Aoidos: A System for the Automatic Scansion of Poetry Written in Portuguese

Adiel Mittmann[1], Aldo von Wangenheim[1], Alckmar Luiz dos Santos[2]

[1]Graduate Program in Computer Science (PPGCC),
National Institute for Digital Convergence (INCOD)
[2]Center for Research in Informatics, Literature and Linguistics (NUPILL)
[1,2]Federal University of Santa Catarina (UFSC)
adiel@inf.ufsc.br, aldo.vw@ufsc.br, alckmar@cce.ufsc.br

Abstract. Scansion is the activity of determining the patterns that give verses their poetic rhythm. In Portuguese, this means discovering the number of syllables that the verses in a poem have and fitting all verses to this measure, while attempting to pronounce syllables so that an adequate stress pattern is produced. This article presents Aoidos, a rule-based system that takes a poem written in the Portuguese language and performs scansion automatically, further providing an analysis of rhymes. The system works by making a phonetic transcription of a poem, determining the number of poetic syllables that the verses in the poem should have, fitting all the verses according to this measure and looking for verses that rhyme. Experiments show that the system attains a high accuracy rate (above 98%).

1 Introduction

Poetry scansion is the task of correctly determining the metrical structure of a verse. The precise rules that must be taken into consideration when scanning a verse depend on the language, but it usually involves placing syllables with specific properties in specific positions. In languages such as Portuguese, this task can be complicated because a verse can be pronounced in many ways, depending on how one chooses to join adjacent vowels.

The objective of this article is to describe Aoidos, a system that automatically performs the scansion of poetry written in the Portuguese language. The system takes a poem, transcribes its words phonetically, discovers the poem's metrical pattern, fits all verses to the pattern and finds rhyming verses. Experiments with works from three different poets show that Aoidos achieves high accuracy and that it can be extended to other poets without difficulty. Although Aoidos has been developed with Brazilian Portuguese in mind, an experiment is presented where verses of a Portuguese poet are analyzed as well. The system was named after the Ancient Greek word ἀοιδός, which means "singer", "bard".

Aoidos is a rule-based system, a fact which provides several advantages. First, unknown words, unknown meters and unknown stress patterns are handled gracefully, since the system requires no training data. Second, the system keeps track of the evolution of a verse, from the raw text all the way to phonemes, which provides accountability

for the system's behavior. Finally, the system uses high-level, declarative rules that can be used to produce stylistic information.

With the help of a system like Aoidos, the automatic analysis of large poetry corpora becomes possible. This is an important step towards distant reading [1], where one would like to extract relevant features from verses in order to draw conclusions from looking at a large number of poems.

This article is organized as follows: Section 2 presents related work; Section 3 provides relevant background information concerning poetry written in Portuguese; Section 4 describes the Aoidos system in detail; Section 5 reports three experiments that intend to validate the system; finally, Section 6 contains discussion and conclusions.

2 Related work

There have been numerous proposals for the automatic analysis of several elements of poetry. Due to the unique features of each language and poetic tradition, such proposals are usually specific to one language, though more general approaches can be found, for example, for rhyme analysis [2].

For English poetry, a connectionist model [3] has been used early on to study differences in style among 10 poets. A program named AnalysePoems [4] has been presented that is capable of identifying the dominant meter of poems in English as well as locating rhymes. More recently, a tool called ZeuScansion [5] has been introduced that robustly discovers the global meter of poems in English and achieves a per-syllable accuracy of 87%. Algorithms have also been proposed for counting the number of syllables in English [6].

Proposals for analyzing several poetic elements exist in other languages too. Poetry written in Ancient Greek hexameter [7] has been subject to automatic scansion. There has been a proposal for the automatic analysis of Czech verse [8]. Several features of Classical Arabic poetry [9] have been used to extract poems from web pages. A system named SPARSAR [10] has been proposed to recognize both rhythm and rhyme in Italian and English poetry. Metrical annotation has been added to a corpus of Old Occitan poetry [11] by using an automatic system. Metricalizer [12] is a tool capable of analyzing the prosody, meter and rhyme of German poetry. A tool named Anamètre [13] has been proposed to extract the meter and rhymes of French verse. An automatic scansion system has been used to annotate a large corpus of Spanish poetry [14].

To the best of our knowledge, the only similar research on the Portuguese language has been conducted by Araújo and Mamede [15,16]. They proposed a system to classify various features of poems written in Portuguese. They use an external tool that provides the phonetic transcription of verses, but they do not attempt to fit verses to the meter; the examples they give show that the system does not find the correct number of syllables in verses where, e.g., a synaloepha is required. They did perform rhyme analysis, but they only conducted experiments to evaluate the system's response time, not its accuracy in any respect.

Research on speech synthesis has produced a number of systems that are related to the early stages of poetry analysis. A rule-based system for the phonetic transcription of European Portuguese has been proposed as early as 1992 [17]. A more complete

description of such a system has also been described [18] and work more focused on syllabification can also be found [19]. Statistical approaches to phonetic transcription have also been proposed [20].

The present article introduces and evaluates Aoidos, the first complete system for the automatic scansion of poetry written in the Portuguese language. As most approaches in the literature, Aoidos is rule-based, which allows for unsupervised poetry scansion.

3 Background

Portuguese is the official language of several countries, including Portugal and Brazil. This article is primarily concerned with poetry written in the context of Brazilian Literature. The system herein described was designed to process poetry written according to the orthography rules currently practiced in Brazil since the 1970s, although poetry from other times can also be processed as long as it is encoded in current orthography. In this article, the adjective *Portuguese* when applied to nouns such as poetry, verse, orthography, etc., refers to the Portuguese language, not to Portugal.

