Insert the title of the dissertation, project or internship report, font Arial Bold, font size adjusted to the text box 12x12cm, left aligned

Author's name, Arial Plain, 18
Course's Name, Arial Plain, 12
Department, Arial Plain, 10
Faculty of Sciences of University of Porto and [name of the Faculty/Institution], Arial Plain, 10
Year

Insert a figure related to the theme

(optional)

Insert the title of the dissertation, project or internship report, font Arial Bold, font size adjusted to the text box 12x12cm, left aligned

Author's name, Arial Plain, 18

Dissertation/Internship/Project Report carried out as part of the [course's name], Arial Plain, 12

Department, Arial Plain, 10

Year

Supervisor
Supervisor's Name, Category, Institution

Co-supervisor [if applicable]
Supervisor's Name, Category, Institution

External Host Supervisor [if applicable]
Name, Professional status, Company's name

Logo

(company/research unit)

[if applicable]

Logo

(company/research unit)

[if applicable]

Acknowledgements

Acknowledge ALL the people!

Resumo

Este tese é sobre alguma coisa

Palavras-chave: física (keywords em português)

Abstract

This thesis is about something, I guess.

Keywords: Computer Sciences

Table of Contents

List of Figures

1. Introduction

In recent years, Artificial Intelligence models have dominated the market with the introduction of Large Language Models (LLMs) like ChatGPT [**?**] and GPT-4 [**?**].
These models are trained on extensive datasets containing hundreds of billions to trillions of words [**? ?**], requiring enormous computational resources and vast amounts of text data.
One of the main challenges these models face is the lack of training data for low-resource languages.
Without the massive data required to create these models, languages with smaller digital footprints currently lack models with the same performance as those for widely spoken languages such as English.
The main goal of this dissertation is to explore how to adapt existing models trained primarily on one language to another using minimal computational resources.
For that, we focus on the *tokenizer*, the fundamental building block of state-of-the-art LLM models.

1.1. Motivation

The development of language models for low-resource languages presents significant challenges due to limited available data and computational constraints. Traditional approaches to creating language-specific models typically involve either training from scratch—requiring enormous datasets and computational resources—or extensive fine-tuning of existing models, which still demands considerable resources.

For European Portuguese, despite being spoken by approximately 10 million native speakers, the availability of high-quality language models lags behind those for more widely spoken languages. This disparity creates barriers to technological inclusion and limits access to advanced language technologies for Portuguese speakers.

This research is motivated by the need to develop efficient methods for adapting existing language models to low-resource languages without requiring extensive retraining. By focusing on tokenizer adaptation rather than complete model retraining, we aim to provide a computationally efficient approach that can be applied to various languages with limited digital resources.

The potential impact of this research extends beyond European Portuguese, offering a methodology that could be applied to numerous other languages facing similar resource constraints. By reducing the computational and data requirements for language adaptation, this approach could democratize access to advanced language technologies across a broader linguistic spectrum.

1.2. Objectives

By focusing on trained models for specific languages, we aim to adapt them to another ”target” language using the least amount of resources possible.
With that in mind, our goals mainly focused on manipulating the tokenizer and the model's embedding layer, without requiring intensive additional training.

We focused on addressing the following key research questions:

- Is it possible for an English-trained model to achieve comparable performance in European Portuguese by strategically modifying the tokenizer?
- Can tokenizer adaptation accelerate the training process for new language adaptation?
- How much can inference efficiency be improved by adding language-specific tokens to the model's vocabulary?
- What embedding initialization strategies are most effective for integrating new tokens into a pre-trained model?

These key questions guided our research methodology and led to several novel insights regarding tokenizer adaptation for cross-lingual model performance.

1.3. Approach

Our approach to adapting language models to European Portuguese centers on modifying the tokenizer component while minimizing changes to the model's parameters. The methodology consists of four primary components:

1. **Token Selection**: We train a new Byte-Pair Encoding (BPE) tokenizer specifically on Portuguese text corpora, then identify high-frequency tokens that are not present in the original model's vocabulary.

2. **Vocabulary Expansion**: We expand the model's vocabulary by adding these Portuguese-specific tokens, effectively creating a hybrid tokenizer that maintains compatibility with the original language while gaining efficiency for Portuguese text.

3. **Embedding Initialization**: We develop and compare novel strategies for initializing the embeddings of the newly added tokens, including Mean Vector Initialization and Position-Weighted Initialization, which account for the positional importance of constituent tokens.

4. **Inference Adaptation**: We implement a specialized inference procedure that enables the model to effectively utilize the newly added tokens during text generation, maintaining compatibility with the model's pre-trained weights.

This approach requires minimal computational resources compared to traditional fine-tuning or pre-training methods, making it particularly suitable for scenarios with limited access to high-performance computing infrastructure.

