Universidade de Coimbra
Faculdade de Ciências e Tecnologia
Departamento de Engenharia Electrotécnica e de Computadores

Rui Pedro de Azevedo Venâncio

Geração de Pseudopalavras para Avaliação Linguística

Coimbra, Fevereiro 2018

UNIVERSIDADE DE COIMBRA

U C

Geração de Pseudopalavras para Avaliação Linguística

Orientador:

Prof. Doutor Fernando Manuel dos Santos Perdigão

Co-Orientador:

Jorge Daniel Leonardo Proença

Júri:

Prof. Doutora Teresa Martinez dos Santos Gomes

Prof. Doutora Carla Alexandra Calado Lopes

Prof. Doutor Fernando Manuel dos Santos Perdigão

Coimbra, Fevereiro 2018

Agradecimentos

Quero agradecer, em primeiro lugar, ao Prof. Doutor Fernando Perdigão, pelos conhecimentos transmitidos, o apoio incansável e o empenho total para o sucesso deste projeto.

Ao Jorge Proença pela disponibilidade, ajuda e à-vontade que demonstrou durante a realização deste trabalho.

A toda a minha família, em especial, ao meu pai, mãe e irmã, aos meus tios e aos meus avós, pela educação que me deram e por todo o apoio prestado, ao longo de todos estes anos.

Gostaria de agradecer também aos eternos LedZener, pelas memórias e aventuras nesta cidade.

A todos os meus amigos de infância e aos que conheci nesta cidade, também foram fulcrais direta ou indiretamente.

À memória dos meus avós paternos, por tudo o que me ensinaram e ajudaram, nunca serão esquecidos.

Resumo

A capacidade de leitura é um aspeto importante durante a aprendizagem da língua e é adquirida, geralmente, em crianças com idade escolar. A avaliação do desempenho da leitura pode ser aferida através de diferentes formas, tanto na leitura de palavras como na leitura de pseudopalavras.

Pseudopalavras são palavras que não existem no léxico, mas que são pronunciáveis, uma vez que seguem as regras fonotáticas de uma determinada língua.

A leitura de pseudopalavras permite avaliar se as regras de conversão de texto para fala (consciência fonológica) estão bem assimiladas, já que o leitor não tem familiaridade com as pseudopalavras que está a ler. Assim é possível avaliar o desempenho na leitura, de modo a, por exemplo, prevenir futuros défices fonológicos. Assim, é importante a criação de um sistema que seja capaz de gerar pseudopalavras, segundo determinados critérios e especificações da língua, porque até ao momento não existe nenhum gerador de pseudopalavras, em Portuguêss Europeu.

Este trabalho aborda o problema da geração de pseudopalavras, propondo algoritmos para a sua concretização. Os algoritmos são baseados em concatenação de sílabas, com a garantia de que todos os pares de sílabas, que formarão as pseudopalavras, são encontros silábicos encontrados no léxico. A frequência de ocorrência desses pares de sílabas, como ínicio, meio e fim de palavra, será crucial para a formação de pseudopalavras, pois os pares de sílabas tenderão a aparecer com mais frequência, consoante o seu número de ocorrência nas diferentes posições das pseudopalavras.

Este projeto também pressupõe a criação de um corpus lexical e um *software* fácil de utilizar e capaz de mostrar as pseudopalavras geradas e medidas adicionais, relacionadas com proximidade lexical. Os algoritmos e o consequente interface com o utilizador foram desenvolvidos em *MATLAB*.

Abstract

Reading ability plays an important role during the process of learning any language and is acquired in children, generally, in elementary school. The evaluation of reading performance can be done by reading words or pseudowords.

Pseudowords are words that respect the phonotactic restrictions of a language and can be read, but don't exist in lexicon.

When reading pseudowords it's possible to evaluate if the rules of conversion from text to speech (phonological awareness) are well assimilated, since the reader doesn't have any kind of familiarity with it. Thus it is possible to evaluate the reading performance in order to, for example, prevent future phonological deficits. So it is important to have a system that can be able to generate pseudowords, according to certain criteria and specifications, because there is none generator, in European Portuguese, at the moment.

This thesis describes the process of generating pseudowords and proposes algorithms for this task. The algorithms are based on concatenation of syllables, with the condition that all pairs of syllables, that will form the pseudowords, were found in the lexicon. The frequency of occurrence of the pairs of syllables, in the beginning, middle and end of words from lexicon, will have an important role in the formation of pseudowords. It means that more frequent pairs of syllables will tend to appear, more frequently, in the different positions of the pseudowords.

This project also presupposes the creation of a lexical corpora and an easy-to-use software capable of showing in a table the generated pseudowords and other metrics related to lexical proximity. The algorithms and the user interface were developed in MATLAB.

Índice

Lista de Figuras

Lista de Tabelas

Lista de Acrónimos

OLD20	Ortographic Levenshtein Distance 20
pt-PT	Português Europeu
Npp	número de pseudopalavras
PP	pseudopalavras
Nsil	número de sílabas
Dist1sub	vizinhos de distância 1 só por substituição
Dist1	vizinhos de distância 1
Dist2	vizinhos de distância 2
Dist3	vizinhos de distância 3
MEX	MATLAB executable
Dists	vizinhos de diferentes distâncias
Lists	lista dos 20 vizinhos mais próximos
HTML	HyperText Markup Language

Capítulo 1

Introdução

As crianças, antes de entrarem para o primeiro ciclo, já são capazes de diferenciar e manipular sílabas; no entanto a sua consciência fonológica só é melhorada com a entrada no 1º ciclo [9]. Contudo, a rima e a consciência de palavras são níveis da consciência fonológica que as crianças têm que ter adquirido antes da entrada para o ensino básico [4], para fazer a associação grafema-fonema (leitura) e fonema-grafema (escrita). [10]

As crianças aprendem os valores fonológicos das letras, ou seja, os sons que as mesmas representam, uma vez que a letra pode ter diferentes sons, ajudando deste modo na identificação das letras e na leitura das palavras. É através do som das palavras que a criança aprende a identificar a semelhança e a diferença entre as mesmas.

Esta dissertação é muito pertinente, já que com a criação de pseudopalavras, seguindo as regras fonotáticas da língua portuguesa, é possível que um professor as possa usar para avaliar o desempenho das crianças através da leitura das mesmas.

Pseudopalavras são palavras que não existem no léxico da língua e não têm qualquer significado, mas são pronunciáveis mais ou menos sem ambiguidade, segundo as regras fonotáticas.

O propósito desta dissertação é ter inicialmente um corpus lexical suficientemente grande e tratado, de maneira a que posteriormente seja usado na criação de pseudopalavras através de um programa eficiente e facilmente utilizável, que gera pseudopalavras e que apresenta cálculos lexicais e/ou métricas das mesmas, consoante o interesse do utilizador.

Este projeto vem preencher um vazio, em relação ao Português Europeu, na medida em que existe a necessidade de ferramentas que permitam a geração de pseudopalavras, já que neste momento só existem (e poucos) sistemas geradores de pseudopalavras noutras línguas, que não o Português Europeu.

1.1 Motivação

Uma das motivações para este projeto provém da implementação das Metas Curriculares de Português do Ensino Básico, que definem diferentes objetivos para diferentes anos de

escolaridade, sendo um deles a avaliação da leitura, nomeadamente, a leitura de pseudo-palavras.

Outra motivação deste projeto recaiu sobre a necessidade de um sistema gerador de pseudopalavras em Português Europeu (pt-PT) facilmente utilizável e adaptado à nossa língua materna, de modo a que qualquer investigador, professor ou utilizador comum, possa utilizar pseudopalavras para o seu estudo.

Até à data da dissertação não existe nenhum gerador de pseudopalavras a nível do Português Europeu e muito poucos a nível mundial, provavelmente devido à sua elevada complexidade.

1.2 Objetivos

Tendo sempre em mente a motivação deste trabalho, os objetivos passaram pela criação de um corpus lexical e de algoritmos para flexão de palavras, divisão silábica, pronunciação, e extração de características de proximidade lexical e fonológica.

O objetivo principal desta dissertação é a criação de um sistema gerador de pseudopalavras em Português Europeu, fácil de utilizar, com diferentes especificações e diferentes métodos de geração através da combinação de sílabas, tendo em conta a frequência de ocorrência de cada par de sílabas na língua.

1.3 Estrutura da dissertação

Esta dissertação está dividida em seis capítulos. O capítulo 1 dá a conhecer o trabalho desenvolvido e as diferentes áreas que esta dissertação abrange. Indica também quais os objetivos e a motivação para o desenvolvimento da mesma.

O capítulo 2 descreve os sistemas existentes para a geração de pseudopalavras, com uma breve descrição dos mesmos e que estudos existem até ao momento da realização desta dissertação, o que equivale ao estado da arte deste assunto.

