Experimenting with Large Language Models for Poetic Scansion in Portuguese: A Case Study on Metric and Rhythmic Structuring

André Valença and Filipe Calegario
Centro de Informática
Universidade Federal de Pernambuco
Recife, Brazil
aaav@cin.ufpe.br, fcac@cin.ufpe.br

Abstract

Poetic scansion — segmenting verses into syllables and identifying stressed positions — is an essential yet challenging analytical step in structured poetry analysis. Despite significant progress in language modeling, automatic scansion in Portuguese poetry remains underexplored, particularly considering rhythmic complexity and phonetic ambiguity. This study investigates how Large Language Models (LLMs) can effectively identify poetic syllables and metric patterns in Portuguese texts. We compare four approaches (*zero-shot prompting*, *few-shot prompting*, *chain-of-thought reasoning*, and *fine-tuning*) to assess accuracy in syllabic segmentation and rhythm detection. While prompting-based techniques exhibit moderate limitations due to phonetic variability and contextual nuances, a targeted *fine-tuning* approach yields better results, achieving an 88.6% syllabic segmentation accuracy and a 97.4% metric correspondence within a ±1 syllable tolerance threshold. Findings underscore both the promise of fine-tuned LLMs for computational poetic analysis and the challenges posed by linguistic variability. As part of developing the Pajeú platform, which aims to empower Northeastern Brazilian folk poetry communities through computational tools, this research sets a foundation for future investigations into culturally informed computational scansion methods. It highlights avenues for further improvements, such as addressing phonemic disambiguation and reducing computational training costs.

Introduction

With advancements in Artificial Intelligence, there have been successive qualitative leaps in Natural Language Processing (NLP). New techniques based on Deep Learning, such as *large language models* (LLMs), have emerged to assist in text generation, though often favoring the English language due to, among other factors, its extensive available *corpus* online.

The challenge becomes even more significant when dealing with structured texts, such as certain forms of poetry. While free verse, popularized by authors like Walt Whitman in the mid-19th century (and later by Brazilian modernist poets) (da Cunha and Cintra 2017, p.710), exists as a prominent style, metrically structured verse remains a fundamental element of songs and poetry. In the countryside of Northeast Brazil, for instance, among poets and singers, it remains the norm.

"In the world of *cantoria* (improvised sung poetry), there are three pillars that a poet must master: rhyme, meter, and meaning" (Caldas 2021), states a report on the so-called "poet of the absurd", Zé Limeira. Rhyme refers to the phonetic pairing from the last stressed syllable of a word (or verse); meter refers to the division of poetic syllables within a verse and its rhythmic pattern; while meaning is the poem's theme, generally conveying a thought or a story. By producing *nonsense* verses ("last year I died, but this year I won't"), Zé Limeira mastered the first two but not necessarily the third.

Analogously, this relationship is inverted in the case of LLMs. When properly handled (and when they do not hallucinate), advanced language models produce coherent and contextually relevant text. However, since they operate as predictive systems—where the primary mechanism is estimating the probability of the next *token*—LLMs inherently struggle with performing systematic tasks. The "atom" of LLMs is the *token*, an arbitrary segmentation determined by training corpus processing, indexing the most common fragments. This unit is not necessarily a word, syllable, letter, or phoneme. Success in dividing syllables or forming rhymes is thus governed by a probabilistic system that, while increasingly precise, remains imperfect.

Given these challenges, this research aims to contribute to the automatic generation of poetry in Portuguese. The goal is to develop a system that assists in creating metrically structured verses by leveraging different LLM-based generation approaches. Specifically, this work aimed to use GPT-4 and/or GPT-3.5 models to reliably perform poetic scansion (identifying metrical structure and a verse's rhythmic pattern).

This project is part of a broader research initiative to develop a platform for poetic training, text generation, and literary games focused on Northeastern Brazilian folk poetry. This platform is titled *Pajeú*.

The Pajeú Platform

Pajeú is the name of a river that flows through Northeastern cities known for their strong tradition in folk poetry, the most notable being São José do Egito in Pernambuco. Many renowned poets, such as Cancão, Antônio Marinho,

and Lourival Batista (Loro do Pajeú), originate from this region. The area's poetic legacy is so profound that a local legend claims that drinking water from the Pajeú River grants one the gift of rhyme. In his book *Roteiro de Velhos Cantadores e Poetas Populares do Sertão* (Wilson 1986), researcher Luís Wilson argues that composing poetry, particularly in improvisational contexts, is a skill honed through extensive practice and memory training.

> "True master poets are agile thinkers. They are undoubtedly talented individuals, but they are also meticulously trained in poetic performance. If they appear to improvise effortlessly in front of an audience—and indeed, they do improvise often—it is because their talent is built upon a vast repertoire accumulated through long years of practice and experience". (Wilson 1986, p.39, our translation)

With this perspective in mind, the proposed platform aims to serve as a training ground for poets to refine their craft. Users can consult rhymes, generate images to inspire their poems, receive thematic prompts (*motes*) to incorporate into their verses, and, at an advanced level, engage in verbal duels against the system. These poetic duels, known as "pelejas" or "duels", involve two poets exchanging arguments or playful insults through verse.

To complete this framework, the platform will integrate low-latency speech-to-speech functionality to preserve the often-oral nature of these interactions. The system will not only convert spoken input into text but also deliver its responses in audio format. This functionality positions the project at the intersection of creative assistance, automatic text generation, and human-computer collaboration.