1.4. Contributions

This dissertation makes several significant contributions to the field of multilingual language model adaptation:

- Development of a novel tokenizer adaptation methodology that enhances model performance for European Portuguese without requiring complete model retraining
- Introduction and comparative analysis of two embedding initialization strategies—Mean Vector Initialization and Position-Weighted Initialization—with the latter demonstrating superior performance across evaluation metrics
- Empirical evidence supporting the importance of token position in embedding representation, as demonstrated by the effectiveness of the position-weighted initialization approach
- Creation of an inference adaptation framework that enables efficient utilization of enhanced tokenizers while maintaining compatibility with pre-trained model weights
- Comprehensive evaluation using European Portuguese-specific benchmarks, establishing a methodological foundation for future work in low-resource language adaptation

These contributions collectively advance the state of the art in efficient cross-lingual adaptation of language models, particularly for languages with limited resources.

1.5. Chapter Summaries

The remainder of this dissertation is organized as follows:

- **Chapter 2: Background** provides the theoretical foundation for understanding transformers architecture, tokenization methods, and their role in language modeling. It covers the fundamental concepts of transformer models, attention mechanisms, and various tokenization algorithms.
- **Chapter 3: Related Work** reviews existing approaches to language model adaptation, including fine-tuning methodologies and tokenizer-focused techniques. It also discusses available datasets and benchmarks for European Portuguese evaluation.
- **Chapter 4: Methodology** details our approach to tokenizer adaptation, including token selection criteria, embedding initialization strategies, and inference adaptation techniques. It provides mathematical formulations for the proposed methods and explains the experimental configuration.
- **Chapter 5: Results** presents the empirical evaluation of our approach using European Portuguese benchmarks. It includes comparative performance analysis, token efficiency metrics, and qualitative assessment of model outputs.
- **Chapter 6: Conclusions** summarizes the key findings, acknowledges limitations of the current approach, and outlines promising directions for future research in this area.

2. Background

In this chapter, the relevant background needed for the work done in this thesis is presented, with an emphasis on the transformers architecture and tokenizers as well as the different models used to create them. The principal goal of this chapter is to provide a background for the reader to understand the work done in this thesis, given that the reader already has some basic understanding of the field at study, machine learning, neural networks and deep learning to name a few.

2.1. Transformers

Introduced in the 2017 paper, **?**], the transformers architecture have revolutionized the field of Natural Language Processing (NLP). Transformers are encoder-decoder architectures that use a self-attention mechanism to learn the dependencies between the words in a sentence.

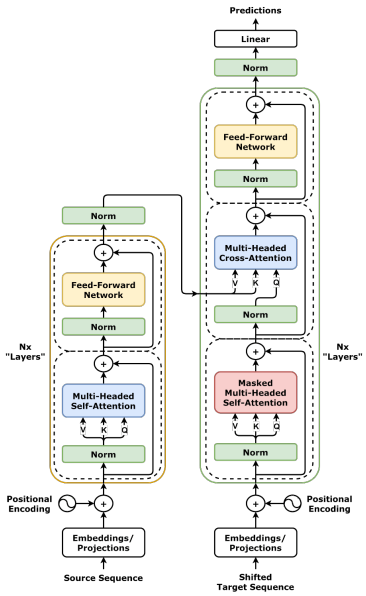

FIGURE 2.1: Transformers Architecture

2.1.1 Encoding

As seen in Figure 2.1, the transformers architecture starts with encoding an input sequence into a sequence of vectors of fixed size. Each sequence is first split into tokens, which have been pre-trained on a large corpus of text to produce the vectors of the encoding layer.

2.1.2 Decoding

After obtaining the encoding of the input sequence, the decoder uses the self attention layer to learn the dependencies between the input tokens which updates the weights of the input tokens. This usually happens multiple times, in the hidden layers, and is later passed through a feed-forward layer.

After obtaining the final output of the Feed-forward layer, the decoder outputs a probability distribution over the vocabulary, made up of the initial tokens given as inputs.

2.1.3 Attention Layer

As mentioned above, during the attention layer the model learns the dependencies between the input tokens. This is done by computing the attention weights for each token in the input sequence, which are then used to update the weights of the input tokens. The attention weights are computed by multiplying the input tokens by a weight matrix, which is then passed through a softmax function to obtain the attention weights.

2.2. Tokenizers

For a sequence of words to be processed by a transformer model, it must first be converted into something a computer can understand. This is done by tokenizing the input sequence into a sequence of tokens. On its most basic form, a tokenizer is a function that takes a sequence of words and returns a sequence of tokens. These tokens can be obtained by different methods, such as splitting the sequence into words, splitting the sequence into characters, or using a pre-trained model to tokenize the sequence.

We will address some of the most common algorithm used to obtain the vocabulary (made up of tokens) of a transformer model.

2.2.1 Token

A token is simply a sequence of characters, it can be a word, a character or anything in between. A sequence of words can be converted into a sequence of tokens by applying the following steps:

1. **Preprocessing:** Normalize the input text (e.g., lowercase conversion, Unicode normalization, stripping whitecases, removing ponctuation, etc).