No capítulo 3 é descrita a obtenção do *corpus* lexical e das formas e/ou técnicas que foram utilizadas para a extração, verificação e validação do mesmo.

O capítulo 4 aborda, detalhadamente, como funcionam os algoritmos geradores de pseudopalavras e métricas lexicais.

No capítulo 5 são visualizados os resultados obtidos com os algoritmos, em termos de pseudopalavras, tempos de processamento, entre outras coisas.

Por fim no capítulo 6 temos as conclusões da dissertação, indicando os possíveis melhoramentos.

Capítulo 2

Trabalhos relacionados

Neste capítulo será abordado, primeiramente, um estudo que analisa o formato das pseudopalavras e a sua importância para prevenir défices fonológicos e de seguida os programas e sistemas de desenvolvimento que foram explorados na realização desta dissertação.

2.1 Estudo sobre Pseudopalavras

Um primeiro trabalho que foi estudado no âmbito desta dissertação foi a dissertação de mestrado em Ciências da Linguagem [1]. Este trabalho demonstra que "as dificuldades no processamento fonológico em crianças com dislexia tendem a ser reproduzidas de forma mais consistente em provas com pseudopalavras linguisticamente motivadas, pelo facto do processamento fonológico de crianças disléxicas se encontrar perturbado e ser melhor avaliado por provas com pseudopalavras".

O estudo analisou os resultados de quatro provas onde foram usadas pseudopalavras linguisticamente motivadas: "uma prova de discriminação auditiva, uma prova de leitura e duas provas de repetição (uma com pseudopalavras fonologicamente motivadas e outra com pseudopalavras morfologicamente motivadas)". Os resultados destes testes foram bastante positivos e significativos, tendo-se verificado uma "correlação positiva entre o desempenho dos sujeitos e o grau de índice de probabilidade fonológica (IPF), [1], associado às pseudopalavras". Também é verificado que "quanto maior o (IPF) maior o número de respostas corretas e melhor o desempenho dos pacientes. Tal, valida, a utilização deste tipo de instrumento no diagnóstico desta patologia."

Este indicador é baseado em probabilidades de fonemas e não foi usado no presente trabalho uma vez que apenas são usados grafemas nas definições das sílabas. Contudo, a vantagem de se usarem PP em estudos de avaliação linguística é evidenciada. De seguida vão ser abordados programas geradores de PP de domínio público.

2.2 Geradores de Pseudopalavras

2.2.1 Wuggy

O *Wuggy*, [6], é um *software* gerador de pseudopalavras que veio melhorar os métodos existentes, até então (2010). O *Wuggy* está disponível na página `http://crr.ugent.be/programs-data/wuggy`, de modo a ser descarregado, pois necessita de ser instalado. Permite a geração de pseudopalavras polissilábicas que obedeçam às restrições fonotáticas de diferentes línguas. O programa está disponível, em holandês, inglês, alemão, francês, espanhol, sérvio, basco e vietnamita, com a possibilidade de ser expandido para outras línguas, [6]. Funciona com base num dicionário de palavras divididas em sílabas. Este programa foi desenvolvido em *Python* e tem como janela inicial a figura seguinte:

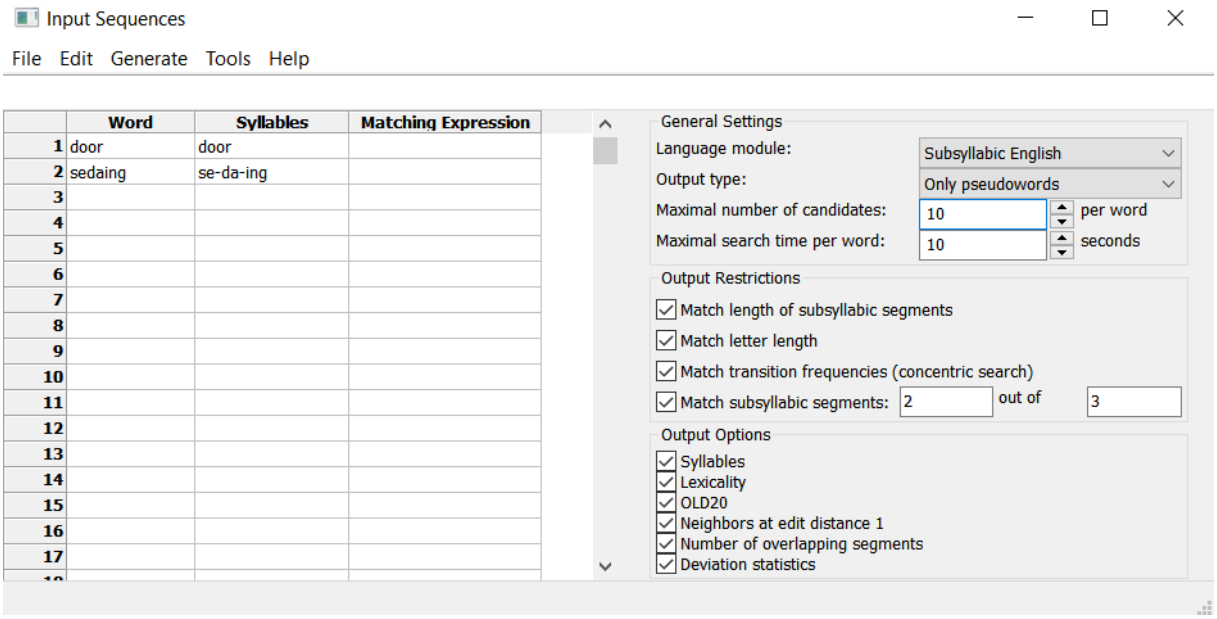

Figura 2.1: Janela do *Wuggy*.

Em termos de especificações gerais, o utilizador escolhe a língua das palavras e que tipo de palavras quer, pois é permitida a geração de palavras do léxico, pseudopalavras ou ambos. Há a possibilidade de escolha do número de palavras/pseudopalavras a gerar. Por predefinição é de 10 candidatos por palavra introduzida. O tempo de procura máximo por parte do algoritmo por palavra, por predefinição é de 10 segundos.

Depois de escolhidas as especificações gerais, é necessário a introdução de uma palavra (pelo menos) na 1ª coluna da tabela, da figura 2.1, na 2ª coluna se a palavra existir no léxico, faz a sua divisão silábica, automaticamente, caso contrário é necessário a introdução da mesma. A 3ª coluna permite resultados parecidos a uma expressão regular. A geração também pode ser feita através da leitura a partir de um ficheiro ou através da introdução de uma pseudopalavra, com a respetiva introdução da sua divisão silábica.

Em termos de restrições o Wuggy apresenta as seguintes:

- "*Match length of subsyllabic segments*": escolhendo esta opção, as palavras geradas vão apresentar a mesma estrutura silábica da(s) palavra(s) introduzida, ou seja, se a(s) palavra(s) introduzidas tiverem uma estrutura sílabica, por exemplo de, consoante,vogal-consoante,vogal-consoante, à saída obtém-se palavras com a mesma estrutura silábica.

- "*Match letter length*": com esta opção, são obtidas correspondências com o mesmo número de letras da(s) palavra(s) introduzidas.

- "*Match transition frequencies (concentric search)*": esta opção num primeiro caso garante candidatos com a mesma transição de frequências, caso não seja possível, o máximo desvio possível de transição de frequências aumenta em potências de 2 (± 2, ± 4, etc.).

- "*Match subsyllabic segments*": corresponde ao quão parecidas as palavras serão em comparação às palavras originais introduzidas. Por predefinição esta relação é 2 em 3 (2/3), esta relação dá origem a palavras semelhantes à palavra original introduzida. Se o valor for 3/3, por exemplo, quer dizer que a única palavra possível de ser retornada é a própria palavra.

No *Wuggy* podemos saber informações lexicais acerca das palavras/pseudopalavras, através das opções seguintes.

- "*Syllables*": esta opção faz com que as palavras geradas estejam divididas em sílabas (divisão silábica).

- "*Lexicality*": esta opção permite saber se um resultado é palavra (w) do inglês *word* ou pseudopalavra (n) do inglês *nonword*.

- "Ortographic Levenshtein Distance 20 (OLD20)": a média da distância de *Levenshtein* dos 20 vizinhos/candidatos mais próximos de uma dada palavra/pseudopalavra. A distância de *Levenshtein* é dada pelo número de operações (substituições, apagamentos e inserções) mínimas para transformar uma *string* noutra. Escolhendo esta opção vai tornar a geração de palavras mais lentas, já que, é necessário calcular a distância de *Levenshtein* entre a(s) palavra(s) candidata e as vinte palavras mais semelhantes no léxico. Se o valor do OLD20 for baixo, quer dizer que existem palavras no léxico que são obtidas alterando apenas 1 ou mais letras. O OLD20_diff obtém a diferença entre a palavra gerada e a palavra escolhida.