> "Creativity support tools augment and enhance human creativity, such as Adobe's Photoshop or Computer Aided Design tools. Generative systems produce creative artifacts (semi-)autonomously, such as computers that paint pictures [...] or generate poetry. [...] Computer colleagues collaborate with human users on creative tasks much like another human would" (Davis et al. 2015, p.110).

Within this context, the poetic scansion tool proposed in this research is a key component for ensuring metrical accuracy in generated verses. It functions as part of a broader mechanism (Pajeú), which will be built to offer multiple functionalities such as rhyme selection, access to regionally typical vocabulary, and guidance on poetic structure. Therefore, this work represents the initial steps toward developing more sophisticated creativity support tools for poets who rely on strict metrical forms.

The primary objective of this research is to develop a tool based on large language models that performs automatic scansion of poetic verses in Brazilian Portuguese, identifying metrical patterns. To achieve this, we experimented with multiple approaches such as zero-shot prompting, few-shot prompting, chain of thought reasoning, and fine-tuning, with fine-tuning emerging as a convenient approach for enhancing accuracy and consistency in metrical and rhythmic analysis.

Related projects

This section provides an overview of LLM capabilities, their application in poetic text generation, and prior computational approaches to metrical analysis, setting the foundation for our work on enhancing LLM-based scansion tools for Brazilian Portuguese poetry.

Computational Poetry

Several prior works by Hugo Gonçalo Oliveira have significantly advanced the field of computational poetry generation, particularly within the Portuguese language.

The platform PoeTryMe (Oliveira 2012), along with its extensions (Oliveira and Alves 2016; Oliveira, Mendes, and Boavida 2017a; Oliveira 2019), supports modular and versatile poetry creation, including co-creative and interactive interfaces (Oliveira et al. 2019; Oliveira, Mendes, and Boavida 2017b). These systems emphasize syntactic and semantic coherence, offering tools for producing meaningful poetic content based on structured linguistic resources.

While prior works have focused on generation, this study shifts the lens to analysis, specifically, automated scansion, while maintaining the shared goal of modeling poetic structure in Portuguese. By situating our contributions within this broader landscape, we highlight how our approach complements existing systems like PoeTryMe. Whereas PoeTryMe relies on explicit syntactic and semantic rules, we investigate the use of LLMs that implicitly learn scansion patterns from annotated examples.

LLMs and Poetic Texts

While large language models are still in the early stages of mainstream adoption, their ability to generate natural language text, often indistinguishable from human-written content, is already evident. Some studies even suggest that ChatGPT can write argumentative essays that receive higher evaluations from teachers than those written by the average high school student. (Herbold et al. 2023)

Beyond generating structured essays, the "creative" potential of these models has led to various literary experiments. **TextFX**[1], a tool developed by Google Creative Lab in collaboration with rapper Lupe Fiasco, uses Google's PaLM 2 API to explore syntactic and semantic wordplay. The creators utilize *few-shot prompting* techniques to develop acronyms, alliterations, associative word chains, phonetically similar expressions, and other literary games. (Wade 2023)

To illustrate how **TextFX** works, one of its features, called "Explode", generates wordplays by breaking a given word into smaller components that, together, sound similar to the original word. For example, if the word "*arara*" (a type of macaw) is input, one possible output could be "*air are a*" (*a bird that breathes air*). The remarkable aspect of **TextFX** is that it employs a relatively simple approach to achieve a rich and well-defined purpose. It even extracts phonetics from spelling—an exciting feature given the inconsistencies between spelling and pronunciation in English.

[1] `https://textfx.withgoogle.com/`

PoeLM[2] is a language model designed for generating structured poetry in Spanish and Basque. The underlying heuristic involves using a non-poetic text corpus, which is then segmented into punctuation-based phrases. These phrases are assigned control codes indicating their length and the ending of the last word (representing meter and rhyme). At inference time, a structure descriptor conditions the model's output to adhere to the predefined patterns and the best result is selected from multiple generated options. (Ormazabal et al. 2022) The training approach used in our project follows a similar methodology in that we also incorporate visual markers in training examples—analogous to how **PoeLM** uses control codes.

CoPoet[3] (Chakrabarty, Padmakumar, and He 2022) employs multiple T5 models (Text-to-Text Transfer Transformer) to create a collaborative poetry-writing system. The system's core is a language model trained on a corpus with explicit instructions for writing poetry. This solution is particularly interesting because it fosters collaboration between humans and AI to pursue a common goal: composing a poem. The model also serves an educational purpose, encouraging users to explore their creativity. Additionally, the training corpus consists of instruction-verse pairs, where the instruction may involve literary devices (e.g., metaphors, onomatopoeia), rhyme schemes, themes, or other poetic techniques.

Poet Vicuna uses LLaMA-13B to generate metrically structured verses in English. To achieve this, the system employs a MetricGenerator class, which "controls the metrical structure of the generated output by constraining the model's token probability distribution" (Hoover 2023, our translation). This approach relies on external filtering functions outside the LLM's default behavior—a more feasible strategy with open-source models. However, the **Poet Vicuna** platform[4] has a somewhat "mechanical" interface, requiring users to input multiple parameters, which may discourage engagement.