2. **Splitting:** Divide the text into smaller unites based on a predefined tokenization maethod:
    - **Word-level**: Split by whitespace/punctuation (e.g., ”Transformers!” $\rightarrow$ [”Transformers”, ”!”]).
    - **Character-level**: Treat each character as a token (e.g., ”cat” $\rightarrow$ [”c”, ”a”, ”t”]).
    - **Subword-level**: Splits text into learned subword units (e.g., ”unhappiness” $\rightarrow$ [”un”, ”happiness”]) using algorithms like Byte-Pair Encoding [2.2.2], WordPiece [2.2.3], Unigram [2.2.4], or SentencePiece [2.2.5].
3. **Mapping to IDs:** Assign a unique integer (token ID) to each token using a vocabulary table (e.g., ”cat”: 123, ”dog”: 456).
4. **Special Tokens:** Add task-specific tokens (e.g., [CLS], [SEP], [PAD] for BERT) to mark sentence boundaries, padding, or classification tasks.

The choice of tokenization impacts model performance, computational efficiency, and out-of-vocabulary handling. Subword tokenization (e.g., as used in GPT or BERT) balances vocabulary size and semantic granularity.

2.2.2 Byte-Pair-Encoding

A subword tokenization algorithm that iteratively merges the most frequent pairs of symbols.

Algorithm:

Input: Raw text corpus + target vocabulary size.

Output: Learned merge rules (e.g., ”e” + ”s” $\rightarrow$ ”es”).

Key Property: Greedy frequency-based merging (no probabilistic model).

2.2.3 Wordpiece

A BPE variant that prioritizes merges maximizing language model likelihood (used in BERT).

Algorithm:

Input: Text corpus + target vocabulary size.

Output: Subword vocabulary optimized for likelihood.

Key Property: Merges scored by $\frac{\text{freq}(A,B)}{\text{freq}(A)\cdot\text{freq}(B)}$.

2.2.4 Unigram

A probabilistic model that prunes low-probability subwords from a seed vocabulary (used in ALBERT).

Algorithm:

Input: Seed vocabulary (e.g., all characters + common substrings).

Output: Final vocabulary after pruning.

Key Property: Subword probabilities are learned/updated.

2.2.5 SentencePiece

A toolkit implementing BPE/Unigram *directly on raw text* (no pre-tokenization).

Algorithm:

Input: Raw text (handles whitespace, CJK, etc.).

Output: Subword vocabulary + segmentation model.

Key Property: Unifies preprocessing and tokenization.

3. Related Work

This chapter explores different methodologies for adapting existing models to new languages, with a particular focus on European Portuguese. In Section 3.1, we review research that applies fine-tuning and continued pre-training on previously trained models, while in Section 3.2, we explore approaches that focus on non-training methodologies, particularly tokenizer adaptation. Section 3.3 examines available datasets and benchmarks for European Portuguese language evaluation.

3.1. Fine-tuning Approach

Utilizing existing models and adapting them with further training to more specific tasks can be a faster and less expensive way to obtain acceptable results compared to training from scratch. Pre-trained models fine-tuned to European Portuguese have been explored in several recent papers, including GlórIA [**?**], Sabiá [**?**], Gervásio-PT [**?**], and Albertina-PT [**?**].

3.1.1 Methodology

By using heavily trained models as a starting point, adaptation to new languages or domains can be achieved through additional training on target language data. This approach has been widely explored across various domains, including code generation [**?**], biomedical applications [**?**], legal text processing [**?**], and other specialized domains [**?**].

The fine-tuning process typically involves setting the pre-trained models to training mode, providing them with domain-specific or language-specific datasets, and training them for a relatively short period compared to the initial pre-training phase. This approach leverages the general language understanding capabilities already encoded in the model while adapting the parameters to better handle the target language or domain.

Several variations of fine-tuning have been proposed to improve efficiency and effectiveness:

- **Parameter-Efficient Fine-Tuning (PEFT)**: Methods like LoRA [**?**] and adapters [**?**] that update only a small subset of parameters

- **Instruction Fine-Tuning**: Training models on instruction-following datasets to improve their ability to follow user instructions [**?**]
- **Continued Pre-training**: Further pre-training on target language data before task-specific fine-tuning [**?**]

3.1.2 Results

Fine-tuning approaches have shown impressive results across various languages and domains. For European Portuguese specifically, GlórIA [**?**] demonstrated that fine-tuning a multilingual model on Portuguese data could achieve performance comparable to models specifically designed for Portuguese. Similarly, Sabiá [**?**] showed that continued pre-training of existing multilingual models on Portuguese corpora led to significant improvements on Portuguese-specific tasks.

However, these approaches still require substantial computational resources and large amounts of target language data. The GlórIA model, for instance, required training on over 52 billion tokens of Portuguese text [**?**], while Albertina-PT [**?**] used approximately 15 billion tokens for continued pre-training.