- "Neighbors at edit distance 1"(Ned1): número de palavras do léxico que se podem obter a partir de cada da(s) palavra(s) candidata(s) substituindo, apagando ou inserindo uma letra. Esta opção também diminui a rapidez do sistema.

- "*Number of overlapping segments*": indica o rácio de parecença dos resultados (palavras/pseudopalavras) com a palavra original. Se o parâmetro *Match subsyllabic segments* estiver selecionado nas restrições, o valor deste parâmetro será o mesmo, por predefinição no *Match subsyllabic segments* é 2/3.

- "*Deviation Statistics*": que mostra a maior diferença nas frequências de transição entre as sílabas na sequência gerada e na sequência original. Se por exemplo, o valor for 14, quer dizer que existem mais 14 palavras com a mesma transição. Esta opção também mostra a soma de todas as variações de frequência de transição (valores absolutos). E por fim também tem uma coluna que mostra a zona da palavra, onde existe o maior desvio de transição.

Resultados possíveis para a introdução da palavra "door" e da pseudopalavra "sedaing", com as restrições por predefinição e com as informações lexicais todas selecionadas, apresenta os resultados seguintes:

Results

	Word	Match	Lexicality	Old20	Old20_Diff	Ned1	Ned1_Diff	Overlap_Ratio	Maxdeviation	Summed_Deviation	Maxdeviation_Transition
1	door	woor	N	1.5	0.0	10	2	2/3	-3	6	[_w]oor_
2	door	dier	N	1.3	-0.2	14	6	2/3	-3	5	_d[ier]_
3	door	deir	N	1.8	0.3	4	-4	2/3	-3	5	_d[eir]_
4	door	doos	N	1.1	-0.4	18	10	2/3	13	14	_doo[s_]
5	door	foor	N	1.5	0.0	10	2	2/3	12	14	[_f]oor_
6	door	doob	N	1.85	0.35	3	-5	2/3	-25	30	_doo[b_]
7	door	doon	N	1.45	-0.05	11	3	2/3	31	35	_doo[n_]
8	door	hoor	N	1.45	-0.05	11	3	2/3	61	70	[_h]oor_
9	door	roor	N	1.45	-0.05	11	3	2/3	43	53	[_r]oor_
10	door	dood	N	1.6	0.1	8	0	2/3	64	69	_doo[d_]
11	se-da-ing	be-co-ing	N	1.95	0.0	1	0	2/3	-376	879	_bec[o]ing_
12	se-da-ing	be-vo-ing	N	2.0	0.05	0	-1	2/3	-376	987	_bev[o]ing_
13	se-da-ing	be-ro-ing	N	2.0	0.05	0	-1	2/3	-376	867	_ber[o]ing_
14	se-da-ing	be-wo-ing	N	2.0	0.05	0	-1	2/3	-376	951	_bew[o]ing_
15	se-da-ing	be-zo-ing	N	2.0	0.05	0	-1	2/3	-376	986	_bez[o]ing_
16	se-da-ing	be-lo-ing	N	1.85	-0.1	3	2	2/3	-376	867	_bel[o]ing_
17	se-da-ing	be-so-ing	N	1.95	0.0	1	0	2/3	-376	754	_bes[o]ing_
18	se-da-ing	be-fo-ing	N	2.0	0.05	0	-1	2/3	-376	834	_bef[o]ing_
19	se-da-ing	be-mo-ing	N	2.0	0.05	0	-1	2/3	-376	840	_bem[o]ing_
20	se-da-ing	be-ho-ing	N	1.95	0.0	1	0	2/3	-376	903	_beh[o]ing_

Figura 2.2: Resultados para a introdução da palavra "door" e da pseudopalavra "sedaing".

Durante a realização deste trabalho foi possível colocar o *Wuggy* a gerar pseudopalavras em português, através da introdução de um ficheiro de léxico de antigos projetos, em Português Europeu, com palavras de léxico, divisão silábica e frequência por milhão. E num segundo ficheiro bastou a introdução de letras acentuadas, letras que podiam ser seguidas de outras e letras duplamente acentuadas, em português, em *Python*.

Possíveis resultados em pt-PT com a introdução das palavras "espátula"e "carruagem", sem alterar os valores predefinidos no *Wuggy*, mantendo as restrições e as informações lexicais todas selecionadas, os resultados foram os seguintes:

Results

	Word	Match	Lexicality	Old20	Old20_Diff	Ned1	Ned1_Diff	Overlap_Ratio	Maxdeviation	Summed_Deviation	Maxdeviation_Transition
1	es-pá-tu-la	es-mí-rá-la	N	3.85	0.65	0	0	2/3	-206	737	_esmír[á]la_
2	es-pá-tu-la	es-mí-ru-pa	N	3.95	0.75	0	0	2/3	-194	865	_e[sm]írupa_
3	es-pá-tu-la	es-mí-ru-za	N	4.0	0.8	0	0	2/3	-194	609	_e[sm]íruza_
4	es-pá-tu-la	es-mí-ru-lo	N	3.6	0.4	0	0	2/3	-194	792	_e[sm]írulo_
5	es-pá-tu-la	es-mí-ru-ma	N	3.8	0.6	0	0	2/3	-194	694	_e[sm]íruma_
6	es-pá-tu-la	es-mí-ru-fa	N	3.95	0.75	0	0	2/3	-204	890	_esmíru[f]a_
7	es-pá-tu-la	es-mí-ru-sa	N	3.9	0.7	0	0	2/3	-194	734	_e[sm]írusa_
8	es-pá-tu-la	es-mí-ru-ja	N	4.0	0.8	0	0	2/3	-226	913	_esmíru[j]a_
9	es-pá-tu-la	es-mí-ru-ga	N	3.8	0.6	0	0	2/3	-194	715	_e[sm]íruga_
10	es-pá-tu-la	es-mí-ru-xa	N	4.0	0.8	0	0	2/3	-226	916	_esmíru[x]a_
11	ca-rru-a-gem	da-flo-a-pem	N	4.4	1.4	0	0	2/3	-256	896	_[da]floapem_
12	ca-rru-a-gem	da-flo-a-zem	N	4.3	1.3	0	0	2/3	-256	781	_[da]floazem_
13	ca-rru-a-gem	da-flo-a-lem	N	4.4	1.4	0	0	2/3	-256	841	_[da]floalem_
14	ca-rru-a-gem	da-flo-a-mem	N	4.3	1.3	0	0	2/3	-256	916	_[da]floamem_
15	ca-rru-a-gem	da-flo-a-fem	N	4.45	1.45	0	0	2/3	-256	919	_[da]floafem_
16	ca-rru-a-gem	da-flo-a-sem	N	4.3	1.3	0	0	2/3	-256	1054	_[da]floasem_
17	ca-rru-a-gem	da-flo-a-bem	N	4.4	1.4	0	0	2/3	-256	937	_[da]floabem_
18	ca-rru-a-gem	da-gro-a-pem	N	4.0	1.0	0	0	2/3	-256	957	_[da]groapem_
19	ca-rru-a-gem	da-gro-a-zem	N	4.0	1.0	0	0	2/3	-256	842	_[da]groazem_
20	ca-rru-a-gem	da-gro-a-lem	N	4.0	1.0	0	0	2/3	-256	902	_[da]groalem_

Figura 2.3: Resultados para a introdução das palavras “espátula” e “carruagem”.

Verifica-se, através destes resultados, encontros silábicos que não foram encontrados em nenhuma palavra do léxico, caso de por exemplo, “da-flo”. Verifica-se também, através da primeira pseudopalavra “es-mí-rá-la”, que é uma palavra com dupla acentuação, o que não é algo desejável pois na língua só acontecem determinados casos particulares casos, de por exemplo, “órgão” e “sótão”.

Observou-se, independentemente da língua escolhida que se mantivermos as mesmas especificações, que os resultados são determinísticos, ou seja a geração retornará sempre os mesmos resultados. Muitas pseudopalavras geradas em português eram impossíveis de ler, pois não seguiam as regras fonotáticas da língua, o que motivou a realização desta dissertação, de modo a serem desenvolvidos outros algoritmos geradores de pseudopalavras, em Português Europeu, que sigam essas mesmas regras da língua. Os cálculos lexicais provenientes do *Wuggy* serviram de base para desenvolvimentos semelhantes, no nosso sistema. Os programas em 2.2.3 e 2.2.2 não foram explorados em detalhe, principalmente porque o *Wuggy* tem a possibilidade de ser expandido para outras línguas.