The current orthography of Portuguese is phonetic to a certain degree. In particular, the position of the stress in a word can be found deterministically in the vast majority of words. Syllabification can also be performed in most cases without errors. Phonetic transcription can be carried out with a certain accuracy, but there are cases where the precise pronunciation of a word cannot be deduced from orthography; such cases, however, do not interfere much, if at all, with the scansion of poetry.

Metrical poems written in the Portuguese language follow syllabic patterns. Scansion of a poem, therefore, requires determining the number of syllables its verses have and fitting every verse to the metric. Vowels are commonly put in contact with each other, forming groups of two, three or more; there are many possible ways of resolving such groups, so that an inexperienced person who begins to read a poem might face difficulties determining the number of syllables of the first verse in isolation. Even when the number of syllables is known, there is frequently more than one way to divide the verse into syllables. Stress patterns are also important in Portuguese poetry. For example, it is very common for verses containing 10 poetic syllables to be stressed on their 6th; such verses are called heroic decasyllables. This kind of stress pattern is important for the correct scansion of a verse.

The current practice in both Brazil and Portugal dictates that syllables in a verse are only counted up to and including the last stressed syllable. The remaining syllables are to be pronounced regularly, but they are not included in the total number of poetic syllables a verse has. Thus, a verse with 10 poetic syllables might have 10, 11 or 12 actual syllables — but usually not more, since the stress in Portuguese can only go back to as far as the third syllable from the end of a word. In general, starting from a conservative, unhurried pronunciation of a verse, one is looking to diminish the number of syllables in a verse by joining vowels or converting them into semivowels. Portuguese poetry also features the so-called broken verses. These are verses that are shorter than the full-length verses that appear in the same poem, but the number of poetic syllables in a broken verse is counted in the same fashion as in full verses.

4 Methods

This section describes Aoidos in detail. The first body of poems fully analyzed by Aoidos was the complete works of Brazilian poet Augusto dos Anjos (1884–1914), which comprises 284 poems and 6,580 verses. This choice was made because prior experience indicated that his poetry would provide rich material for analyzable poetic elements. All examples in this section are taken from his works, except those provided in the discussion about phonetic transcription. The latter ones were not taken from any particular source.

4.1 Overview

The complete scansion system is composed of several modules. Before going into the details of each one of them, this overview subsection attempts to provide a general idea of how the system works. An example of the evolution of a verse across all modules in the systems is given by Figure 1.

Input	<l>Com a cara hirta, tatuada de fuligens</l> *With his stiff face, tattooed with soot*
Pre-processing	com a cara hirta tatuada de fuligens
Transcription	/kõ/ /a/ /'ka.ra/ /'ir.ta/ /ta.tu'a.da/ /dɪ/ /fu'li.ʒẽs/
Naïve prosody	/kõ.a'ka**'rir**.ta.ta**'twa**.da.dɪ.fu'li.ʒẽs/
Rhythm	10/6
Informed prosody	/**kwa**'ka'rir.ta.ta'twa.da.dɪ.fu'li.ʒẽs/
Rhyme	AABCCB

Fig. 1. An example of a verse going through the pipeline of the modules that make up the scansion system. The two highlighted modules (rhythm and rhyme) take into consideration the poem as a whole. Highlighted syllables are the ones that changed from the previous step.

The system takes as input a Text Encoding Initiative (TEI) XML file. Each verse in that file is extracted and undergoes *pre-processing*, which involves regularizing the text, removing all punctuation and converting it to lower-case letters. In the next stage a broad *phonetic transcription* is produced for every word in each verse. Words are then joined and processed by a *prosody* module that applies rules ranging from simple ones like obligatory voicing of a word-final sibilant before a vowel or voiced consonant to more advanced rules such as crasis and synaloepha.

The prosody module is used twice. It is employed to perform a scansion based on a *naïve prosody*, one that has no access to information regarding the overall metric of the poem. The *rhythm* module then analyzes the result produced by naïve prosody for all verses in the poem and discovers the metric of the poem (10 syllables in Figure 1), as well as secondary stress patterns (the 6th should be stressed). Now scansion can be

performed again, this time with an *informed prosody* that knows the metrical patterns that must be obeyed and will do everything in its power to fit the verse to those patterns.

Finally, the *rhyme* module analyzes the final phonetic transcription of all verses in a poem and assigns them letters which correspond to the rhyme scheme. In the example, the verse being analyzed corresponds to the first "C" — the other "C" corresponding to the rhyming word in the next verse, *origens* "origins".

4.2 Pre-processing

Aoidos takes as input a file in the TEI XML format. TEI defines elements for many types of texts, including prose, drama and poetry. By using this format, the same source file can be used for Aoidos to perform its analyses and for a publishable HTML version to be generated by a stylesheet. Within TEI, the `lg` element is a stanza, the `l` element is a verse and `div` can be used for poems and poem sections.

The system looks for `l` elements within an `lg` and considers `lg` elements with the same parent as composing a poem. Furthermore, the input file can use the `choice` element to aid the system in two cases when phonetic transcription would produce wrong results or fail: foreign words, which are not written according to Portuguese orthography, and symbols, abbreviations, numbers, etc., which are not written out as words. Thus `choice` can be used to specify that the English word *clown* should be read as if it were written *cláun* in Portuguese, and that *D.* "Mrs." should be read as *dona*.