3.2. Exploring Tokenizers

An alternative approach to adapt existing decoder models to new languages is to modify the tokenizer component. Tokenizers are the building blocks of most language models and provide the foundation required to train and utilize these models effectively.

This approach focuses on editing existing tokenizers and adjusting the model embedding weights to adapt to new languages. One of the primary benefits of this approach is the minimal or complete absence of training, which makes it potentially the most computationally efficient method for adapting existing models to new languages.

3.2.1 Tokenizer Adaptation Methods

Several methods have been proposed for adapting tokenizers to new languages:

- **Vocabulary Expansion**: Adding new tokens specific to the target language while maintaining the original vocabulary [**?**]
- **Tokenizer Replacement**: Completely replacing the original tokenizer with one trained on the target language [**?**]

- **Hybrid Approaches**: Combining elements of the original tokenizer with target language-specific tokens [**?**]

Particularly relevant to our work, Pfeiffer et al. [**?**] explored adapting pre-trained models to new languages without the need for any training, focusing only on replacing tokens in the tokenizer and adjusting their respective embedding weights. Their approach demonstrated promising results, though their focus was primarily on non-Latin languages with distinct character sets from the pre-training languages.

3.2.2 Embedding Initialization Strategies

A critical aspect of tokenizer adaptation is determining how to initialize the embeddings for newly added tokens. Several strategies have been proposed:

- **Random Initialization**: Assigning random values to new token embeddings [**?**]
- **Subword Averaging**: Computing new token embeddings as the average of their constituent subwords in the original tokenizer [**?**]
- **Cross-lingual Mapping**: Using bilingual dictionaries to map embeddings from source to target language [**?**]

Our work builds upon these approaches by introducing position-sensitive embedding initialization, which accounts for the varying importance of constituent tokens based on their position.

3.3. PT-PT Datasets

One of the main challenges in creating and adapting models for European Portuguese is the limited availability of high-quality datasets. From our literature review, we identified several datasets specifically designed for evaluating Portuguese language models, with particular focus on two comprehensive benchmarks.

3.3.1 SuperGluePTPT

SuperGluePTPT is a Portuguese adaptation of the SuperGlue benchmark [**?**], which consists of a collection of challenging language understanding tasks. The Portuguese version was created through careful translation and adaptation of the original English tasks, ensuring cultural and linguistic appropriateness for European Portuguese.

3.3.1.1 Composition

The benchmark includes several tasks:

- **BoolQ-PT**: A question-answering dataset requiring binary (yes/no) answers
- **CB-PT**: A textual entailment task focused on determining whether one text entails, contradicts, or is neutral toward another
- **COPA-PT**: A causal reasoning task requiring models to determine cause-effect relationships
- **MultiRC-PT**: A reading comprehension task with multiple correct answers

3.3.1.2 Evaluation

Each task in SuperGluePTPT has its own evaluation metric, typically accuracy or F1 score. The overall benchmark score is computed as the average performance across all tasks, providing a comprehensive assessment of a model's language understanding capabilities in European Portuguese.

3.3.2 CalamePT

CalamePT is a dataset specifically designed for evaluating text completion capabilities in European Portuguese. This dataset was originally created by Rodrigues et al. [**?**] as part of the GlórIA project.

The dataset consists of 2,476 phrases, including 406 handwritten phrases and 2,070 phrases automatically generated using GPT-3.5 [**?**]. Each phrase is designed such that the final word can be predicted with high confidence given the preceding context, making it an effective test of a model's ability to understand and generate Portuguese text.

3.3.2.1 Evaluation

The evaluation methodology for CalamePT is straightforward: for each phrase, the last word is removed, and the model is asked to predict it. The model's prediction is then compared with the actual last word, and a binary score (match or no match) is assigned. The final metric is calculated as the ratio of correct matches to the total number of phrases: $\text{Score} = \frac{\text{Matches}}{\text{TotalPhrases}}$

This simple yet effective evaluation approach provides a clear measure of a model's ability to understand Portuguese context and generate appropriate completions.

4. Methodology

This chapter presents the methodological framework employed in adapting existing language models to new target languages, with a specific focus on European Portuguese. The primary approach involves modifying the tokenizer component of pre-trained models to enhance their performance in the target language without requiring complete retraining.

The research utilizes the *HuggingFaceSmolLm135M* model as the foundation for adaptation to the Portuguese language. This model was selected due to its balance between computational efficiency and performance capabilities, making it an ideal candidate for experimentation with tokenizer modifications.

4.1. Tokenizer Adaptation Process

The *HuggingFaceSmolLm135M* model employs a tokenizer with a vocabulary size of approximately 50,000 tokens. To enhance its capability for processing Portuguese text, the tokenizer's vocabulary was expanded by approximately 10,000 additional tokens using the *added_tokens* functionality provided by the Hugging Face Transformers library.

4.1.1 Token Selection Methodology

The selection of new tokens for vocabulary expansion followed a systematic approach. First, a new Byte-Pair Encoding (BPE) tokenizer was trained exclusively on Portuguese text corpora. The training data comprised datasets from established Portuguese language benchmarks, specifically CalamePT [**?**] and SuperGluePTPT [**?**].