2.2.2 MCWord

O *MCWord*, [7], é um gerador de pseudopalavras e/ou palavras em inglês, que tem como base de dados o *CELEX*. O *MCWord* é uma página HyperText Markup Language (HTML) e está disponível em `http://www.neuro.mcw.edu/mcword/`. O gerador também permite a obtenção de métricas lexicais e gerar pseudopalavras com diferentes graus de proximidade lexical. A geração é feita numa página semelhante à seguinte:

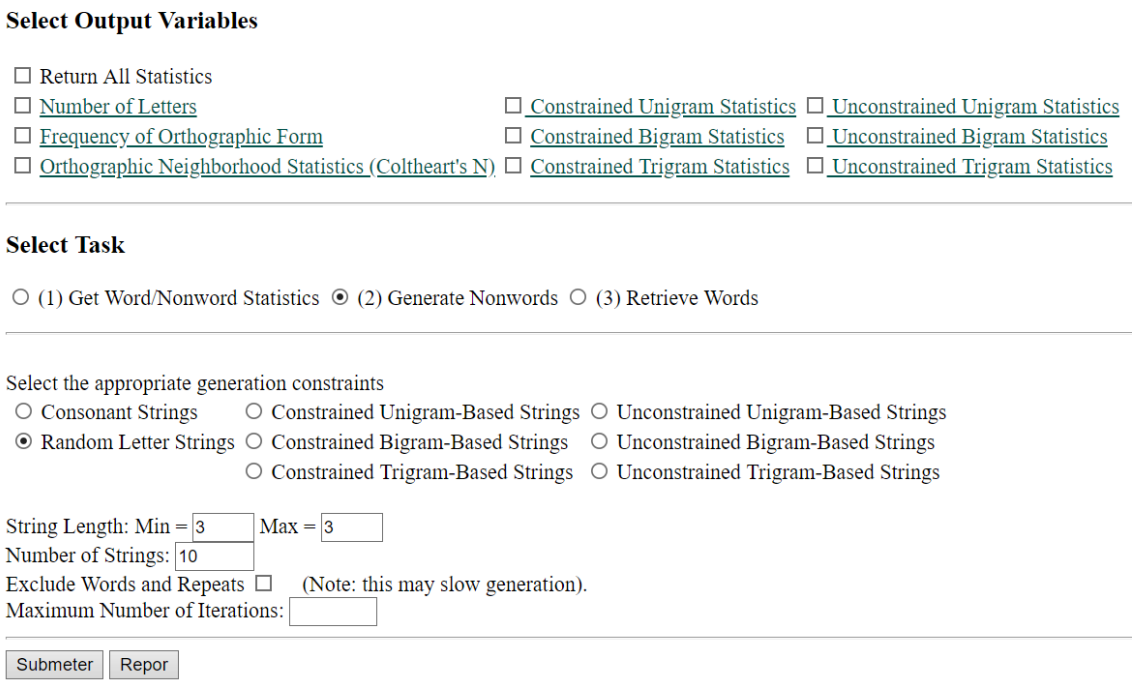

Figura 2.4: Especificações gerais, restrições e tipos de palavras possíveis gerar.

De modo a gerar pseudopalavras, foi escolhido o modo de geração baseado em "Constrained Bigram-Based Strings" e os resultados possíveis para pseudopalavras de 3 a 20 letras, foram os seguintes:

STRING	LEN	FREQ	Orth	Orth_F	N1_F	N1_C	N2_F	N2_C	N3_F	N3_C	UN1_F	UN1_C	UN2_F	UN2_C	UN3_F	UN3_C
and	3	28470.7	5	375.88	37263.41	37.67	29341.41	2.50	28470.70	1.00	278303.78	33985.00	55124.62	4203.00	32583.14	814.00
connection	10	34.6843	2	0.65	2716.69	816.20	1250.52	235.56	746.14	103.75	295858.57	37863.80	23651.83	4271.89	5221.43	1077.62
electroencephalogram	20	0.0594928	0	0.00	0.06	1.00	0.06	1.00	0.06	1.00	287151.73	36206.65	15697.62	2619.11	1368.60	296.44
everweight	10	NA	1	4.94	1184.53	405.50	153.21	28.67	100.80	13.88	296654.25	35605.70	16477.71	2098.44	3684.95	335.25
industrialization	17	6.24674	0	0.00	10.84	13.12	8.45	5.88	7.38	3.80	272357.19	36836.47	25756.96	4569.25	3755.26	896.53
inforethed	10	NA	0	0.00	2138.55	709.50	507.15	154.67	42.51	12.75	309888.24	36656.70	54327.55	4470.22	13677.12	336.62
internationalization	20	0.0594928	0	0.00	0.06	1.10	0.06	1.00	0.06	1.00	313692.25	39517.90	29714.90	5541.32	5557.72	1341.94
rationalization	15	4.81891	1	13.39	52.10	51.53	31.17	22.07	26.58	16.38	298767.20	37499.47	25792.91	5093.86	6122.78	1561.69
throure	7	NA	0	0.00	5443.86	663.57	1179.12	59.17	725.04	17.60	297858.54	34943.43	44522.23	3125.83	3669.28	337.60
yeard	5	NA	4	301.91	8984.10	346.60	2028.57	48.75	996.07	11.00	278425.55	34992.80	16960.46	2218.75	3528.22	367.00

Figura 2.5: Pseudopalavras geradas através de "Constrained Bigram-BasedStrings".

Verificou-se que foram obtidas muitas palavras do léxico, tais como "and" e "connection", em vez de efetivamente pseudopalavras, o que se pode concluir que não é feita uma verificação para isso mesmo.

Outro exemplo, foi a escolha do modo de geração através de "Constrained Unigram-Based Strings" e os resultados possíveis para pseudopalavras de 3 a 20 letras, foram os seguintes:

STRING	LEN	FREQ	Orth	Orth_F	N1_F	N1_C	N2_F	N2_C	N3_F	N3_C	UN1_F	UN1_C	UN2_F	UN2_C	UN3_F	UN3_C
casine	6	NA	4	2.11	8866.97	781.17	1584.60	137.00	126.63	14.50	312596.13	44356.00	33276.92	5168.60	2220.89	520.50
eotrnrsasopaielm	16	NA	0	0.00	12.45	25.19	0.73	0.73	0.05	0.07	300081.63	40226.19	10758.34	1701.27	158.37	34.64
inecrratioealalatiam	20	NA	0	0.00	0.06	1.10	0.03	0.53	0.01	0.17	315360.51	40248.40	23340.74	4480.11	2358.39	694.28
intetnoeionahilagron	20	NA	0	0.00	0.06	1.00	0.03	0.47	0.01	0.22	318234.79	39281.05	20658.09	3353.84	1786.27	350.00
iouotirnntellbgioae	19	NA	0	0.00	0.50	1.95	0.20	0.56	0.13	0.18	290474.88	37472.84	13857.30	2435.78	706.54	150.76
nerlrcalalciils	15	NA	0	0.00	44.75	60.27	2.80	3.29	0.00	0.00	256358.67	38343.20	15206.96	2755.29	511.55	155.00
pasloare	8	NA	0	0.00	3565.13	870.12	297.72	81.71	17.33	6.00	295401.92	39544.50	21361.05	2948.14	1529.84	135.67
pdmuocseattice	14	NA	0	0.00	138.28	92.57	39.44	19.62	0.00	0.08	271421.44	35267.21	13979.07	2420.15	1115.60	196.50
prrarer	7	NA	0	0.00	7255.03	1123.57	1328.65	211.83	21.26	7.40	279034.65	39860.00	31257.08	5226.33	1652.80	147.40
tsgueemaeet	11	NA	0	0.00	758.01	427.09	17.13	8.30	0.65	0.56	347242.40	43001.18	8826.94	1391.40	441.95	61.33

Figura 2.6: Pseudopalavras geradas através de “Constrained Unigram-BasedStrings”.

Verificou-se com estes resultados que existem bastantes pseudopalavras que são impossíveis de ler. Este programa, tal como aconteceu com o programa anterior, não foi explorado com o devido detalhe e por isso só é feita uma breve abordagem.