Once raw text has been produced for every `l` element, the system proceeds to resolve apostrophes. Portuguese orthography does not require apostrophes the way, for example, English does. Common contractions are written without them, as in *do* "of the" or *daquele* "of that". In poetry, however, the author or publisher may choose to indicate certain poetic contractions by using an apostrophe, as in *minh'alma* "my soul" instead of the full *minha alma*. Such apostrophe usage is neither required nor applied consistently, so that the system must be ready to perform such contractions on its own and to accept text that indicates the contractions by using apostrophes; the former is accomplished by using suitable phonetic rules in later stages, the latter by removing the apostrophe and joining words together.

Finally, punctuation is removed and letters are converted to lower case. It may be strange that punctuation is deleted at this stage, since it could convey information regarding the pronunciation of verses. For example, it could be that a full stop or an exclamation mark would strongly separate words and prevent them from being closely joined together phonetically. However, this is not at all the case with Portuguese poetry. The punctuation of verses carry important information, but words are to be scanned as if they did not exist. Consider this example:

> *Meu Deus! E este morcego! E, agora, vede:*
> /ˈmewˈdewˈzjes.tɪ.morˈse.gjaˈgɔ.raˈve.dɪ/
> "My God! And this bat! And, now, consider:"

Here, the vowel before the second exclamation mark and the two vowels that follow it must be coalesced into one syllable for the verse to be correctly scanned.

4.3 Phonetic transcription

The system's phonetic transcription is rule-based and is divided into three stages: location of the stress position, syllabification and mapping of letters to phonemes. The three stages are represented in Figure 2.

Original	com	a	cara	hirta	tatuada	de	fuligens
1. Stress position	com	a	c**a**ra	h**i**rta	tatu**a**da	de	ful**i**gens
2. Syllabification	com	a	c**a**-ra	h**i**r-ta	ta-tu-**a**-da	de	fu-l**i**-gens
3. Transcription	/kõ/	/a/	/'ka.ra/	/'ir.ta/	/ta.tu'a.da/	/dɪ/	/fu'li.ʒẽs/

With his stiff face, tattooed with soot

Fig. 2. The three stages of phonetic transcription.

Stress location, in this article, refers to the process of finding the letter in a written word that corresponds to the phonetic vowel that bears the stress. In Portuguese, the stress can be deterministically found in the vast majority of cases. The algorithm used by Aoidos has four rules for stress location. The first rule deals with exceptions in the writing system, that is, words that cannot be fully and correctly represented in the current official orthography; that's the case with *com* "with", which is unstressed but would otherwise be considered stressed, as *tom* "tone". The second rule deals with words with a stress-determining diacritic, as in *ácido* "acid", *âmbar* "amber", *maçã* "apple". The third rule finds the stress in words that end in certain consonants, in which case the stress falls on the vowel letter that precedes them, as in *aluguel* "rent", *comer* "to eat", *duplex* "duplex" and *acidez* "acidity". The fourth rule examines the final vowel letters and either finds the letter within that group that bears the stress, as in *torneio* "tournament", *uruguaio* "Uruguayan", *caju* "cashew", or finds a letter further back in the word, as in *imagem* "image", *ainda* "yet", *caieira* "lime kiln".

Syllabification proceeds by classifying letters into consonants, vowels and semivowels. In Portuguese, the main difficulty at this stage is to correctly determine the status of the letters *u* and *i*, which can be vowels or semivowels. The system first finds the consonants, then some vowels (e.g., *a* is always a vowel; *e* and *o* are in most positions vowels; the stressed letter in a word is always a vowel) and finally resolves groups of *u* and *i*. For each such group, the algorithm classifies letters by alternating between vowel and semivowel. The first letter in the group is a semivowel if the letter that precedes the group is a vowel; otherwise it is a vowel. Thus, the *i* in *migalha* "crumb" is a vowel, and the group in *buiuçu* "horse-eye bean" is a vowel, a semivowel and a vowel. Once letters are thus classified, syllabification becomes a matter of correctly splitting consonant groups. For most cases, splitting can be accomplished by keeping the first consonant in one syllable and placing the remaining ones in the next. Exceptions are groups containing one of *b*, *p*, *c*, *g*, *t*, *d*, *f* or *v*, followed by either *l* or *r*. Such groups are split immediately before these groups.

The final stage is mapping letters to phonemes. Most consonants can be directly mapped, such as *b*, *d*, *ç* and others; some consonants can be part of digraphs, as *lh* and

nh, which become, respectively, /ʎ/ and /ɲ/. Most vowels can also be mapped without further difficulties, as is the case with nasal vowels or the stressed vowels in proparoxytone words.

By far the two biggest sources of errors in the phonetic transcription module arise when determining whether a stressed *e* or *o* is open or closed (that is, whether they are /ɛ/ or /e/, /ɔ/ or /o/), and which consonant an *x* corresponds to. Both are difficult problems that cannot be gracefully handled by a rule-based system. For example, *olho* "eye" and *olho* "I look" are written exactly the same, but the former is pronounced /ˈo.ʎʊ/ and the latter /ˈɔ.ʎʊ/; *fixo* "fixed" is pronounced /ˈfik.sʊ/, but *lixo* "garbage" is pronounced /ˈli.ʃʊ/.