Despite these advancements in AI-assisted poetry generation, the field of Portuguese-language literary AI remains underdeveloped. Recent evidence suggests that LLMs (even those initially trained in diverse languages and corpora) can be strong foundations for fine-tuning domain-specific content. A notable example is Sabiá (Pires et al. 2023), Sabiá 2 (Almeida et al. 2024), and Sabiá 3 (Abonizio et al. 2025), a series of models fine-tuned on top of LLaMA series of language models to enhance Portuguese-language generation. This initiative also highlights the cultural and linguistic challenges of English-dominant training datasets. The Sabiá series of models demonstrates that refining existing large models for a specific linguistic and cultural context can be far more cost-effective than training an entirely new model from scratch. Considering Sabiá first model, its development cost was estimated at less than 3% of what would be required to pre-train a comparable model from the ground up.

[2] https://github.com/aitorormazabal/poetry_generation

[3] https://copoet-emnlp.github.io/

[4] https://replicate.com/replicate/poet-vicuna-13b

Poetic Text Processing in Computational Systems

Blending literature and computation is not a new endeavor. Various structured and algorithmic approaches have been developed over the years to address different aspects of literary text processing. Examples in Portuguese range from the primitive yet amusing Gerador de Lero-Lero[5], which generates pseudo-intellectual gibberish, to more advanced tools such as Aoidos, a poetic scansion system launched in 2016 to systematically analyze syllabic division and stress patterns in verses (Mittmann 2016).

The Aoidos platform is a cornerstone of the present research, as it provided the corpus of correctly scanned verses used in the fine-tuning phase of the GPT-3.5 model. This dataset, comprising thousands of examples, enables GPT-3.5 to learn metrical patterns more effectively. In a way, this project takes one step back to move two steps forward—leveraging structured rule-based approaches to enhance deep learning techniques in poetic scansion.

Poetic Scansion

To understand the task that the LLM must perform in this project, it is first necessary to define poetic scansion. "To scan a verse means to divide it into syllables to highlight its meter and rhythm; the verb scan corresponds to the noun scansion" (Mittmann 2016, p. 1). The "rhythmic pattern", or "rhythmic scheme", determines the stressed syllables in a verse and creates rises and falls in pronunciation by alternating stressed and unstressed syllables. The meter, on the other hand, refers to the division of poetic syllables, which differs from standard grammatical syllabification in several aspects. An example:

- nos verdes ramos → nos / * ver- / des / * ra- # / mos

The verse "*nos verdes ramos*" ("on the green branches" in English) is divided into five poetic syllables, with stresses on "*ver*" and "*ra*", represented by the asterisk (*). It is a tetrasyllable, meaning it has four counted syllables since all syllables following the last stress (in this case, only "mos") are not considered for metric purposes. The final counted syllable is represented by both * and the # symbol. This verse has an unambiguous scansion, as there is no alternative way to separate the syllables. However, this is not true for all verses.

> "To scan means to divide a verse into poetic syllables. Note that poetic syllables do not always correspond to grammatical syllables. The reader-listener may merge (or separate) syllables when vowel sequences occur, depending on the verse's melody. Each person's ear will indicate how to proceed". (Goldstein 1988, p. 16, our translation)

In other words, although syllabic division has an algorithmic aspect, it also contains a contextual and culturally specific dimension, making it subjective. A verse that most

[5] https://lerolero.bgnweb.com.br/

people might naturally scan as six syllables could, within a seven-syllable stanza[6], be "forced" into seven syllables through the application of metrical transformations (called *metaplasms*).

This work focuses specifically on Brazilian Portuguese, which exhibits phonological and prosodic features that diverge from those of its European counterpart. One notable difference is in the treatment of vowel sequences: a word pronounced one way in Brazilian Portuguese may be naturally pronounced differently in European Portuguese, particularly regarding whether hiatuses are merged (via synaloepha or syneresis) or kept separate. Since these metaplasms are optional, their application depends on the stylistic choices of the poet and/or the reader. These variations affect how syllables are counted and stresses are assigned in poetic scansion, justifying the need for a model tailored to Brazilian Portuguese. While the approach may be adapted to other variants, this would require additional phonological calibration and dedicated training data.

Approaches to Automatic Scansion

Despite the lack of a deterministic method for scansion, this project proposes using LLMs to receive a verse as input (*prompt*) and return a scanned version as output (*response*). An indicative rhythmic pattern must accompany the verse's metrical structure. In the project's chosen format, poetic syllables are separated by the "/" symbol, stressed syllables are marked with "*", and the last stressed syllable is followed by "#".

Aoidos is a system developed by Adiel Mittmann as part of his doctoral research in Computer Science at the Federal University of Santa Catarina (UFSC). It accounts for all the nuances of scansion discussed above—and more—using an algorithmic approach. The tool was created to scan verses and stanzas automatically, and for example, when receiving the following input (before the arrows), it returns the corresponding scansion output (after the arrows):

- o pato lamenta quando quer → o / pa / to / la / men / ta / quan / do / quer #
- a patinha de quem tem saudades → a / pa / ti / nha / de / quem / tem / sau / da # / des

The first verse has, unambiguously, nine poetic syllables, ending on the stress in "*quer*." The second verse is scanned in a way that applies a synizesis to the word "saudades," treating the "sau" as a single syllable to maintain the nine-syllable pattern. However, consider the following modification:

- o pato lamenta quando quer ver → o / pa / to / la / men / ta / quan / do / quer / ver #
- a patinha de quem tem saudades → a / pa / ti / nha / de / quem / tem / sa / u / da # / des

[6]A stanza is a structured unit of a poem, composed of a group of verses that follow a specific metrical and rhythmic pattern, often adhering to a particular rhyme scheme or thematic cohesion within the poem.