2.2.3 WordGen

O *Wordgen*, [5], é um *software* gerador de pseudopalavras e de palavras, que tem como base de dados o *CELEX* (holandês,inglês,alemão e cirílico) e o *Lexique* (francês). Este sistema permite a geração de palavras/pseudopalavras em diversas línguas entre elas, o holandês, o inglês, o alemão e o francês; através da combinação de restrições linguísticas [5]. O programa está disponível em `http://www.wouterduyck.be/?page_id=29` e através da introdução de um email, recebe-se um link de modo a poder ser extraído o programa e posteriormente instalado. Este programa não foi explorado com o devido detalhe e por isso só é feita uma breve abordagem. A janela que permite a geração de pseudopalavras é a seguinte:

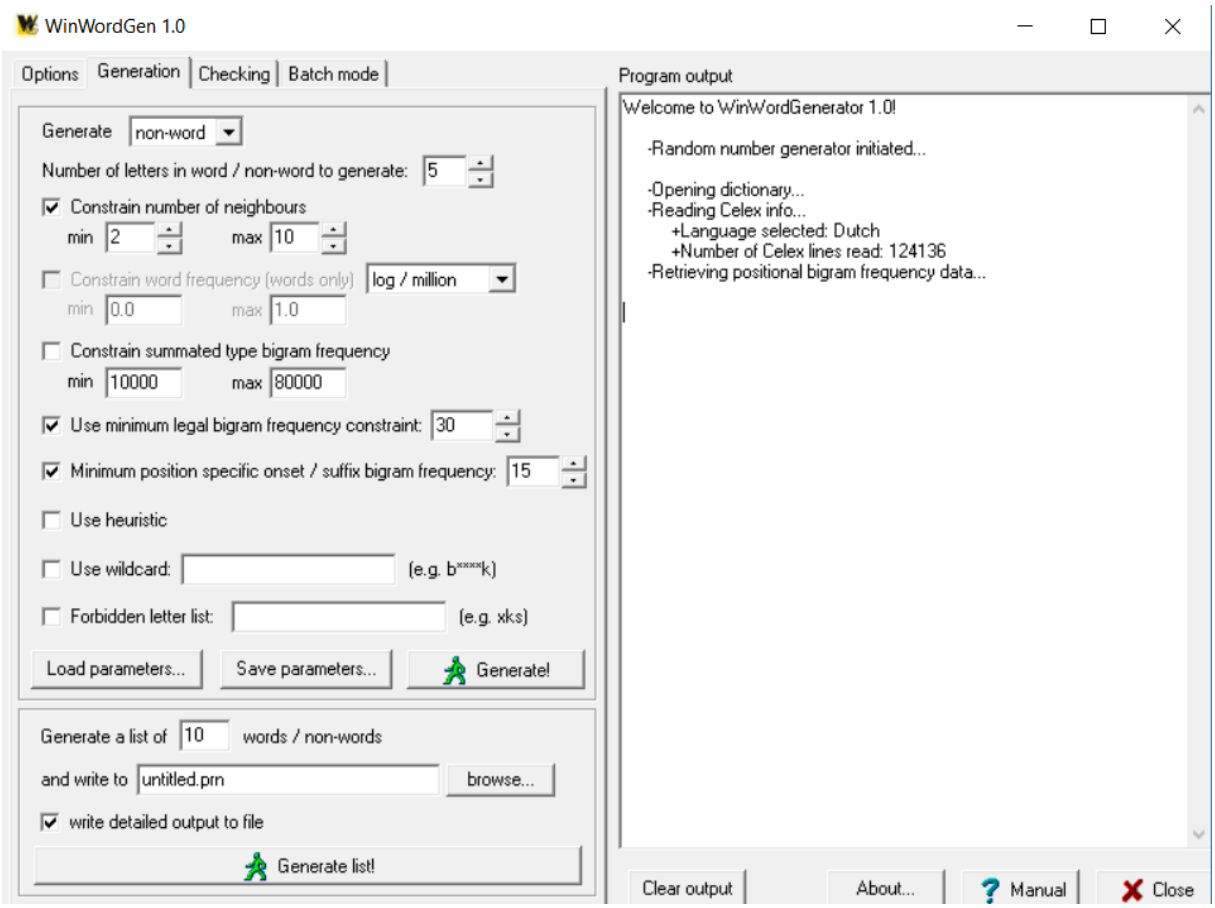

Figura 2.7: Janela inicial do WordGen

Capítulo 3

Corpus lexical

Neste capítulo é abordada a formação do corpus lexical que será usado na geração de PP. Este *corpus* deverá ter as palavras e a sua divisão silábica, uma vez que a geração de PP é baseada em sílabas.

3.1 *Corpura* lexicais

Para a formação de pseudopalavras é necessário ter uma boa base de dados lexical com diferentes tipos de informação, isto é, o *corpus* lexical tem que ser o mais rico possível. Assim, o primeiro objetivo foi a extração de palavras de léxico e a sua divisão silábica e guardadas à medida que eram extraídas. A esse ficheiro de léxico derivam diferentes ficheiros que serão, por sua vez, a espinha dorsal para todo este projeto. Esses processo serão descritos mais para a frente.

Com a necessidade de uma base de dados em pt-PT, uma das escolhas recaiu para o projeto P-Pal [11], que depois de comparada com outras base de dados pareceu a mais completa, já que para além de conter cerca de 200,000 palavras (incluindo formas verbais), continha a divisão silábica de todas essas palavras e outras informações lexicais. É o caso, por exemplo, da frequência por milhão, que ao início se achou importante mas acabou por não ser usada, pois as informações necessárias para a formação de pseudopalavras foram calculadas, provenientes dos pares de sílabas e não das palavras em si.

3.1.1 P-PAL

O P-PAL, também conhecido como Procura-PALavras, é uma aplicação *Web* desenvolvida pela Universidade do Minho e baseia-se num *corpus* de 227 milhões de palavras. A construção da base de dados de suporte ao P-Pal é proveniente de textos jornalísticos, literários, género técnico-científico e didático e ainda o género miscelânea [11].

3.1.2 CETEMPúblico

Depois de obtido o léxico, proveniente do P-PAL, recorreu-se a um léxico com as 50,000 palavras mais frequentes do CETEMPúblico, léxico esse que foi utilizado, previamente,

em projetos que decorreram no mesmo laboratório que esta dissertação foi desenvolvida. O CETEMPúblico [8] é um *corpus* composto por palavras do português europeu, provenientes do jornal Público, que conta com cerca de 180 milhões de palavras. E verificou-se que não existiam nomes próprios, tais como nomes de pessoas, cidades, países, etc. Todas as palavras que não estivessem no nosso ficheiro de léxico foram adicionadas.

3.2 Técnicas para tratamento do léxico

De seguida, foram arranjadas estratégias para apagar formas não existentes em português (caso de estrangeirismos), palavras compostas, siglas ou ainda palavras sem divisão silábica ou muito pouco frequentes. Estes processos são descritos nas subsecções seguintes:

3.2.1 Estrangeirismos

Estrangeirismos são palavras provenientes de outras línguas que são utilizadas e/ou empregadas na nossa língua, como por exemplo, "surf" e "jazz", através de certas combinações de sílabas que na nossa língua muitas vezes não acontecem.

Um dos objetivos deste trabalho passava pelo tratamento do *corpus*, depois de recolhidas as palavras provenientes do P-Pal [11] e do CETEMPúblico [8], de modo, a que não existam nenhuns estrangeirismos, ou o menor número possível dos mesmos. Como seria de esperar é impossível afirmar que não exista pelo menos um estrangeirismo, contudo foram utilizadas diferentes técnicas para garantir que esse número tende para zero.

Muitos estrangeirismos foram identificados pela falta de divisão silábica das palavras provenientes do P-Pal, já noutros não foi assim tão simples e assim foi necessário fazer uma procura manual através de técnicas de expressões regulares, por exemplo. Os outros estrangeirismos foram identificados através de alguns dos seguintes critérios:

- Palavra que contivessem "k", "y" e "w". Exemplo: "ketchup".
- Dígrafos em "cc", "dd", "ff", "gg", "ll", "pp", "tt", "zz". Exemplo: "Garrett".
- Dígrafos derivados do inglês: "th", "sh", "oo". Exemplo: "theme".
- Composições de letras não existentes no português, tais como "ght", "gns". Exemplo: "light".
- Palavras terminadas em "ing", "ingle", "ium", "n". Exemplo: "jingle".
- Palavras começadas com "up". Exemplo: "upgrade".
- Palavras que contivessem "diesel", "design", "hertz". Exemplo: "biodiesel".

É de notar que nem todas as palavras que contivessem uma certa sequência de letras, enumeradas ou não anteriormente, eram, efetivamente, estrangeirismos, ja que todas as

palavras apagadas do léxico foram revistas uma a uma.

Foi feito um algoritmo para identificar possíveis sílabas que só aconteciam uma vez em todo o léxico, o que deu entre 400 a 500 sílabas. De seguida, foi feita uma verificação manual e concluiu-se que muitas são estrangeirismos, outras tinham a divisão silábica mal indicada. As sílabas que faziam parte de estrangeirismos, resultou num apagamento de todos esses estrangeirismos.