4.4 Rule engine

The rule engine is used by the prosody module in order to apply *rules* to *utterances*. In the context of this article, an utterance is a data structure that holds four main items:

- A string of phonemes, i.e., symbols from the International Phonetic Alphabet (IPA);
- Stress information on each phoneme, i.e., a boolean array specifying whether the corresponding phoneme is stressed or not;
- Syllabic information, such as boundaries and whether a given syllable marks the beginning or the end of a word (or both);
- A score, which is an integer that the system uses to attempt to keep track of how far-fetched the pronunciation of the utterance is.

Rules, in Aoidos, specify how to modify utterances. They are completely declarative, that is, they contain no code, only data. The prosody module works by applying rules to utterances, attempting to stick to the principle that the original utterance is pronounceable and that the modified utterance produced by them should also be.

A rule has a pattern, which the engine matches against the utterance string. If no match is found, the rule is not applied. The pattern is divided into three consecutive sections: the middle one specifies the phonemes or letters that are directly involved in the change, while the two surrounding sections provide context, as shown in Figure 3. A rule also specifies the total number of syllables (even if partial ones) that the pattern must span. This condition is important to make sure that the rule is not applied in situations which it was not designed for.

Original	/'ka.ra'ir.ta/	/ta.tu'a.da/
Matching	/'ka.[r]**[a]**'[i]r.ta/	/ta.[t]**[u]**'[a].da/
Replacing	/'ka'[r]**[]**[i]r.ta/	/ta'[t]**[w]**[a].da/
Final	/'ka'rir.ta/	/ta'twa.da/
	stiff face	*tattooed*

Fig. 3. Application of two rules, an elision and a synaeresis. The brackets indicate the portion of the string that was matched. The middle brackets, in bold, indicate the text that is changed; the other brackets define the context.

Rules may specify whether the syllables matched by the pattern must follow a certain scheme concerning word boundaries. For example, elision rules match two vowels, the first of which must be at the final syllable of a word and the second must be at the beginning of the next word. Rules may also indicate whether the syllables matched by the pattern must follow a stress scheme. For example, an /i/ after an /a/ can become the semivowel /j/, but only if it is unstressed — or, at any rate, a stressed /i/ becoming a /j/ is an entirely different matter. In Figure 3, it should be noted that, though the IPA stress mark changes position, internally the stressed vowel remains the same.

A rule specifies a replacement string, which is used when it has matched all conditions. The part of the original string that gets replaced is the middle section of the rule pattern; the other sections remain unchanged. The rule engine, as it applies the changes, adjusts the stress and syllabic information of the utterance.

Rules further contain a delta, that is, an integer that gets added to the utterance score if the rule is applied. Such scores are crucial in the prosody module, where many utterances are generated and only one of them will be chosen as the final version. Therefore, the delta is a measure of how unlikely or how drastic the change specified by the rule is. For example, elision is fairly common, and elision rules should therefore have a low delta; on the other hand, a rule that transforms the stressed /u/ in /ˈsu.a/ "his" into the semivowel /w/ should have a higher delta, since, although this is not a drastic change, other rules should be tried first.

4.5 Prosody

The prosody module takes the phonetic transcription of individual words, joins them and produces a series of possible utterances, each with its own score, by applying rules. The application of a rule produces a new utterance, which is kept alongside the original. The rules in this module are divided into two groups: basic and advanced. The difference between the two groups is that rules in the former use criteria concerning word boundaries and are only applied in the beginning of the process; rules in the latter group do not use word boundaries as criteria and might be allowed to combine indefinitely. For example, elision rules are located in the basic group, since such rules are based on word boundaries and should not be compounded; crasis rules, as performed by Aoidos, are applied regardless of word boundaries and may be compounded, and therefore are placed in the advanced group.

The prosody module is used twice during the scansion process: once before rhythm analysis, when the meter is not known (naïve prosody), and once afterwards, when such information is available (informed prosody). During the naïve prosody analysis, rules are applied to all matching instances in a verse at once. For example, an elision rule that matches two vowels in naïve prosody would elide both of them at once, while in informed prosody a total of *three* new derivations will be created from the original utterance: one with only the first elision applied, another with only the second elision, and a third with both. This ensures that the space of possible ways to read a verse is thoroughly explored in informed prosody. Another difference between naïve and informed prosody is that only rules with a low delta are applied during the naïve prosody analysis, while informed prosody is allowed access to the full rule set.

Naïve prosody, after applying all rules, chooses the utterance with the lowest score as its final result. Informed prosody is iterative: it begins by using rules with a low delta and rules with increasingly higher deltas are applied until at least one satisfactory utterance has been produced. A satisfactory utterance is one whose number of poetic syllables matches either the verse length or one of the secondary stresses prescribed by the rhythm analysis module. If more than one satisfactory utterance is found, then they are sorted according to an index based on their score and on an evaluation of how well they fit the stress pattern found by the rhythm analysis, and the best utterance is selected.

Basic rules The basic rules are responsible for joining words together in one unified utterance and taking care of phenomena which rely on word boundary information. The majority of rules in this module can be divided into:

- Elision rules. These rules remove the final vowel of a word when the next one begins with a vowel too; examples are *esfera opaca* "opaque sphere" and *escapava entre* "escaped between", which become /esˈfɛ.roˈpa.ka/ and /es.kaˈpaˈvẽ.trɪ/.
- Syncope rules, which remove certain unstressed phonemes from the interior of a word. Examples are *símbolo* "symbol" and *século* "century", which become /ˈsĩ.blʊ/ and /ˈsɛ.klʊ/.
- Apheresis rules. They remove an initial unstressed vowel when followed by certain consonant clusters, as in *espírito* "spirit" and *escaldante* "scalding", which become /ˈspi.ri.tʊ/ and /skawˈdã.tɪ/.