Now, the first verse contains ten poetic syllables, unambiguously, ending on "*ver*." This time, the system forces a diaeresis in the word "*saudades*", splitting it as "sa / u / da / des" to preserve the decasyllabic structure of the stanza.

Aoidos employs grapheme-to-phoneme conversion to determine syllables and applies naturalness criteria to pronunciation. Scanning a poem creates a base pronunciation for each verse, determines a dominant metrical pattern within a stanza, and re-scans, applying metrical transformations that adjust pronunciation to fit the rhythmic pattern.

The author explains that the generated pronunciations include a parameter called PTS, which aims to reflect their naturalness (Mittmann 2016, p 15). Initially, when a pronunciation is formed by simply concatenating the words in a verse, it receives a PTS value of zero. However, this starting point is not a neutral baseline but a reference for comparison. The PTS score functions as a relative measure, meaning that it helps compare different versions of the same verse, but not for drawing comparisons across entirely different verses.

In other words, the system applies naturalness rules based on a standardized model of Portuguese pronunciation, which the author acknowledges is inherently subjective (Mittmann and Maia 2017, p. 162). These rules become increasingly flexible as they adapt to the rhythmic constraints of a given stanza.

Rhythm and Poetic Metrics

Aoidos defines canonical rhythmic structures based on traditional Portuguese poetic forms, such as heroic verse, martelo, and sapphic verse. By scoring rhythm patterns according to probabilistic models, the system selects the most natural and metrically consistent interpretation for a given stanza.

Methodology and Results

Our research evaluated four approaches for the automatic scansion of Brazilian Portuguese verses using Large Language Models (LLMs): zero-shot prompting, few-shot prompting, chain-of-thought prompting, and fine-tuning. We conducted initial experiments with the first three approaches using a limited dataset and manual verification to establish a baseline for LLM capabilities in this task. These preliminary tests utilized both GPT-3.5-turbo-0125 and GPT-4-turbo-preview models. The final approach, fine-tuning, was conducted with a larger dataset and quantitative metrics using GPT-3.5-turbo-0125.

Zero-shot Prompting

In this approach, we evaluated the LLMs' ability to perform poetic scansion without being shown examples, only by interpreting a structured instruction. We used two models: `gpt-3.5-turbo-0125` and `gpt-4-turbo-preview`. The temperature parameter was set to 0.2 (down from the OpenAI default of 0.7) to increase determinism and reduce output randomness.

The prompt was structured to guide the model using four key elements:

- **Role**: The assistant acts as a poet's helper.

- **Context**: The user needs poetic scansion.
- **Instruction**: Divide poetic lines into syllables and mark stressed syllables.
- **Output format**: Use "/" for syllable division, "*" before stressed syllables, and "" after the final stressed syllable.

Example Prompt:

```
You are an excellent poetry assistant.

I need to scan some verses (scansion is the syllabic
↪  division of poetic lines and identification of
↪  stresses).

When I give you a poem as input, I want you to return
↪  each line scanned with its rhythmic pattern marked.

The desired output format is:
1. Syllables separated by " / ".
2. Each stressed syllable is preceded by "⋆".
3. The final stressed syllable is followed by "#".
```

The GPT-4 model followed the format reasonably well, but still made frequent errors in syllable division and stress assignment. GPT-3.5 performed worse, often failing to even tokenize properly or respect syllabic boundaries.

This experiment showed that LLMs, when prompted in a zero-shot manner, are not sufficiently accurate for poetic scansion in Brazilian Portuguese. The likely cause is a lack of phonetic and orthographic specialization in the models' training.

Few-shot Prompting

Few-shot prompting improves over the zero-shot method by including a small number of correctly scanned examples before presenting the target input. We used `gpt-4-turbo-preview` with temperature 0.2.

The model was given the following few-shot prompt, with examples scanned using the Aoidos tool:

```
You are an excellent poetry assistant.

I need to scan some verses (scansion is the syllabic
↪  division of poetic lines and identification of
↪  stresses).

When I give you a poem as input, I want you to return
↪  each line scanned with its rhythmic pattern marked.

The desired output format is:
1. Syllables separated by " / ".
2. Each stressed syllable is preceded by "⋆".
3. The final stressed syllable is followed by "#".

Examples:

USER: Quando eu canto o seu coração se abala
RESPONSE: Quando eu canto o seu coração se abala ->
↪  ⋆Quan- / do eu / ⋆can- / to o / ⋆seu / co- / ra- /
↪  ⋆ção / se a- / ⋆ba-# / lha

USER: Pois eu sou porta-voz da incoerência
RESPONSE: Pois eu sou porta-voz da incoerência -> Poi- /
↪  ⋆s eu / sou / ⋆por- / ta{ / ⋆voz / da in- / coe- /
↪  ⋆rên-# / cia

USER:
```

The model scanned 16 verses of the poem *Agalopado* by Alceu Valença, with results summarized in Table 1. Few-shot prompting significantly improved output quality compared to zero-shot prompting. Most outputs respected syllabic boundaries and occasionally captured *sinalefas*, but errors in stress assignment persisted, and the "#" marker was sometimes misused, appearing multiple times or in incorrect positions. While few-shot prompting outperformed zero-shot approaches, it still fell short of the precision required for reliable scansion, thus motivating our exploration of fine-tuning methods.