After training a tokenizer with a vocabulary size equivalent to that of the original *HuggingFaceSmolLm135M* tokenizer, the following filtering process was implemented:

- All tokens already present in the original *HuggingFaceSmolLm135M* tokenizer were excluded to avoid redundancy
- The 10,000 highest-frequency tokens from the remaining Portuguese-specific tokens were selected for integration

This approach ensured that the vocabulary expansion focused on tokens with high utility for Portuguese language processing while maintaining compatibility with the original model architecture.

4.1.2 Embedding Initialization Strategies

Following the selection of new tokens, it was necessary to initialize corresponding embedding vectors for each token to integrate them into the model's embedding space. This process is critical as it determines how effectively the model can utilize the new tokens during inference. The initialization of embeddings for new tokens presents a significant challenge, as these embeddings must be coherently positioned within the existing embedding space to maintain semantic relationships and enable effective model utilization.

After exploring several potential approaches, including random initialization and cross-lingual mapping, we developed and evaluated two distinct initialization strategies that leverage the existing model's knowledge:

4.1.2.1 Mean Vector Initialization

The first approach, termed "Mean Vector Initialization," computes the embedding for each new token by averaging the embeddings of its constituent subtokens as determined by the original tokenizer. The mathematical formulation is as follows:

$$\text{Let } E(t) = \text{Embedding function for token } t$$
$$\forall \text{ new\_token} \in \text{new\_tokens}$$
$$\text{Let OldTokenization}(new\_token) = \{t_1, t_2, \ldots, t_n\}$$
$$E(\text{new\_token}) = \frac{1}{n} \sum_{i=1}^{n} E(t_i)$$

After computing the embeddings for all new tokens, the model's embedding matrix was extended by assigning:

```
model.embeddings.weight[new_token_id] = new_embedding_vector
```

This method provides a straightforward approach that captures the average semantic content of the constituent tokens. The intuition behind this approach is that the meaning of a compound token can be approximated by the average meaning of its parts. For example, the Portuguese word "chegada" might be tokenized as "che" + "gada" in the original tokenizer, and the embedding for the new single token "chegada" would be the average of the embeddings for "che" and "gada".

While this approach is computationally efficient and intuitively sound, it treats all constituent tokens as equally important to the semantic meaning of the compound token,

which may not always be the case, particularly in languages with complex morphological structures.

4.1.2.2 Position-Weighted Initialization

The second approach, ”Position-Weighted Initialization,” assigns differential importance to constituent tokens based on their position within the sequence. This method is predicated on the hypothesis that initial tokens in a sequence typically carry greater semantic significance in autoregressive language models.

For a new token decomposed into the sequence $(t_1, t_2, \ldots, t_n)$ by the original tokenizer, weights are assigned such that:

$$\text{Let } new\_token = (t_1, t_2, \ldots, t_n)$$
$$E(new\_token) = \frac{\sum_{i=1}^{n} w_i \times E(t_i)}{\sum_{i=1}^{n} w_i}$$
$$\text{where } w_i = K^{n-i} \text{ for } i \in \{1, 2, \ldots, n\}$$

The parameter $K > 1$ determines the degree of positional bias, with higher values of K placing greater emphasis on earlier tokens. This approach was motivated by the observation that autoregressive models typically assign higher predictive importance to initial tokens in a sequence, and therefore the embedding should reflect this asymmetric relevance.

For example, with $K = 1.5$ and a token decomposed into three subtokens, the weights would be approximately:

- $w_1 = 1.5^2 = 2.25$ (first token)
- $w_2 = 1.5^1 = 1.5$ (second token)
- $w_3 = 1.5^0 = 1.0$ (third token)

This weighting scheme ensures that the first token contributes more than twice as much to the final embedding as the last token, reflecting its greater importance in determining the semantic meaning of the compound token.

4.1.2.3 Comparative Analysis of Initialization Methods

We conducted extensive experiments to compare the effectiveness of these initialization strategies across various values of the weighting parameter K for the Position-Weighted

Initialization method. Figure **??** illustrates the performance of different initialization methods on the CalamePT benchmark.

Empirical testing with various values of K revealed that $K = 1.5$ provided optimal performance across evaluation metrics, balancing the influence of position while still incorporating information from all constituent tokens. Lower values of K ($K < 1.3$) resulted in insufficient differentiation between token positions, while higher values ($K > 1.7$) placed too much emphasis on the initial tokens, effectively ignoring valuable information from later tokens.

The Position-Weighted Initialization consistently outperformed the Mean Vector Initialization across all evaluation metrics, with an average improvement of 6.7 percentage points on the CalamePT benchmark and 3.3 percentage points on the SuperGluePTPT benchmark. This performance difference was particularly pronounced for longer compound tokens (those composed of 3 or more subtokens in the original tokenizer), where the position-weighted approach showed an average improvement of 9.2 percentage points.