Each item in the list above talks about *rules*, in the plural, because a given phonetic phenomenon, such as elision, manifests itself in different contexts, which must be individually considered if a meaningful delta is to be assigned to each case.

Advanced rules The rules in this group deal with phenomena that take place regardless of word boundaries. The majority of the rules in this group are:

- Crasis rules. These rules mix two similar vowels into one, thus reducing the number of syllables in the verse by one. Examples are *vêem-se* "they are seen" and *a alma* "the soul", which from /ˈve.ẽ.sɪ/ and /aˈaw.ma/ are converted into /ˈvẽ.sɪ/ and /ˈaw.ma/.
- Synaloepha rules, which compress vowels and semivowels from different syllables into one syllable, forming diphthongs, triphthongs and, in more extreme cases, dropping phonemes. A simple example is *e o sol* "and the sun", which is converted from /ɪ.ʊˈsɔw/ to /jʊˈsɔw/. Such rules are also responsible for synaeresis, which is similar to synaloepha but happens inside a word, as in *afluem* "they flow", which is transformed from /aˈflu.ẽ/ into /aˈflwẽ/ by shifting the stress from one vowel to the other.

The system currently has 9 crasis rules and 20 synaloepha rules. Such a number is required to cover a great number of contexts while keeping each context as specific as

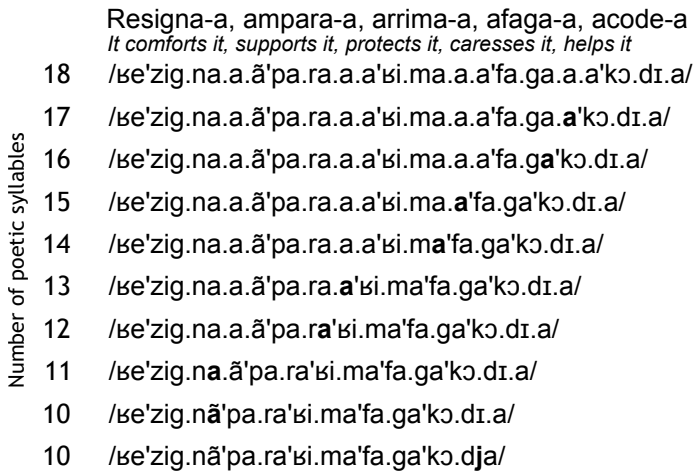

Fig. 4. Sequence of rules applied in the prosody module, reducing the number of poetic syllables from 18 to 10.

possible. When rules are too general, the risk is high that they will end up being applied in circumstances they were not planned for.

Figure 4 shows an example where crasis rules compound to greatly reduce the number of syllables in a verse. The steps shown in the figure were produced by recursively tracing back each utterance's origin. Because rules are applied to each match separately, there were many more possible combinations; this is just one of them.

4.6 Rhythm analysis

The rhythm analysis module is responsible for determining the number of poetic syllables that verses in a poem should have and the syllables that are commonly stressed in those verses. If the module fails to determine the number of poetic syllables for the poem as a whole, it resorts to a more flexible, stanza-based analysis.

The module takes the result of the prosody analysis and looks at the distribution of the number of poetic syllables among verses, as shown in Table 1. The poems in that table were chosen because they provide a good source of examples, but they are not necessarily typical of poet A. dos Anjos.

Finding the correct number of syllables in a poem can be problematic because, even in the context of metric poetry, not all of its verses may have the same number of poetic syllables. In particular, a common feature in Portuguese poetry are the so-called broken verses, which are shorter than the full verses in a given poem. The system, in such cases, finds the length of full verses and the informed prosody attempts to fit broken verses according to the commonly stressed syllables, which generally include the exact poetic length of the broken verses.

The algorithm that detects the full length of verses begins by trying to locate a pair of consecutive lengths whose presence in the verses of a poem add up to at least 75%. The

Table 1. Distribution (%) of the number of poetic syllables found by the prosody module in the verses of several poems. The bold values indicate the correct number of syllables in the full length verses.

Poem	6	7	8	9	10	11	12	13
Canto Íntimo	10.7	**85.7**	3.6					
Numa Forja	47.7	2.3			**45.3**	4.7		
Vênus Morta	3.1			12.5	**81.2**	3.1		
O Morcego				7.1	**78.6**	7.1	7.1	
Agonia de um Filósofo					**92.9**	7.1		
Asas de Corvo					**71.4**	21.4	7.1	
Guerra					**64.3**	28.6		7.1
Súplica num Túmulo						7.1	**85.7**	7.1

idea is that, although the naïve prosody may not generate very precise results, it will not be wrong by a wide margin, so that two adjacent lengths are usually enough to reach the threshold. Among poems in Table 1, this strategy finds the right answer (shown there in bold) in all cases but the second. The second poem contains a good amount of broken verses (exactly half of them), so that the lengths are distributed around the full length of the verse (10 syllables) and the broken length (6 syllables). The second poem is also an example of why picking the largest value does not work: there are more verses with length 6 than with length 10. It should be noted that the third poem in that table also contains broken verses, but in a much smaller proportion, so that a more sophisticated strategy is not required for analyzing that poem.

If there is no consecutive group of two lengths whose proportions add up to at least 75%, the system then considers the hypothesis that perhaps there are two main lengths instead of just one. In this case, the system looks for two groups of consecutive lengths. The combined percentage of each group should reach at least 37.5%, for a total combined percentage of 75%. If such groups are found, then the system considers that the full length of the verses in the poem is the largest length among the two groups. This strategy takes care of the case of the second poem in Table 1.