Chain-of-Thought Prompting

Poetic scansion involves a sequence of interdependent operations: identifying syllables, assigning stress, and applying phonological rules like *sinalefa*. Due to its procedural nature, we investigated whether Chain-of-Thought (CoT) prompting could improve LLM performance on this task. CoT has shown promise in reasoning-intensive tasks by guiding models to generate intermediate steps before final answers (Wei et al. 2022).

We tested the **zero-shot CoT** strategy, in which the prompt includes the steps and a reasoning cue (e.g., "Let's think step-by-step") but no example.

To test poetic CoT reasoning, we supplied a step-by-step prompt incorporating canonical versification rules, especially those by Bilac (Bilac and Passos 1905), along with simplified guidelines for syllabification and stress marking. An excerpt:

```
You are an excellent poetry assistant.

I need you to scan some verses. Scansion means dividing
↪  each poetic line into syllables and identifying the
↪  stressed ones.

Follow this process:
1. Identify all verses (one per line).
2. Identify stressed syllables for each word.
3. Apply sinalefa and synaeresis where needed.
4. Separate poetic syllables using " / ".
5. Mark stressed syllables with "⋆".
6. Mark the last stress with "#".

Example format:
" / ⋆syllable / syllable / ⋆syllable#"

Let's think step-by-step.
```

Despite clear instructions, the model's output was flawed. It frequently marked multiple # symbols, ignored *sinalefa* rules, and misassigned stresses. For example:

- 1. Do Gosto para o desgosto → Do / *Gos# / to / pa / *ra / o / des / *gos / to
- 6. O que fui hoje não sou → O / que / *fu# / i / *ho / je / não / *sou

Table 1: Few-shot prompting results for *Agalopado*

Verse	Scansion Output
Cego o sol seu cavalo de batalha	*Ce- / go / o / *sol / seu / ca- / va- / *lo / de ba- / *ta- / lha
E faço a lua brilhar no meio-dia	E / *fa- / ço a / *lu- / a / bri- / *lhar / no / *mei- / o- / *di- / a
Tempestade eu transformo em calmaria	Tem- / *pes- / ta- / de eu / trans- / *for- / mo / em / cal- / *ma- / ri- / a
E dou um beijo no fio da navalha	E / *dou / um / *bei- / jo / no / *fi- / o / da na- / *va- / lha

These results suggest that even with algorithmic cues, LLMs struggle to internalize linguistic heuristics for scansion, possibly due to insufficient training on phonological rules. CoT offered structure but did not yield reliable performance, reinforcing the need for fine-tuning approaches with explicit supervision.

Fine-tuning Approach

After observing limited success with prompt engineering techniques, we implemented a fine-tuning strategy using the Aoidos system as our ground truth. While Aoidos has demonstrated remarkable reliability in processing over 100,000 verses spanning three centuries of Portuguese poetry, it operates as a rule-based system with specific contextual dependencies. It requires complete stanzas for analysis and applies complex metaplasm rules based on the broader poetic context.

Our goal was to create a more flexible solution to handle individual verses independently, maintain natural phonetic divisions, and be easily integrated into modern NLP pipelines. By fine-tuning an LLM on Aoidos-processed data, we aimed to combine the accuracy of Aoidos's rule-based approach with the adaptability of neural models. This would allow for real-time processing of single verses, better handling of contemporary poetry variations, and easier integration with existing language models. Additionally, while Aoidos requires careful consideration of stanza-level metrics and complex phonetic rules, a fine-tuned LLM could learn to make these decisions implicitly, simplifying the scansion process while maintaining accuracy.

Model and Infrastructure The fine-tuning implementation utilized OpenAI's API with a structured message format containing system instructions, user input, and expected assistant output. We configured the base model GPT-3.5-turbo-0125 with default training temperature settings as determined by OpenAI's backend, while setting a low inference temperature of 0.1 to ensure largely deterministic outputs. This small amount of variability was intended to accommodate minor phonetic or segmentation ambiguities; however, preliminary tests with a temperature of 0.0 yielded nearly identical results, suggesting that this stochasticity had a negligible impact on performance. The training process followed OpenAI's recommended defaults, including three epochs, with batch size and learning rate managed automatically by the platform's backend systems.

For computational infrastructure, the training was conducted on OpenAI's cloud-based platform. The fine-tuning jobs were submitted via OpenAI's CLI and monitored through the OpenAI developer dashboard, which provided real-time updates on training progress, token consumption, and performance metrics. The training data was preprocessed locally and uploaded to OpenAI's storage system in JSONL format, ensuring compatibility with the fine-tuning pipeline. Each fine-tuning iteration was performed asynchronously, with model checkpoints managed internally by OpenAI's infrastructure. Upon completion, the fine-tuned model was deployed via OpenAI's API.

Dataset Preparation Our dataset preparation began with approximately 13,500 verses from Aoidos's test corpus — the data processing pipeline involved extracting individual verses and treating them as independent units to avoid contextual dependencies. We implemented batch processing of 800 verses at a time, which we determined empirically as the optimal batch size. Using BeautifulSoup for XML parsing, we extracted clean text and relevant metadata. The standardization process established consistent formatting for syllable divisions using "/" delimiters, stress markers with "*" prefixes, and end-of-count markers with "#" suffixes. We also preserved necessary metadata, including rhythmic patterns, verse classifications, and stress patterns.