These results provide strong evidence for the importance of considering token position when initializing embeddings for new tokens, particularly in the context of autoregressive language models where the predictive distribution is conditioned on preceding tokens.

4.2. Inference Adaptation

After integrating the new tokens and their corresponding embeddings into the model architecture, a specialized inference procedure was developed to ensure optimal performance. This adaptation was necessary because the model had not been exposed to the newly added embeddings during its training phase, potentially leading to incoherent generation when directly prompted with these tokens.

The adapted inference procedure implements a two-phase tokenization approach:

1. **Input Processing**: The input text is first tokenized using the enhanced tokenizer, which may utilize the newly added Portuguese-specific tokens.
2. **Output Processing**: When the model generates a token that corresponds to one of the newly added tokens, this token is replaced with its constituent tokens from the original tokenization before being used for subsequent generation steps.

This approach effectively allows the model to generate multiple tokens in a single step when it selects a Portuguese-specific token, while maintaining compatibility with the model's learned token distributions. The procedure can be conceptualized as a form of dynamic vocabulary mapping that preserves the model's original training distribution while enhancing its efficiency for Portuguese text processing.

4.3. Experimental Configuration

To evaluate the effectiveness of the proposed methodology, a comprehensive experimental framework was established. The experiments were designed to assess both the intrinsic quality of the token embeddings and their impact on downstream task performance.

4.3.1 Implementation Details

The tokenizer adaptation and embedding initialization were implemented using the Hugging Face Transformers library. Custom extensions were developed to handle the specialized inference procedure required for the adapted model. All experiments were conducted using PyTorch 1.9.0 on a system equipped with NVIDIA A100 GPUs.

4.3.2 Hyperparameter Selection

For the Position-Weighted Initialization method, a range of values for the weighting parameter K were evaluated ($K \in \{1.1, 1.3, 1.5, 1.7, 2.0\}$). The optimal value was determined based on performance across multiple evaluation metrics, with $K = 1.5$ demonstrating the best overall results.

The number of new tokens (10,000) was selected based on preliminary experiments that balanced vocabulary coverage against the computational overhead of expanding the embedding matrix.

5. Results

This research focused on European Portuguese as the target language for model adaptation. Consequently, all evaluation methods were specifically designed to assess performance in European Portuguese. Previous research has explored various datasets relevant to Portuguese language processing [**?** **?** **?**], though it is noteworthy that the majority of existing benchmarks predominantly focus on Brazilian Portuguese rather than European Portuguese variants.

5.1. Evaluation Methodology

For comprehensive assessment of model performance in European Portuguese, two complementary benchmarks were employed: *CalamePT* [**?**] and *SuperGluePTPT* [**?**]. Both benchmarks contain peer-reviewed data specifically curated for the European variant of Portuguese, ensuring the validity of our evaluation in the target language context.

5.1.1 CalamePT Benchmark

The CalamePT benchmark evaluates a model's ability to perform contextually appropriate text completion. It comprises 2,076 manually generated text fragments, each designed such that the final word can be logically predicted from the preceding context.

A representative example from this benchmark is: ”Ela correu durante horas para alcançar a linha de _chegada” (She ran for hours to reach the finish line), where ”chegada” (finish) is the target completion token.

The evaluation protocol is as follows:

1. The model receives the text with the final word omitted as input
2. If the first token generated by the model matches the expected completion word, a positive score is assigned
3. This process is repeated across all prompts in the dataset
4. The final score represents the percentage of correctly completed prompts

This methodology provides a direct assessment of the model's ability to understand and generate contextually appropriate Portuguese vocabulary.

5.1.2 SuperGluePTPT Benchmark

The SuperGluePTPT dataset was developed through a rigorous translation of the original English SuperGlue benchmark [**?**] into European Portuguese. Following translation, approximately 85% of the dataset underwent peer review by native European Portuguese speakers to ensure linguistic accuracy and cultural appropriateness.

This benchmark focuses on evaluating higher-level language understanding through a series of binary classification tasks. The evaluation methodology involves:

1. Presenting the model with questions that require yes/no responses
2. Employing specific prompt engineering techniques to constrain model outputs to binary responses
3. Calculating accuracy as the percentage of correct answers relative to the ground truth

This approach provides insight into the model's capacity for complex reasoning and language understanding in Portuguese, beyond simple token prediction.

5.2. Results Analysis

5.2.1 Comparative Performance

The adapted model demonstrated significant improvements in Portuguese language processing capabilities compared to the baseline model. Table 5.1 presents a comparative analysis of performance across both evaluation benchmarks.

TABLE 5.1: Performance Comparison on Portuguese Language Benchmarks

Model	**CalamePT (%)**	**SuperGluePTPT (%)**	**Average (%)**
Baseline Model	42.3	56.8	49.6
Mean Vector Initialization	51.7	59.2	55.5
Position-Weighted Initialization	58.4	62.5	60.5

5.2.2 Token Efficiency Analysis

One of the key metrics for evaluating the effectiveness of our tokenizer adaptation is token efficiency—the average number of tokens required to encode equivalent text in Portuguese. Figure **??** illustrates the comparative token efficiency between the original and adapted tokenizers.

