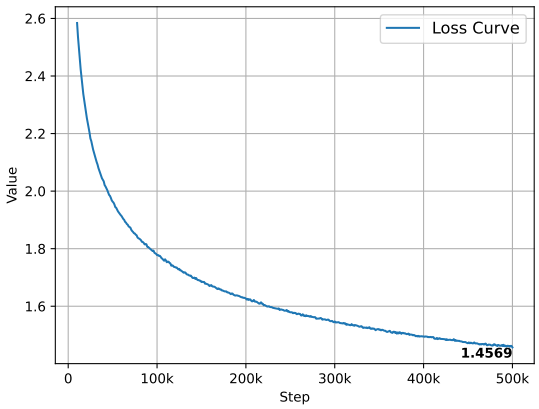


Figure 5. Training loss for (i).

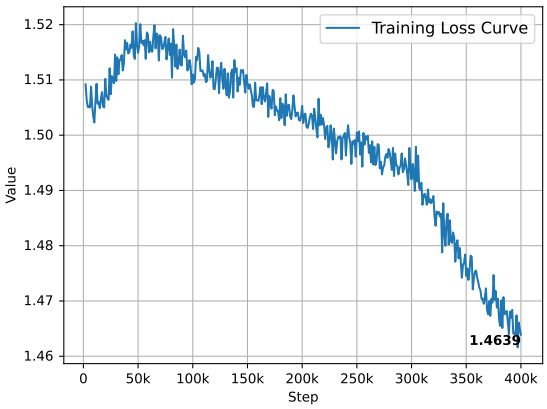


Figure 6. Training loss for (ii).

such as BERTimbau and BERTugues, surpassing these models in specific metrics and tasks. Additionally, when comparing it to ModernBERT, which incorporates the latest advancements in BERT architecture, it becomes clear that pre-training a model within the specific language domain of the tasks significantly enhances its performance. Furthermore, comparisons with mBERT confirm that specialized models outperform generalist models in monolingual tasks.

The analysis of the plotted curves indicates that ModBERTBr has not yet reached saturation, meaning that additional training iterations can still considerably enhance the model's performance and robustness. Furthermore, incorporating more recent and diverse data sources during the training stage is also highly advantageous for improving the model's knowledge. Finally, the literature has presented additional techniques, such as Unpadding with Sequence Packing and expanding the context size further than 512. These advancements warrant consideration for future research endeavors.

Acknowledgments

Advanced Micro Devices, Inc. partially funded this research through the AMD AI & HPC Cluster Program. Furthermore, the UFRGS team and Luz, P. et al. (2018) are appreciated for, respectively, providing the BrWAC [Wagner et al. 2018] and for the LeNER-BR [Luz de Araujo et al. 2018] datasets. Finally, the usage of AI tools for text reviewing and code generation is also mentioned.

Disclosure of Interests

The authors declare that they have no competing interests relevant to this article's content.

References

Ali, M., Fromm, M., Thellmann, K., Rutmann, R., Lübbering, M., Leveling, J., Klug, K., Ebert, J., Doll, N., Buschhoff, J. S., Jain, C., Weber, A. A., Jurkschat, L., Abdelwahab, H., John, C., Suarez, P. O., Ostendorff, M., Weinbach, S., Sifa, R., Kesselheim, S., and Flores-Herr, N. (2024). Tokenizer choice for llm training: Negligible or crucial? arXiv preprint arXiv:2310.08754.

Dao, T. (2023). Flashattention-2: Faster attention with better parallelism and work partitioning. arXiv preprint arXiv:2307.08691.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Gao, T., Jin, J., Ke, Z. T., and Moryoussef, G. (2025). A comparison of deepseek and other llms. arXiv preprint arXiv:2502.03688.

Goldman, O., Caciularu, A., Eyal, M., Cao, K., Szpektor, I., and Tsarfaty, R. (2024). Unpacking tokenization: Evaluating text compression and its correlation with model performance. arXiv preprint arXiv:2403.06265.

Gu, A. and Dao, T. (2024). Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752.

Hägele, A., Bakouch, E., Kosson, A., Allal, L. B., Werra, L. V., and Jaggi, M. (2024). Scaling laws and compute-optimal training beyond fixed training durations. arXiv preprint arXiv:2405.18392.

Izsak, P., Berchansky, M., and Levy, O. (2021). How to train BERT with an academic budget. In Moens, M.-F., Huang, X., Specia, L., and Yih, S. W.-t., editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10644–10652, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Koroteev, M. V. (2021). BERT: A review of applications in natural language processing and understanding. *CoRR*, abs/2103.11943.