To ensure data quality, we implemented rigorous validation procedures for syllable count consistency, stress marker placement, and proper end-of-count marker positioning. Duplicate verses were eliminated to prevent training bias. The final dataset was split using an 80-20 ratio for training and validation/testing, with stratified sampling to maintain the distribution of verse types.

Training Strategies We designed two training strategies to explore the relationship between data volume and model performance. The first was an iterative approach consisting of 18 sequential iterations, starting with 400 examples and incrementally adding 400 examples per iteration, eventually reaching 7,200. Each iteration preserved previous training while incorporating new data, with the test set growing proportionally at 30% of cumulative training data. The second strategy employed a single-pass training approach with 3,520 examples, designed to compare against the iterative approach while matching the data volume of middle-range iterative runs.

Both training approaches maintained consistent configurations, including OpenAI's default fine-tuning rate, automatically optimized batch sizes, and three training epochs. We set a maximum of 4,096 tokens per example during training. For model deployment, we configured strict parameters with an inference temperature of 0.1, a maximum output of 1,024 tokens, and neutral frequency and presence penalties to ensure consistent and deterministic outputs.

Performance Evaluation

To evaluate model performance, we developed three complementary metrics that assess different aspects of scansion accuracy. The Syllable Match Percentage (SYL_MATCH_%) performs an exact string comparison between the model's output and Aoidos's scansion, measuring both correct syllabic division and accurate stress marking. The Unstressed Syllable Match (SYL_MATCH_UNS_%) focuses solely on syllabic division by removing stress markers before comparison, acknowledging that stress patterns can vary in poetic recitation while maintaining valid scansion. The Syllable Count Match (SYL_COUNT_MATCH_%) compares the total number of syllables to the last stressed position, addressing the fundamental requirement of maintaining consistent metrical length in Brazilian Portuguese poetry.

We evaluated the model's performance using two training approaches: compound (iterative) and single-pass training. The compound approach achieved its best performance in the 14th iteration, reaching 88.60% syllable match (Figure 1), 92.30% unstressed syllable match (Figure 2), and 87.56% syllable count match (Figure 3). We conducted single-pass experiments with two different dataset sizes: an intermediate run with 3,520 examples achieved comparable results (87.19% syllable match, 91.45% unstressed syllable match, and 88.30% syllable count match). In contrast, a more extensive run with 7,200 examples showed marginal improvements (89.15% syllable match, 92.80% unstressed syllable match, and 89.10% syllable count match).

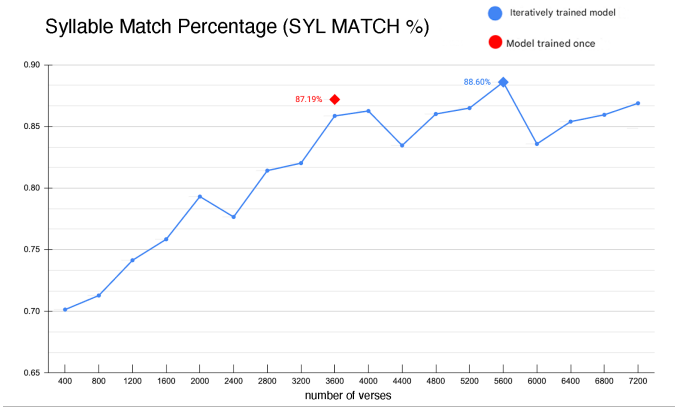

Figure 1: Syllable Match Percentage (SYL _ MATCH _ %): iterative and single-pass training

These results suggest that while larger datasets tend to improve performance, the gains are incremental, and even a moderate-sized single training run can achieve competitive results. When allowing for a ±1 syllable margin of error to account for valid poetic variations, the syllable count accuracy improved to 97.38%, 97.12%, and 97.85% for compound, intermediate single-pass, and large single-pass approaches, respectively.

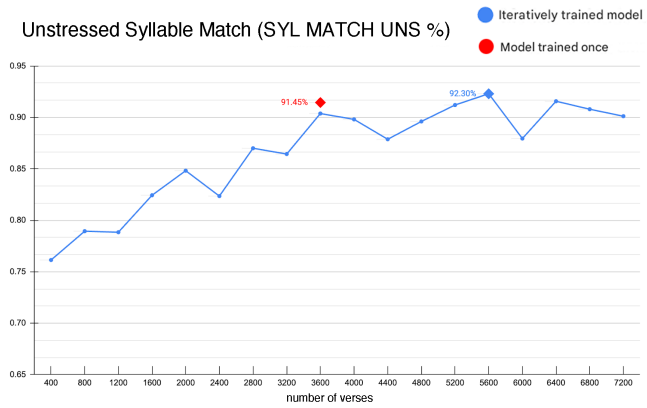

Figure 2: Unstressed Syllable Match (SYL _ MATCH _UNS _ %): iterative and single-pass training