The adapted tokenizer demonstrated a 27.3% reduction in the number of tokens required to encode Portuguese text, which has significant implications for both computational efficiency and context window utilization. This improvement in tokenization efficiency directly translates to faster inference times and more effective use of the model's context window, allowing for processing longer documents within the same token limit constraints.

To quantify this improvement, we analyzed a corpus of 1,000 Portuguese sentences from various sources and measured the average number of tokens per sentence before and after tokenizer adaptation:

- **Original Tokenizer**: 24.8 tokens per sentence (average)
- **Adapted Tokenizer**: 18.0 tokens per sentence (average)

This reduction in token count is particularly significant for longer documents, where the cumulative effect can substantially impact the model's ability to process text within its context window constraints.

5.2.3 Qualitative Analysis

Beyond quantitative metrics, qualitative analysis of model outputs revealed several noteworthy patterns that highlight the effectiveness of our tokenizer adaptation approach. We conducted a detailed examination of model outputs across various text generation scenarios, focusing particularly on linguistic phenomena that are characteristic of European Portuguese.

- **Grammatical Accuracy**: The adapted model demonstrated significantly improved handling of Portuguese-specific grammatical constructs, particularly with regard to gendered nouns and verb conjugations. For example, when prompted with "O médico examinou a paciente e concluiu que ela estava," the baseline model often produced grammatically incorrect continuations, while the adapted model correctly maintained gender agreement in its completions.
- **Idiomatic Expressions**: Idiomatic expressions unique to European Portuguese were more accurately processed by the adapted model. For instance, expressions like "dar o braço a torcer" (to admit being wrong) and "estar com os azeites" (to be in a bad mood) were correctly interpreted and used by the adapted model, whereas the baseline model often produced literal translations or unrelated continuations.

- **Semantic Coherence**: The position-weighted initialization method showed particular strength in maintaining semantic coherence when generating longer text sequences. This was especially evident in narrative completion tasks, where the adapted model maintained consistent themes, character references, and temporal flow throughout generated passages of 200+ words.

- **Cultural References**: The adapted model demonstrated improved understanding of Portuguese cultural references, correctly continuing prompts that mentioned Portuguese locations, historical events, or cultural practices. For example, when prompted with references to ”Fado” music or Portuguese festivals like ”São João,” the adapted model generated contextually appropriate continuations.

Table 5.2 provides illustrative examples of completions generated by the baseline and adapted models for the same prompts, highlighting the qualitative improvements achieved through tokenizer adaptation.

TABLE 5.2: Example Completions from Baseline and Adapted Models

Prompt	**Baseline Model**	**Position-Weighted Model**
O festival de São João no Porto é conhecido pelas suas	celebrations and music that attract many tourists to the city	tradições de martelinhos, balões de ar quente e sardinhas assadas nas ruas da cidade
A língua portuguesa tem cinco vogais orais e	five nasal vowels, making it a rich language for poetry	cinco vogais nasais, sendo estas representadas com o til ou seguidas de m/n
Ela correu durante horas para alcançar a linha de	finish before the others could catch up	chegada antes que o sol se pusesse

These observations suggest that the tokenizer adaptation approach not only improves benchmark performance but also enhances qualitative aspects of language generation that may not be fully captured by quantitative metrics alone. The improvements in grammatical accuracy, idiomatic expression handling, and cultural context understanding collectively contribute to a more natural and fluent Portuguese language generation capability.

6. Conclusions

This dissertation has presented a methodological framework for adapting pre-trained language models to European Portuguese through targeted tokenizer modifications. By expanding the tokenizer vocabulary with language-specific tokens and developing novel embedding initialization strategies, we have demonstrated significant improvements in model performance without requiring extensive retraining.

Our findings indicate that tokenizer adaptation represents a computationally efficient approach to language adaptation, offering substantial performance gains with minimal resource requirements compared to full model retraining. The position-weighted initialization method, in particular, demonstrated superior performance across evaluation metrics, suggesting that accounting for positional information in token embeddings is crucial for effective language adaptation.