If the system fails to find the full length of verses using these two strategies, then it considers that there might not be a very discernible pattern at the poem level and that it should instead examine the poem stanza by stanza. Stanzas vary in length but they can be very short, as in two or three verses, so that a complex analysis of frequencies is not possible. The system, therefore, simply chooses the length with higher frequency. This was added to the system as a fall back strategy, since there were no cases in the works of A. dos Anjos that required it.

Once the full length of verses has been determined, the system looks for secondary stress patterns. The strategy here is straightforward: all lengths that are stressed in at least 60% of verses are considered to be secondary stresses. The final poetic syllable of broken verses is usually found among the secondary stresses.

4.7 Rhyme analysis

This module attempts to identify rhymes at the end of verses. An uppercase Latin letter starting from "A" is attributed to each verse; those verses that rhyme are assigned the same letter. An example is given in Figure 5.

Verse	Ending	Rhyming Segment	Result
Aí vem sujo, a coçar chagas plebeias,	ple'bɛj.as	ɛj.as	**A**
Trazendo no deserto das ideias	zi'dɛj.as	ɛj.as	**A**
O desespero endêmico do inferno,	dwĩ'fɛr.nʊ	ɛr.nʊ	**B**
Com a cara hirta, tatuada de fuligens	fu'li.ʒẽs	i.ʒẽs	**C**
Esse mineiro doido das origens,	zo'ri.ʒẽs	i.ʒẽs	**C**
Que se chama o Filósofo Moderno!	mo'dɛr.nʊ	ɛr.nʊ	**B**

There he comes, dirty, scratching plebeian wounds,
Dragging through the desert of ideas
Hell's endemic despair,
With his stiff face, tattooed with soot
This mad miner of origins,
Whom we call the Modern Philosopher!

Fig. 5. Rhyme analysis for a complete stanza. The ending displayed here is composed of the last three *phonetic* syllables.

The rhyme analysis algorithm extracts the final syllables from verses, starting from the last stressed syllable and going up to the very last phoneme. Consonants are then dropped from the beginning of the stressed syllable, leaving what is called, in this article, the rhyming segment. As can be seen in Figure 5, the rhyming segments can be compared to each other in order to establish the rhymes that exist between verses.

When the rhyming segments for each verse have been found, the algorithm assigns a letter to each verse. To do so, the algorithm keeps a dictionary of the 10 most recently seen rhyming elements and the letters originally attributed to them. If the current verse's rhyming element is found in the dictionary, it is assigned the corresponding letter; otherwise, the next available letter is chosen and a new rhyming element is added to the dictionary.

5 Experiments

This section describes three experiments, in decreasing order of significance. In each experiment, poems by different poets were evaluated. The first experiment evaluates the phonetic transcription, scansion and rhyme analysis produced by Aoidos; the second evaluates one component of scansion (number of poetic syllables); and the third provides evidence that Aoidos can be applied without difficulty to poems from other times and cultures.

5.1 Augusto dos Anjos

The complete works of Brazilian poet Augusto dos Anjos comprise 284 poems, among which 167 are sonnets (poems with four stanzas and fourteen verses). A total of 111 sonnets (1,554 verses) were randomly chosen and were subject to manual evaluation by experts. Each sonnet was independently assessed by two experts. The experts that participated in the experiment were not involved in the definition of the rules used by the prosody module. Figure 6 shows the output of Aoidos as seen in the evaluation screen. Experts had the option to toggle between phonetic transcription and regular text, and between highlighting or unhighlighting the stressed syllables. Experts could further add comments to each verse, explaining their decisions or providing comments.

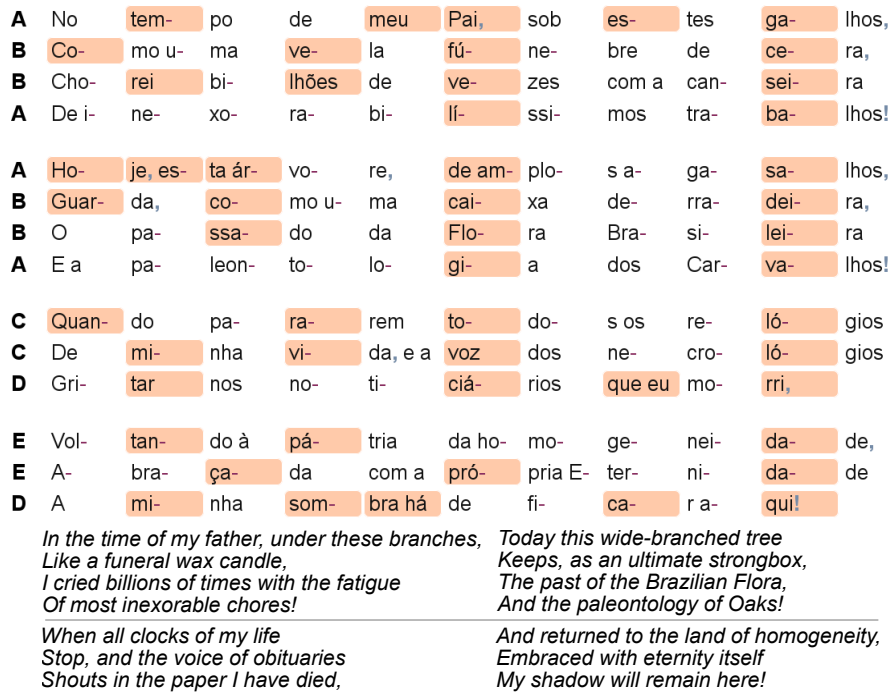

Fig. 6. Final result produced by Aoidos for A. dos Anjos' poem *Under the Tamarind*.