Kudo, T. (2018). Subword regularization: Improving neural network translation models with multiple subword candidates. *CoRR*, abs/1804.10959.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Luz de Araujo, P. H., de Campos, T. E., de Oliveira, R. R. R., Stauffer, M., Couto, S., and Bermejo, P. (2018). LeNER-Br: a dataset for named entity recognition in Brazilian legal text. In *International Conference on the Computational Processing of Portuguese (PROPOR)*, Lecture Notes on Computer Science (LNCS), pages 313–323, Canela, RS, Brazil. Springer.

Nguyen, T., Raghu, M., and Kornblith, S. (2020). Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. In *International Conference on Learning Representations*.

Rae, J. W., Borgeaud, S., Cai, T., Millican, K., Hoffmann, J., Song, H. F., Aslanides, J., Henderson, S., Ring, R., Young, S., Rutherford, E., Hennigan, T., Menick, J., Cassirer, A., Powell, R., van den Driessche, G., Hendricks, L. A., Rauh, M., Huang, P., Glaese, A., Welbl, J., Dathathri, S., Huang, S., Uesato, J., Mellor, J., Higgins, I., Creswell, A., McAleese, N., Wu, A., Elsen, E., Jayakumar, S. M., Buchatskaya, E., Budden, D., Sutherland, E., Simonyan, K., Paganini, M., Sifre, L., Martens, L., Li, X. L., Kuncoro, A., Nematzadeh, A., Gribovskaya, E., Donato, D., Lazaridou, A., Mensch, A., Lespiau, J., Tsimpoukelli, M., Grigorev, N., Fritz, D., Sottiaux, T., Pajarskas, M., Pohlen, T., Gong, Z., Toyama, D., de Masson d'Autume, C., Li, Y., Terzi, T., Mikulik, V., Babuschkin, I., Clark, A., de Las Casas, D., Guy, A., Jones, C., Bradbury, J., Johnson, M. J., Hechtman, B. A., Weidinger, L., Gabriel, I., Isaac, W., Lockhart, E., Osindero, S., Rimell, L., Dyer, C., Vinyals, O., Ayoub, K., Stanway, J., Bennett, L., Hassabis, D., Kavukcuoglu, K., and Irving, G. (2021). Scaling language models: Methods, analysis & insights from training gopher. *CoRR*, abs/2112.11446.

Shen, Y., Stallone, M., Mishra, M., Zhang, G., Tan, S., Prasad, A., Soria, A. M., Cox, D. D., and Panda, R. (2024). Power scheduler: A batch size and token number agnostic learning rate scheduler. arXiv preprint arXiv:2408.13359.

Souza, F., Nogueira, R., and Lotufo, R. (2020). Bertimbau: Pretrained bert models for brazilian portuguese. In Cerri, R. and Prati, R. C., editors, *Intelligent Systems*, pages 403–417. Springer International Publishing, Cham.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. (2024). Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Tay, Y., Dehghani, M., Rao, J., Fedus, W., Abnar, S., Chung, H. W., Narang, S., Yogatama, D., Vaswani, A., and Metzler, D. (2021). Scale efficiently: Insights from pre-training and fine-tuning transformers. *CoRR*, abs/2109.10686.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

Wagner, J., Wilkens, R., Idiart, M., and Villavicencio, A. (2018). The brwac corpus: A new open resource for brazilian portuguese. In *Conference: 11th Language Resources and Evaluation Conference (LREC 2018)*.

Warner, B., Chaffin, A., Clavié, B., Weller, O., Hallström, O., Taghadouini, S., Gallagher, A., Biswas, R., Ladhak, F., Aarsen, T., Cooper, N., Adams, G., Howard, J., and Poli, I. (2024). Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. arXiv preprint arXiv:2412.13663.

Wettig, A., Gao, T., Zhong, Z., and Chen, D. (2023). Should you mask 15 arXiv preprint arXiv:2202.08005.

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., and Liu, T. (2020). On layer normalization in the transformer architecture. *CoRR*, abs/2002.04745.

Xuan, H., Stylianou, A., Liu, X., and Pless, R. (2020). Hard negative examples are hard, but useful. *CoRR*, abs/2007.12749.

Zago, R. and Pedotti, L. (2024). Bertugues: A novel bert transformer model pre-trained for brazilian portuguese. *Semina: Ciências Exatas e Tecnológicas*, 45:e50630.