As palavras que continham sílabas com frequência igual a 1 que foram apagadas foram: "bad", "bies", "blen", "blé", "blés", "clai", "clips", "corn", "cors", "cto", "cue", "cues", "céns", "dho", "proust", "dplo", "due", "fif", "fles", "gangs", "gies", "gins", "glam", "gne", "gour", "greens", "grom", "gés", "hot", "ib", "il", "jé", "leib", "lud", "lús", "ners", "nietz", "nox", "ohm", "ohms", "pgra", "poufs", "poule", "prez", "reau", "rries", "rrés", "sign", "soc", "spiel", "stend", "ssier", "sta", "teaus", "vs", "tris", "zló", "zón" e "ción". Nota: se anteriormente não tivessem sido apagados tantos estrangeirismos, através da pesquisa exaustiva de palavras que contivessem uma certa sequência de letras, encontraríamos muitas mais sílabas com frequência igual a 1, em termos de ocorrência na língua, que seriam para apagar.

3.2.2 Hífen

Foram retiradas todas as palavras com hífen da nossa base de dados até então, mas não apagadas, isto é, as palavras foram separadas em duas palavras isoladas, pelo elemento em comum entre elas, o hífen, de modo a serem comparadas individualmente com cada uma das palavras do léxico. Se essas palavras não forem estrangeirismos e caso não existam ainda, são adicionadas, de modo a enriquecerem a nossa base de dados. Por exemplo, a palavra "escolas-piloto" foi separada em "escolas" e "piloto". De seguida foi comparada com o léxico até então, de modo a verificar se existe "escolas" e "piloto", se não existirem são adicionadas.

3.2.3 Siglas e outros

As siglas também foram retiradas, caso de por exemplo, "ADN", já que, não vão beneficiar na formação de PP. Palavras acabadas em "n", foram igualmente eliminadas, já que são palavras provenientes do latim, exemplo de "íman".

3.2.4 Lince

Verificou-se que o léxico obtido até este ponto continha palavras que não cumpriam as normas estabelecidas pelo mais recente acordo ortográfico (AO90) [2] e por isso houve a necessidade da utilização de um conversor ortográfico para o efeito. O conversor ortográfico utilizado foi o Lince [3], disponibilizado pelo "Portal da Língua Portuguesa", que permite converter ficheiros de texto para o mais recente acordo ortográfico. Exemplo de conversão:

Objecto → Objeto

Depois de convertidas as palavras, mantiveram-se, tanto as palavras pré como as pós-acordo ortográfico, de modo, a aumentar ainda mais o léxico e sem que haja alguma distinção do acordo ortográfico em uso. Este processo permitiu ter mais encontros silábicos (pares de sílabas) possíveis e mais sílabas únicas.

3.3 Base de dados lexical

Foi possível obter, depois de todos estes processos e técnicas, um ficheiro final com o léxico e a sua divisão silábica. Este ficheiro tem o nome "Dicionário_div_PPAL_v7.txt" e tem as seguintes características (tabela 3.1):

1. Está ordenado por ordem alfabética;

2. Contém 194,034 palavras de léxico e a correspondente divisão silábica.

	Palavras	**Divisão silábica**
1	a	a
2	aba	a-ba
3	abacate	a-ba-ca-te
...	...	...
194034	úvulas	ú-vu-las

Tabela 3.1: Léxico e a sua divisão silábica.

3.4 Vocabulários e bigramas

Depois de criado o ficheiro de léxico, ilustrado na tabela 3.1, houve a necessidade de incluir informações extra, necessárias para o funcionamento dos algoritmos de geração de PP. Para esse efeito, foram criados vocabulários e bigramas de sílabas. A criação de um vocabulário de sílabas, foi importante pois conteria todas as sílabas únicas existentes na língua que serviria para a posterior construção das pseudopalavras. Foi necessário um vocabulário extra, para que a geração de pseudopalavras de 1 sílaba fosse imediata. Esse vocabulário conteria apenas sílabas que pudessem ser pseudopalavras de 1 sílaba. Por fim, foi necessário a criação de bigramas de sílabas, pois conteriam a informação de todos os encontros silábicos encontrados na língua.

Para o efeito, foi criada uma função de nome "Cria_voc_bigrama", que recebe como parâmetros de entrada todas as palavras do léxico e a respetiva divisão silábica e produz quatro ficheiros, indicados de seguida.

3.4.1 Ficheiros criados a partir da base de dados lexical

É escrito para um primeiro ficheiro, "word_v2.txt" todas as palavras do léxico, apenas, de modo a ser posteriormente usado para cálculos e informações de proximidade lexical que foram desenvolvidos em C++. Apresenta as seguintes características:

1. Está ordenado por ordem alfabética;
2. Contém todas as palavras do léxico (apenas);

A tabela 3.2 mostra exatamente o que foi enunciado.

	Palavras
1	a
2	aba
3	abacate
...	...
194034	úvulas

Tabela 3.2: Todas as palavras do léxico.

Outro dos ficheiros criado, de nome "vocabulario_v4.txt", é um vocabulário de sílabas únicas. Contém 2749 sílabas diferentes que foram obtidas através da separação das divisões silábicas das palavras e adicionadas sempre que houvessem sílabas novas. Apresenta as seguintes características:

1. Está ordenado por ordem alfabética;
2. Contém todas (2749) as sílabas (únicas).

Um exemplo da forma e do conteúdo do ficheiro é dado pela tabela 3.3:

	Vocabulário de sílabas
1	a
...	...
518	diz
...	...
2477	vrai
...	...
2749	ús

Tabela 3.3: Vocabulário de sílabas.

Outro, "vocabulario_v4_nlex.txt", é semelhante ao enunciado anteriormente, ou seja, é um vocabulário de sílabas, mas neste caso difere do anterior, na medida em que as sílabas não podem formar sozinhas palavras (de 1 sílaba) e têm que poder ser de ínicio e fim

de palavra. Ou seja, sílabas que contenham dígrafos e afins e acabem com determinadas consoantes não aparecem. As palavras, por exemplo, "tem", "çar" e "frac" não existem neste vocabulário, pois o primeiro exemplo é uma palavra do léxico, o segundo começa com um "ç" e o terceiro acaba com um "c". Apresenta as seguintes características:

1. Está ordenado por ordem alfabética;
2. Contém sílabas que sozinhas não formam palavra, como por exemplo "abs", "guim", "fer" (Não contém palavras que sozinhas formam palavra, exemplo, "tem", "a", etc);
3. Não tem palavras/sílabas, que comecem por "rr", "ç", "nh", "ss";
4. Não tem palavras/sílabas, que acabem com uma das seguintes consoantes, "c", "d", "f", "g", "j", "n", "p", "q", "t", "v".
5. 1596 palavras de 1 sílaba.

	Vocabulário de sílabas sem léxico
1	ab
...	...
518	féns
...	...
1415	vrai
...	...
1596	ús

Tabela 3.4: Vocabulário de sílabas que por si só não sejam palavras do léxico.

E por fim, um ficheiro denominado de bigramas de sílabas ("todas_big_v7.txt") que vai ter um papel crucial na criação das pseudopalavras, pois garante que as pseudopalavras sigam as regras fonotáticas da língua.

Este ficheiro contém na 1ª coluna (Par) todos os pares de sílabas possíveis. Tome-se como exemplo, "a-ba", "por-ta" ou "ta-ção".

A segunda (Ind1) e a terceira coluna (Ind2), correspondem ao índice no vocabulário completo de sílabas em "vocabulario_v4.txt", da primeira e da segunda sílaba, do par de sílabas. Por exemplo, no par de sílabas "a-ba", estas colunas têm os valores inteiros, 1 e 25, respetivamente. 1 porque "a" é a primeira sílaba no vocabulário e 25 porque "ba" é a 25ª sílaba.

A quarta coluna (Início) indica se a primeira sílaba pode ser de início de palavra e a quinta coluna (Fim) indica se a última sílaba pode ser de fim, isto através de valores lógicos. A obtenção destes valores proveio da verificação, em todo o léxico, se alguma vez as sílabas começaram como início e de fim de palavra, respetivamente. 1 se pode ser de início/fim, 0 caso contrário.

A sexta (Ac1) e a sétima coluna (Ac2) também têm valores lógicos e indicam se a primeira e a segunda sílaba do par têm algum acento, respetivamente.

O ficheiro está ordenado de forma decrescente pela coluna seguinte, que é neste caso a oitava (Oc), que indica o número de ocorrências de cada par de sílabas no léxico.

Por fim as três últimas colunas (Ini, Meio e Final), indicam o número de ocorrências de cada par de sílabas, como um todo, no início, meio e fim das palavras. Um resumo deste ficheiro é apresentando de seguida:

1. Contém todas as combinações de pares de sílabas possíveis;
2. Informação acerca dos índices das sílabas no vocabulário de sílabas;
3. Indica se a primeira sílaba do par pode ou não ser de início de pseudopalavra e se a última pode ou não ser de fim;
4. Indica se a primeira e a última sílaba do par têm algum acentuado ou não;
5. Está ordenado, de forma decrescente, pelo número de ocorrências de cada par de sílabas na língua;
6. Número de ocorrências no início, meio e fim das palavras;
7. Ficheiro com 52,319 pares de sílabas diferentes.