Experts were asked to evaluate three criteria for each verse:

- *Phonetic transcription:* is the phonetic transcription acceptable for the purposes of scansion?
- *Scansion:* is the final scansion performed by the system acceptable?
- *Rhyme analysis:* is the rhyme analysis provided by the system acceptable?

Because the definition of "acceptable" may vary from one expert to another, a criterion for a given verse was considered unacceptable if any of the two experts deemed it

so. Table 2 summarizes the results; all criteria were judged acceptable for at least 98% of verses.

Table 2. Results of the evaluation by experts of 111 poems automatically scanned by Aoidos.

	Total	Acceptable		Unacceptable	
Phonetic transcription	1,554	1,549	99.68%	5	0.32%
Scansion	1,554	1,523	98.01%	31	1.99%
Rhyme analysis	1,554	1,550	99.74%	4	0.26%

Scansion was found unacceptable in 31 cases, generally because the rules applied by the system produced a pronunciation that, though independently valid, was not the adequate choice. For example, in the following verse, a different scansion was indicated by the expert, who explained that it is better to keep *que* "that" in a syllable of its own in order to avoid two consecutive stressed syllables:

	1	2	3	4	5	6	7	8	9	10	11
Aoidos	Ven-	cen-	do o	a-	zul	que an-	te	si	s'er	gue	ra.
Expert	Ven-	cen-	do o a	zul	que	an-	te	si	s'er	gue	ra.

Overcoming the blue sky that had risen.

There were also cases when the experts were impressed by the results produced by the system. For example, there was a case when the same word was used in two adjacent verses, but in one verse the system produced a synaeresis within that word, and in the other verse it did not. Such behavior is required for the correct scansion of these two verses and is perceived as "intelligent". There were also cases when the experts commented that they reached the end of a verse only to find out that their scansion was not appropriate and that the system had produced the scansion that they should have used. This is a testimony to the fact that verses in Portuguese can be quite difficult to be scanned right the first time: an early decision to make a synaloepha or not might cause an extra or missing syllable once the reader has finished the verse.

5.2 Gustavo Teixeira

Two books by Brazilian poet Gustavo Teixeira (1881–1937), *Notebook* and *Lyric Poems*, were read by an expert, who classified poems and verses according to their number of poetic of syllables. The two books contain a total of 112 poems and 3,178 verses. The automatic scansion system was applied to these poems and the final scansions were compared to those indicated by the expert; results are shown in Table 3.

The system found the correct number of poetic syllables in 99.8% of cases. It mostly failed to correctly scan broken verses, which are more prominent in the poetry of G. Teixeira than in that of A. dos Anjos.

Table 3. Proportion of correctly identified lengths of verses in poems by G. Teixeira.

Full length	Actual length	Count	Correct	
12	12	1,523	1,523	100.0%
12	8	10	8	80.0%
12	6	17	17	100.0%
10	10	1,031	1,031	100.0%
10	6	30	29	96.7%
9	9	42	42	100.0%
9	4	14	14	100.0%
8	8	172	172	100.0%
7	7	303	303	100.0%
7	4	36	32	88.9%
		3,178	3,171	99.8%

5.3 Luís de Camões

In order to test the system on a more diverse body of poetry, the automatic scansion system was applied to *The Lusiads*, an epic poem written by Portuguese poet Luís de Camões (1524–1580), which comprises 10 cantos and a total of 8,816 verses. A total of 8 new rules had to be added to the system so that all verses could be scanned. These new rules were mostly synaloephas not found in the works of A. dos Anjos, as in the following verse:

> Respond**e ao em**baixador, que tanto estima:
> *He replies to the ambassador, whom he highly cherishes:*

The system correctly identified that all verses contain 10 poetic syllables, and for all cantos the 6th syllable was correctly considered a secondary stress. This needs not be manually evaluated: it is known that all verses in the *The Lusiads* are decasyllables and that most of them are heroic, that is, have their 6th syllable stressed. The results produced by the system indicate that the sixth syllable was stressed in 97,03% of verses. The system was able to process the poetry of L. de Camões significantly faster than that of either A. dos Anjos and G. Teixeira. The main reason is that the number of pronunciation alternatives considered by the system before settling on its final version was larger for the two latter poets. Whereas the system considered an average of 35.79 alternatives for each verse in the poetry of A. dos Anjos and 26.03 for G. Teixeira, the average for L. de Camões was only 9.30. This is certainly a reflection of stylistic features.

6 Discussion and Conclusions

This article introduced Aoidos, a rule-based system that extracts information regarding scansion and rhyming from poetry written in Portuguese. Its results can be used for several purposes. The system can be employed to process poetic corpora that are too large to be analyzed by humans; at a rate of about 40 verses per second in an ordinary computer, processing even millions of verses becomes feasible. The analysis of verses

and poems can be used to understand the style of different poets and times; and not only the final, scanned verse can be used for such comparisons, but also the very set of phonetic rules employed during the analysis. It might be that certain poets favor certain types of synaloephas, or that they use them more sparingly. Finally, as a robotic minstrel, Aoidos does not get emotionally involved when scanning poetry and therefore spots mistakes in the source material that render verses impossible to fit the metric; indeed, quite a few mistakes have been found in our digital editions thanks to Aoidos.