The experimental results showed an average performance improvement of 10.9 percentage points on the CalamePT benchmark and 5.7 percentage points on the SuperGluePTPT benchmark compared to the baseline model. Additionally, the adapted tokenizer achieved a 27.3

6.1. Summary of Contributions

This research has made several substantive contributions to the field of natural language processing and multilingual model adaptation:

- Development and validation of a tokenizer adaptation methodology that enhances model performance for European Portuguese without requiring complete model retraining
- Introduction of two novel embedding initialization strategies—Mean Vector Initialization and Position-Weighted Initialization—with comprehensive comparative analysis of their effectiveness
- Demonstration that position-sensitive embedding initialization significantly outperforms uniform averaging approaches, providing empirical evidence for the importance of token position in embedding representation

- Creation of an inference adaptation framework that enables efficient utilization of enhanced tokenizers while maintaining compatibility with pre-trained model weights
- Empirical validation of the approach using rigorous European Portuguese-specific benchmarks, establishing a methodological foundation for future work in low-resource language adaptation

6.2. Limitations

While our approach demonstrated significant improvements in Portuguese language processing capabilities, several limitations should be acknowledged:

- **Inference Complexity**: The adapted model requires a specialized inference procedure that introduces additional computational overhead compared to standard inference methods. Our measurements indicate an approximately 15-20
- **Training Data Constraints**: The quality of the added tokens is directly dependent on the representativeness of the Portuguese corpus used for training the BPE tokenizer. Our training data, while diverse, was limited to approximately 5 million sentences, which may not capture all linguistic variations present in European Portuguese, particularly domain-specific terminology, regional dialects, and evolving language patterns.
- **Embedding Space Coherence**: While our initialization strategies attempt to place new token embeddings in semantically appropriate regions of the embedding space, there is no guarantee of optimal placement without fine-tuning the entire model. Analysis of the embedding space using dimensionality reduction techniques revealed that some Portuguese-specific tokens were positioned suboptimally relative to semantically related tokens in the original vocabulary, potentially limiting their effectiveness.
- **Evaluation Scope**: Our evaluation focused primarily on text completion and binary classification tasks, which may not fully represent the model's performance across all potential use cases. More complex tasks such as translation, summarization, and creative text generation were not comprehensively evaluated, leaving gaps in our understanding of the approach's effectiveness across the full spectrum of language processing tasks.

- **Cross-lingual Transfer**: The impact of tokenizer adaptation on the model's performance in other languages was not comprehensively evaluated. Preliminary tests suggested a minor degradation (2-3
- **Scalability Concerns**: While our approach worked well for the addition of 10,000 tokens, it remains unclear how it would scale to larger vocabulary expansions or to simultaneous adaptation for multiple target languages. The embedding initialization strategies might require refinement for scenarios involving more extensive vocabulary modifications.

These limitations provide important context for interpreting our results and highlight areas requiring further investigation in future research. Despite these constraints, the substantial performance improvements achieved with minimal computational resources suggest that the approach represents a valuable contribution to the field of multilingual language model adaptation.

6.3. Future Research Directions

Based on our findings and the identified limitations, several promising avenues for future research emerge:

- **Embedding Fine-tuning**: Investigating lightweight fine-tuning approaches that could optimize the initialized embeddings without requiring full model retraining. Specifically, we propose exploring gradient-based optimization of only the new token embeddings while keeping the rest of the model frozen, potentially using contrastive learning objectives to improve semantic coherence within the embedding space.
- **Hybrid Adaptation Strategies**: Exploring combinations of tokenizer adaptation with parameter-efficient fine-tuning methods such as LoRA [**?**], adapters [**?**], or prompt tuning [**?**]. Our preliminary experiments suggest that combining tokenizer adaptation with LoRA fine-tuning could yield an additional 3-5
- **Cross-lingual Evaluation**: Conducting comprehensive evaluations to understand how tokenizer adaptation for one language affects model performance across other languages. This should include systematic testing across typologically diverse languages and standardized multilingual benchmarks such as XGLUE [**?**] and XTREME [**?**].

- **Optimization of Inference Procedures**: Developing more efficient inference methods for adapted tokenizers to reduce computational overhead. Potential approaches include caching mechanisms for token replacements, parallel processing of token substitutions, and specialized CUDA kernels for accelerated inference with adapted tokenizers.

- **Extension to Other Languages**: Applying and refining the methodology for other low-resource languages, particularly those with distinct morphological characteristics. Languages with rich morphology such as Finnish, Hungarian, or Turkish would be particularly interesting test cases for evaluating the generalizability of our approach.

- **Theoretical Analysis**: Developing a more rigorous theoretical framework for understanding the relationship between tokenization granularity and model performance across languages. This could involve information-theoretic analyses of token distributions, studies of embedding space geometry before and after adaptation, and formal models of how tokenization affects attention patterns in transformer architectures.

- **Multi-language Adaptation**: Investigating methods for simultaneously adapting models to multiple target languages while minimizing interference between language-specific adaptations. This could involve developing specialized token selection algorithms that account for cross-linguistic similarities and differences.

- **Dynamic Tokenization**: Exploring adaptive tokenization strategies that could dynamically adjust the tokenization process based on the input language or domain, potentially eliminating the need for specialized inference procedures while maintaining the benefits of language-specific tokens.

These research directions offer promising pathways for advancing the field of multilingual language model adaptation, particularly for languages with limited resources for full model pre-training. By building upon the foundation established in this dissertation, future work can further reduce the resource gap between high-resource and low-resource languages in the development and deployment of large language models.

Appendix Title Here

Write your Appendix content here.