Um exemplo ilustrativo para as bigramas de sílabas é o seguinte:

	Par	**Ind1**	**Ind2**	**Início**	**Fim**	**Ac1**	**Ac2**	**Oc**	**Ini**	**Meio**	**Final**
1	ri-a	1870	1	1	1	0	0	3143	4	636	2503
...	...	...	...	...	...	...	...	...	...	...	...
831	a-la	1	1153	1	1	0	0	112	105	5	2
...	...	...	...	...	...	...	...	...	...	...	...
9357	rrei-ri	1913	1870	0	1	0	0	10	0	10	0
...	...	...	...	...	...	...	...	...	...	...	...
21623	pa-tá	1586	2366	1	1	0	1	3	0	3	0
...	...	...	...	...	...	...	...	...	...	...	...
35827	lho-tan	1202	2207	0	0	0	0	1	0	1	0
...	...	...	...	...	...	...	...	...	...	...	...
52319	ú-til	2745	2242	1	1	1	0	1	0	1	0

Tabela 3.5: Bigramas de sílabas.

Capítulo 4

Geração de Pseudopalavras

Neste capítulo são abordados em detalhe os algoritmos que foram desenvolvidos para a geração de PP.

4.1 Algoritmos principais para a geração de PP

São chamados de algoritmos principais aos dois métodos diferentes de gerar pseudopalavras, " Gerador de PP de 1-10 sílabas" (gera_pp) e "Palavra Protótipo" (palavra_prot).

4.1.1 Gerador de PP de 1-10 sílabas

O algoritmo "Gerador de PP de 1-10 sílabas" (gera_pp), é uma função que gera pseudopalavras através da combinação de pares de sílabas que sigam as regras fonotáticas da língua, de modo a poder retornar o número de pseudopalavras desejado ou tantas quanto possível. Como o nome indica, gera no mínimo PP de 1 sílaba e no máximo PP de 10 sílabas. Como o maior número de sílabas encontrado, em todas as palavras do léxico construído neste projeto, foi 10, será também o máximo número de sílabas possível que uma pseudopalavra pode ter.

Um exemplo demonstrativo para a geração de 5 pseudopalavras de 4 sílabas através deste algoritmo seria por exemplo:

Pseudopalavras	**Divisão silábica**
colmative	col - ma - ti - ve
assobina	a - sso - bi - na
inducava	in - du - ca - va
procôndia	pro - côn - di - a
sacristirá	sa - cris - ti - rá

A divisão silábica não é retornada pelo algoritmo; e aqui foi apresentada para melhor perceção do algoritmo.

De maneira a que as pseudopalavras sigam as regras fonotáticas da língua, serão usadas informações acerca dos pares de sílabas, provenientes das bigramas de sílabas, para a aquisição de pares de sílabas possíveis para formarem as pseudopalavras.

Tomando como exemplo o caso da geração de pseudopalavras com 4 sílabas, os critérios de escolha para os 3 pares de sílabas que a formam esse tipo de pseudopalavras seriam os seguintes:

Para o 1ª par de sílabas:

- A 1ª sílaba tem de poder ser de início de palavra.
- A 1ª sílaba só pode ter um acento se número de sílabas (Nsil)=3.

Para o 2ª par de sílabas:

- A 1ª sílaba tem de fazer par com a 2ª sílaba do par anterior.
- A 1ª sílaba pode ter um acento mas a 2ª não pode.

Para o 3ª par de sílabas:

- A 1ª sílaba tem de fazer par com a 2ª sílaba do par anterior.
- A 2ª sílaba de poder ser de fim de palavra.
- A 2ª sílaba pode ter um acento se não houver nenhum na antepenúltima sílaba.

Os pares de sílabas que verifiquem estas condições podem ser escolhidos através da função "datasamplemex", implementada em C++. Funciona de forma idêntica à rotina do *MATLAB* com o nome "datasample". Por sua vez está interligada com o *MATLAB*, já que o *MATLAB* tem ferramentas que permitem chamar funções compiladas em C++ e usá-las como se deste se tratassem. As rotinas especiais feitas para serem executadas e chamadas pelo *MATLAB* são denominadas de *MEX Files*. Esta função retorna observações de uma amostra segundo as suas probabilidades (sem reposição). A seleção de pares de sílabas é feita através dos seus valores de ocorrência, no início, meio ou fim de palavra. Este método permite que aconteçam, com mais frequência, encontros silábicos mais frequentes nas diferentes posições (início, meio e fim) das pseudopalavras.

Para as diferentes posições são retirados diferentes valores de pares de sílabas da função anterior, pois estes valores diferem consoante o número de sílabas.

- Se Nsil = 2, são escolhidas dos candidatos 4 * Npp pares de sílabas.
- Se Nsil = 3, 2* Npp pares de sílabas na 1ª e 2ª posição e 4 * Npp na última sílaba.

- Se Nsil é maior que 3, Npp pares de sílabas na 1ª posição e na 2ª e 2*Npp nas restantes.

Os valores diferem na obtenção de pares de sílabas em pseudopalavras com diferentes números de sílabas, pois as palavras com poucas sílabas, nomeadamente as de 2 e as de 3 sílabas, tendem a ser palavras do léxico. Se isso acontecer, serão posteriormente eliminadas.

Como já foi enunciado, são retiradas diferentes amostras por posição da palavra, consoante o número de sílabas. As sílabas são codificadas com o seu índice.

De seguida, são convertidos todos os índices das sílabas para, efetivamente, sílabas através de uma função de nome "int2sil". Esta função converte índices de sílabas no vocabulário total de sílabas para as sílabas (strings).

Depois de obtidas as pseudopalavras são feitas verificações às mesmas. As regras são as seguintes:

1. Verifica se a pseudopalavra existe no léxico;
2. Se a pseudopalavra acabar com "s", verifica se existe no singular no léxico;
3. Se a pseudopalavra não acabar com "s", verifica se existe no plural no léxico;
4. Verifica se tem alguma sílaba com algum acento, se tiver não pode acabar com, "z", "u", "us", "bi", "ci", "di", "fi", "gi", "gui", "hi", "ji", "li", "mi", "ni", "pi", "qui", "ri", "si", "ti", "vi", "xi", "zi", "bis", "cis", "dis", "fis", "gis", "guis", "his", "jis", "lis", "mis", "nis", "pis", "quis", "ris", "sis", "tis", "vis", "xis", "zis".

Depois de efetuadas as verificações o algoritmo escolhe, através da informação proveniente das regras, PP de forma aleatória para serem retornadas ao utilizador tantas quanto desejadas, se possível. Caso contrário retorna todas as possíveis.

4.1.2 Palavra Protótipo

O algoritmo "Palavra Protótipo"(palavra_prot) é uma função que gera o número de pseudopalavras (Npp) desejado ou tantas quanto possível, através de uma palavra protótipo, palavra essa que tem de existir no léxico.

O objetivo deste algoritmo é arranjar combinações diferentes na 1ª sílaba mantendo as restantes, combinações diferentes na 2ª sílaba mantendo as restantes e por aí adiante até chegar ao fim da palavra. O número de combinações é igual ao Npp por posição da palavra. Um exemplo explicativo, para a introdução da palavra "riacho" e pretender-se 100 PP parecidas a esta, é o seguinte:

[100] - **a** - **cho**
ri - [100] - **cho**
ri - **a** - [100]

O algoritmo procura o índice no léxico da palavra protótipo introduzida, de modo a obter informação acerca da sua divisão silábica. Com a divisão silábica é possível saber quantas sílabas a palavra tem, através da contagem do número de hífenes, já que 1 hífen indica que uma palavra tem 2 sílabas, 2 hífenes indica que a palavra tem 3 sílabas e por aí adiante. Por outras palavras cria pseudopalavras com distância igual a 1 sílaba à palavra protótipo.

Se a palavra tiver 2 sílabas é invocado o algoritmo descrito em 4.2.2, caso contrário a geração é feita dentro deste algoritmo. Todos os ficheiros que funcionam como espinha dorsal dos algoritmos, vão ser todos usados à exceção do vocabulario_nlex.txt.

O algoritmo converte as sílabas *strings* que formam a palavra protótipo em índices (inteiros) no vocabulário de sílabas, através da função sil2int.