The system was evaluated by experiments, which show that it is capable of scanning verses with great accuracy (above 98%). Aoidos applies phonetic rules to verses and attempts to produce a pronounceable, metrified transcription of poems. Although its rules were designed initially for a 20th-century Brazilian poet, the system proved capable of analyzing poems by another poet from the same place and time without modifications as well as an 8,816-verse poem written three centuries earlier by a Portuguese poet, this time with the addition of only 8 new rules.

Aoidos can still be improved in several aspects. Its current treatment of broken verses, an important feature of Portuguese poetry, is not yet very robust, as evidenced by the second experiment. Its understanding of stress patterns, i.e., rhythm, works well for most cases, but the few mistakes it does commit are evidence that it can be improved. In particular, it currently has no conception of a rhythmic stress, so that all stress information used in its analyses is derived from each word's primary stress. Its pre-processing stage currently does not handle foreign words or abbreviations (unless annotations are provided), which would certainly cause mistakes to be made in larger, untreated corpora.

The possibility of extending Aoidos to support other languages is currently being investigated. One important challenge to take into account when considering other languages is obtaining a suitable phonetic transcription from the source text: whereas the orthography of Portuguese is fairly phonetic, this is not the case with, e.g., English (where the same letter can correspond to different phonemes) and Russian (where stress position cannot be determined from the written text). Preliminary results of applying Aoidos to poetry written in Spanish are encouraging.

Acknowledgments

We would like to thank Isabela M. B. Sandoval, Lívia Guimarães and Samanta Maia for their contribution in the experiments. This work was partially supported by the Brazilian National Council for Scientific and Technological Development (CNPq) and by the Santa Catarina Foundation for the Support of Research and Innovation (FAPESC).

References

1. Moretti, F.: Distant reading. Verso Books (2013)

2. Reddy, S., Knight, K.: Unsupervised discovery of rhyme schemes. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2, Association for Computational Linguistics (2011) 77–82
3. Hayward, M.: Analysis of a corpus of poetry by a connectionist model of poetic meter. Poetics **24** (1996) 1–11
4. Plamondon, M.R.: Virtual verse analysis: Analysing patterns in poetry. Literary and Linguistic Computing **21** (2006) 127–141
5. Agirrezabal, M., Arrieta, B., Astigarraga, A., Hulden, M.: ZeuScansion: a tool for scansion of English poetry. In: Proceedings of the 11th International Conference on Finite State Methods and Natural Language Processing, St Andrews, Scotland (2013) 18–24
6. Hammond, M.: Calculating syllable count automatically from fixed-meter poetry in English and Welsh. Literary and Linguistic Computing **29** (2014) 218–233
7. Papakitsos, E.C.: Computerized scansion of Ancient Greek hexameter. Literary and Linguistic Computing (2010)
8. Ibrahim, R., Plecháč, P.: Towards the automatic analysis of Czech verse. Formal Methods in Poetics, Lüdenscheid, RAM (2011) 295–305
9. Almuhareb, A., Alkharashi, I., AL Saud, L., Altuwaijri, H.: Recognition of Classical Arabic poems. In: Proceedings of the Workshop on Computational Linguistics for Literature, Atlanta, Georgia, Association for Computational Linguistics (2013) 9–16
10. Delmonte, R.: A computational approach to poetic structure, rhythm and rhyme. In: Proceedings of the First Italian Conference on Computational Linguistics CLiC-it 2014 & and of the Fourth International Workshop EVALITA 2014, Pisa University Press (2014) 144–150
11. Rainsford, T., Scrivner, O.: Metrical annotation for a verse treebank. In: The 13th International Workshop on Treebanks and Linguistic Theories (TLT13). (2014) 149–159
12. Bobenhausen, K., Hammerich, B.: Métrique littéraire, métrique linguistique et métrique algorithmique de l'allemand mises en jeu dans le programme metricalizer. Langages **3** (2015) 67–88
13. Éliane Delente, Renault, R.: Traitement automatique des formes métriques des textes versifiés. In: 22ème Traitement Automatique des Langues Naturelles. (2015)
14. Navarro, B., Ribes-Lafoz, M., Sánchez, N.: Metrical annotation of a large corpus of Spanish sonnets: representation, scansion and evaluation. In: Language Resources and Evaluation Conference. (2016)
15. de Araújo, P.A.M.: Classificação de poemas e sugestão das palavras finais dos versos. Universidade Técnica de Lisboa (2004)
16. de Araújo, P.A.M., Mamede, N.J.: Classificador de poemas. In: Conferência Científica e Tecnológica em Engenharia. (2002)
17. Oliviera, L.C., Viana, M., Trancoso, I.M.: A rule-based text-to-speech system for Portuguese. In: Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on. Volume 2., IEEE (1992) 73–76
18. Braga, D., Coelho, L., Resende Jr, F.G.V.: A rule-based grapheme-to-phone converter for TTS systems in European Portuguese. In: Telecommunications Symposium, 2006 International, IEEE (2006) 328–333
19. Neto, N., Rocha, W., Sousa, G.: An open-source rule-based syllabification tool for brazilian portuguese. Journal of the Brazilian Computer Society **21** (2015) 1–10
20. Couto, I., Neto, N., Tadaiesky, V., Klautau, A., Maia, R.: An open source HMM-based text-to-speech system for Brazilian Portuguese. In: 7th International Telecommunications Symposium. (2010)