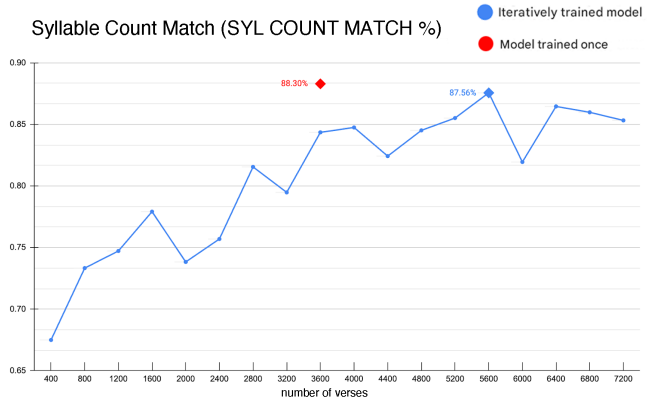

Figure 3: Syllable Count Match (SYL _ COUNT _ MATCH _ %): iterative and single-pass training

Cost Analysis

Our iterative training approach accumulated $116.43 across 18 iterations ($109.78 for training, $5.00 for prompts, and $1.42 for completions), while a single training run with 3,520 examples cost only $6.80. Inference costs remained consistent at approximately 550 tokens per scansion (505 for input, 45 for output), resulting in $0.00025 per verse at current OpenAI pricing. These costs, while minimal for individual verses, become significant when scaling to extensive poetry collections.

Linear regression analysis suggests that achieving a 100% match with Aoidos would require approximately 14,000 examples at an estimated cost of $250. However, this theoretical perfect accuracy is likely unattainable due to inherent poetic variations. We observed significant diminishing returns after the 14th iteration (5,600 examples), where we achieved 88.60% syllable match accuracy, suggesting this as an optimal efficiency threshold for practical applications. Future work might explore more cost-effective architectures

or token optimization techniques while maintaining acceptable accuracy.

Discussion

Our results demonstrate the viability of using fine-tuned LLMs for Brazilian Portuguese verse scansion while revealing significant trade-offs between accuracy, cost, and practical applicability. The progression from prompt engineering approaches to fine-tuning showed that while sophisticated prompting techniques can improve performance, they cannot match the consistency achieved through targeted model adaptation. The high accuracy rates achieved by our fine-tuned models suggest that LLMs can effectively learn the complex rules of Brazilian Portuguese prosody, including challenging aspects such as synaeresis and synalepha.

Limitations

However, several limitations merit discussion. First, while our models achieve high accuracy when compared to Aoidos's output, they inherit any biases or limitations present in Aoidos's rule-based approach. This is particularly relevant for contemporary poetry, in which traditional metrical rules might be deliberately subverted. Second, the cost-benefit analysis reveals diminishing returns in performance improvements beyond certain thresholds, suggesting practical limits to model optimization within current technical and economic constraints.

Second, while assessing the impact of training size, we employed an incremental partitioning strategy in which the test set expanded alongside the training data. We acknowledge that this may have introduced variability in the evaluation. Future experiments should fix the test set to ensure consistent and comparable assessments across training sizes.

A significant technical limitation of our approach is the vendor lock-in associated with OpenAI's fine-tuning platform. Each inference request incurs a cost, and we are dependent on the OpenAI API for both training and deployment. This introduces sustainability concerns for long-term applications and restricts potential adaptations to alternative model architectures or open-source solutions that might better serve specific linguistic contexts.

Another limitation of our study is its temporal nature - given the rapid advancement of language models, with new architectures and larger reasoning models being released frequently, our findings represent a snapshot of current capabilities rather than a definitive solution. Models with enhanced reasoning capabilities or more efficient fine-tuning approaches might soon enable new methodologies that could surpass our results.

Our analysis also revealed systematic errors in specific cases, particularly in syllable fusion and incorrect tonic markings. These errors highlight that scansion remains primarily a phonetic rather than an orthographic problem. Furthermore, while our models can learn metric patterns through fine-tuning, the inherently subjective nature of poetic interpretation and the complexities of oral performance present ongoing challenges for full automation.

Further Research Directions

These findings point to several promising directions for future research. One avenue would be to incorporate human expert annotations to create a more diverse training dataset that captures variations in poetic style and metrical interpretation. Another would be to explore more efficient model architectures or training approaches that could reduce computational costs while maintaining accuracy. Additionally, developing specialized embedding spaces for poetic features could potentially improve the model's understanding of metrical patterns while reducing training data requirements. The rapid evolution of LLM technology also suggests the importance of continuously evaluating new models and architectures as they emerge, particularly those designed for complex reasoning tasks that might be better suited to handling the nuanced rules of prosody.

Conclusion

This study demonstrated the feasibility of using Large Language Models for Portuguese verse scansion through systematic evaluation of different approaches. While GPT-4 showed better comprehension than GPT-3.5 in prompt engineering experiments, both models struggled with zero-shot and chain-of-thought approaches. Few-shot prompting improved metric and rhythmic structure understanding, but fine-tuning emerged as the most effective method. Our evaluation metrics showed that, with a robust dataset, the fine-tuned model achieved 88.6% accuracy in syllabic separation and 97.4% metric correspondence when allowing a margin of error of ±1 syllable, providing a potential foundation for automated scansion tasks.

Looking beyond technical aspects, our work raises fundamental questions about the relationship between poetry and technology. While machines can effectively scan verses and learn metrical patterns, they cannot replicate the whole poetic experience. Scansion, as a culturally conditioned process, requires understanding not just rules but also when and how these rules might be artistically broken. This understanding remains uniquely human, suggesting that while AI can be a valuable tool for literary analysis and experimentation, the essence of poetic creation and interpretation remains firmly in the human domain.

Future Work

This research paves the way for the development of the Pajeú platform, aimed at enhancing the skills of popular poets. The scansion model can be utilized both for validating automatically generated verses and guiding metric construction. Future research directions include:

- Generation of structured stanzas: Application of fine-tuning for décimas, sextilhas, and other traditional forms.
- Semantic and phonetic expansion: Use of RAG to enrich rhymes and cultural references in Northeastern Brazilian poetry.
- Orality and performance: Implementation of speech synthesis models that respect the prosody of improvised poetry.

Our results contribute to the ongoing development of computational approaches to Portuguese poetry analysis, building on prior efforts while acknowledging that poetry, in its essence, is not just about following rules—it's also about knowing when to break them.

References

[Abonizio et al. 2025] Abonizio, H.; Almeida, T. S.; Laitz, T.; Junior, R. M.; Bonás, G. K.; Nogueira, R.; and Pires, R. 2025. Sabiá-3 technical report.

[Almeida et al. 2024] Almeida, T. S.; Abonizio, H.; Nogueira, R.; and Pires, R. 2024. Sabiá-2: A new generation of portuguese large language models.

[Bilac and Passos 1905] Bilac, O., and Passos, G. 1905. *Tratado de versificação*. Rio de Janeiro, Rio de Janeiro: Livraria Francisco Alves.

[Caldas 2021] Caldas, P. 2021. Ano passado eu morri, mas esse ano eu não morro. `http://glo.bo/439ttyW`.

[Chakrabarty, Padmakumar, and He 2022] Chakrabarty, T.; Padmakumar, V.; and He, H. 2022. Help me write a poem: Instruction tuning as a vehicle for collaborative poetry writing.

[da Cunha and Cintra 2017] da Cunha, C. F., and Cintra, L. F. L. 2017. *Nova gramática do português contemporâneo*. Rio de Janeiro, Rio de Janeiro: Lexikon Editora Digital.

[Davis et al. 2015] Davis, N.; Hsiao, C.-P.; Popova, Y.; and Magerko, B. 2015. *An Enactive Model of Creativity for Computational Collaboration and Co-creation*. London: Springer London. 109–133.

[Goldstein 1988] Goldstein, N. 1988. *Análise do poema*. São Paulo, São Paulo: Editora Ática S.A.

[Herbold et al. 2023] Herbold, S.; Hautli-Janisz, A.; Heuer, U.; Kikteva, Z.; and Trautsch, A. 2023. A large-scale comparison of human-written versus chatgpt-generated essays. *Nature*.

[Hoover 2023] Hoover, J. 2023. Make any large language model a better poet. `https://replicate.com/blog/turn-your-llm-into-a-poet`.

[Mittmann and Maia 2017] Mittmann, A., and Maia, S. R. 2017. Análise comparativa entre escansões manual e automática dos versos de Gregório de Matos. (10).

[Mittmann 2016] Mittmann, A. 2016. *Escansão automática de versos em português*. Ph.D. Dissertation, Universidade Federal de Santa Catarina, Centro Tecnológico.

[Oliveira and Alves 2016] Oliveira, H. G., and Alves, A. O. 2016. Poetry from concept maps – yet another adaptation of poetryme's flexible architecture. In *Proceedings of the 7th International Conference on Computational Creativity (ICCC)*.

[Oliveira et al. 2019] Oliveira, H. G.; Mendes, T.; Boavida, A.; Nakamura, A.; and Ackerman, M. 2019. Co-poetryme: Interactive poetry generation. *Cognitive Systems Research* 54:199–216.

[Oliveira, Mendes, and Boavida 2017a] Oliveira, H. G.; Mendes, T.; and Boavida, A. 2017a. Co-poetryme: a co-creative interface for the composition of poetry. In *Proceedings of the 10th International Conference on Natural Language Generation (INLG)*, 70–71.

[Oliveira, Mendes, and Boavida 2017b] Oliveira, H. G.; Mendes, T.; and Boavida, A. 2017b. Towards finer-grained interaction with a poetry generator. In *Proceedings of ProSocrates 2017: Symposium on Problem-solving, Creativity and Spatial Reasoning in Cognitive Systems*, CEUR Workshop Proceedings, 1–10.

[Oliveira 2012] Oliveira, H. G. 2012. Poetryme: a versatile platform for poetry generation. In *Proceedings of the ECAI 2012 Workshop on Computational Creativity, Concept Invention, and General Intelligence (C3GI)*.

[Oliveira 2019] Oliveira, H. G. 2019. On the impact of the grammar and the semantic network in a versatile poetry generator. In *Proceedings of the 6th Computational Creativity Symposium at 2019 AISB Convention*.

[Ormazabal et al. 2022] Ormazabal, A.; Artetxe, M.; Agirrezabal, M.; Soroa, A.; and Agirre, E. 2022. Poelm: A meter- and rhyme-controllable language model for unsupervised poetry generation.

[Pires et al. 2023] Pires, R.; Abonizio, H.; Almeida, T. S.; and Nogueira, R. 2023. *Sabiá*. Springer Nature Switzerland. 226–240.

[Wade 2023] Wade, A. 2023. How it's made. `https://developers.googleblog.com/2023/08/how-its-made-lupe-fiasco-text-fx.html`.

[Wei et al. 2022] Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q.; and Zhou, D. 2022. Chain-of-thought prompting elicits reasoning in large language models.

[Wilson 1986] Wilson, L. 1986. *Roteiro de velhos cantadores e poetas populares do Sertão*. Recife, Pernambuco: Centro de Estudos de História Municipal.