De maneira a que as pseudopalavras sigam as regras fonotáticas da língua, serão usadas informações acerca dos pares de sílabas, provenientes das bigramas de sílabas, para a aquisição de pares de sílabas possíveis para formarem as pseudopalavras. Tomando como exemplo, a palavra "riacho" que tem 3 sílabas, os critérios de escolha para as 3 diferentes sílabas seriam os seguintes:

Para a 1ª sílaba:

- Esta sílaba tem de fazer par de sílabas com a 2ª sílaba da palavra protótipo.
- Esta sílaba não pode ser igual à 1ª sílaba da palavra protótipo.
- Tem de poder ser de início de palavra.
- Esta sílaba só pode ter um acento se a palavra protótipo tiver 3 sílabas .
- Se já houver algum acento nas restantes sílabas da palavra protótipo, esta sílaba não pode ter nenhum acento.

Para a 2ª sílaba:

- Esta sílaba tem de fazer par de sílabas com a 1ª sílaba da palavra protótipo.
- Esta sílaba tem de fazer par de sílabas com a 3ª sílaba da palavra protótipo.
- Esta sílaba não pode ser igual à 2ª sílaba da palavra protótipo.
- Esta sílaba não pode ter nenhum acento.

Para a 3ª sílaba:

- Esta sílaba tem de fazer par de sílabas com a 2ª sílaba da palavra protótipo.
- Esta sílaba não pode ser igual à 3ª sílaba da palavra protótipo.
- Tem de poder ser de fim de palavra.
- Se já houver algum acento nas restantes sílabas da palavra protótipo, esta sílaba não pode ter nenhum.

As sílabas que verifiquem estas condições podem ser escolhidas através da função "datasamplemex", explicada em 4.1.1, ou seja, aqui também são escolhidas sílabas através do seu peso nas diferentes posições, início (para a 1ª sílaba), meio (para a 2ª sílaba) e fim (para a 3 sílaba). Como já foi enunciado, são retiradas (no máximo) Npp amostras por posição da palavra.

Ao obter-se as PP são feitas verificações às mesmas, tal como no algoritmo anterior.

Depois de efetuadas as verificações o algoritmo escolhe, através da informação proveniente da variável lógica anterior, pseudopalavras de forma aleatória para serem retornadas ao utilizador tantas quanto desejadas, se possível, sem qualquer tipo de ordenação. Caso sejam pedidas mais PP do que as hipóteses possíveis, o algoritmo retorna todas as hipóteses, ordenadas pela sílaba mudada.

Cinco pseudopalavras possíveis através do exemplo "riacho", são os seguintes:

Pseudopalavras	**Divisão silábica**
giacho	[gi] - **a** - **cho**
rimancho	**ri** - [man] - **cho**
saiacho	[sai] - **a** - **cho**
riana	**ri** - **a** - [na]
rigacho	**ri** - [ga] - **cho**

4.2 Algoritmos auxiliares para a geração de PP

Os algoritmos auxiliares são incorporados nos algoritmos principais e são invocados em determinados casos concretos passando despercebidos quando o utilizador está a gerar pseudopalavras. Estes algoritmos serão abordados e explicados devidamente nas subsecções, 4.2.1 e 4.2.2 .

4.2.1 Gerador de PP de 1 sílaba

A função que gera PP de 1 sílaba é denominada de gera_pp_1sil e pode ser invocada por qualquer um dos dois algoritmos principais, isto é, caso se pretenda gerar PP de 1

sílaba tanto se pode escolher o algoritmo descrito em 4.1.1 como o algoritmo em 4.1.2, caso a escolha recaia em 1 sílaba ou na introdução duma palavra protótipo de 1 sílaba, respetivamente. Tal processo está ilustrado na figura 4.1.

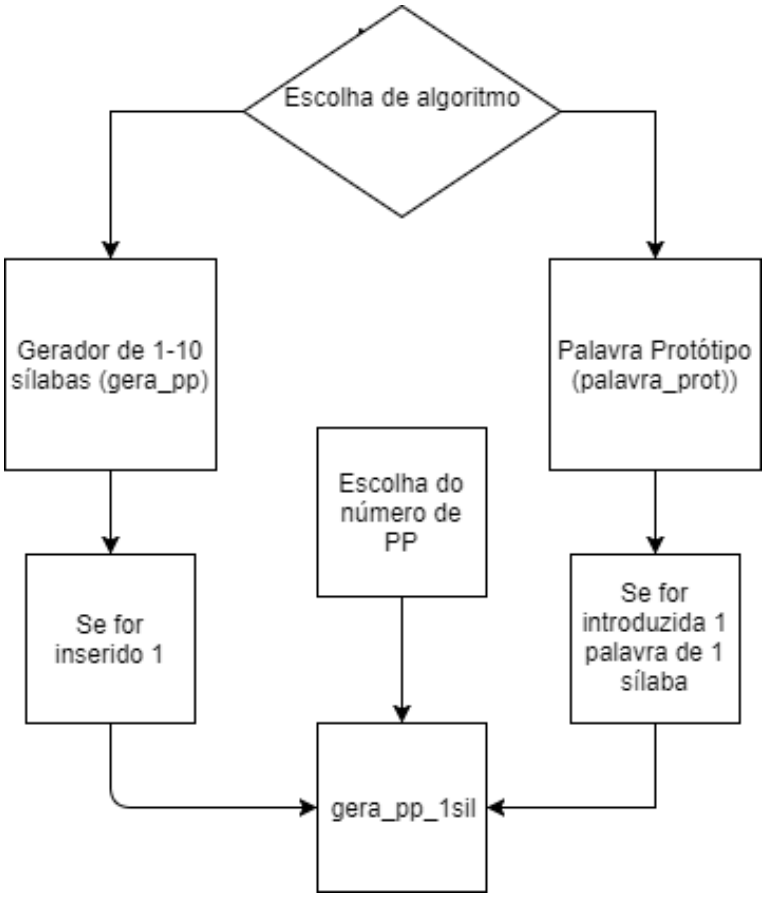

Figura 4.1: Fluxograma explicativo da invocação do algoritmo gera_pp_1sil.

Este algoritmo retorna aleatoriamente tantas sílabas quanto o número desejado, provenientes do vocabulario_v4_nlex.txt que já está devidamente preparado, para que todas as sílabas sejam pseudopalavras de 1 sílaba. Retorna no máximo 1596 pseudopalavras.

4.2.2 Palavra protótipo de 2 sílabas

A função de nome palavra_prot_2sil é invocada quando se introduz uma palavra protótipo de 2 sílabas no algoritmo descrito em 4.1.2 e retorna PP de 2 sílabas, variações de 1 sílaba, da palavra protótipo. A chamada a esse algoritmo é descrita pela figura 4.2.

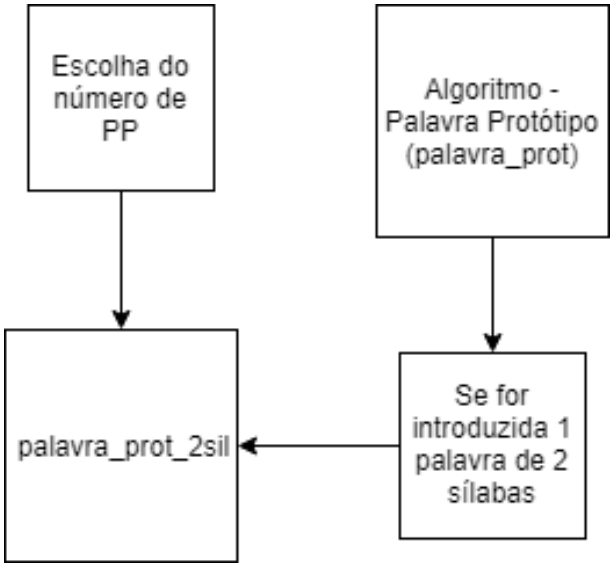

Figura 4.2: Fluxograma explicativo da invocação do algoritmo palavra_prot_2sil.

O algoritmo em questão é muito semelhante ao algoritmo "Palavra Protótipo", com a diferença que neste caso não interessa saber se a 1 ª sílaba é de início ou não e se a 2ª sílaba é de de fim ou não. E por isso, são feitas verificações extra:

1. As pseudopalavras não podem começar com "rr", "ç", "nh", "ss";

2. As pseudopalavras não podem acabar com "c", "d", "f", "g", "j", "n", "p", "q", "t" e "v".

Em detrimento da escolha de Npp sílabas por posição, como acontece no algoritmo "Palavra Protótipo", este algoritmo escolhe (3×Npp) sílabas por posição da palavra e assim é de esperar que pelo menos um 1/6 das hipóteses sejam pseudopalavras. Muitas hipóteses serão eliminadas devido às verificações enunciadas anteriormente, principalmente porque existe uma elevada probabilidade das pseudopalavras de 2 sílabas serem palavras do léxico. As sílabas são escolhidas através de amostragem sem reposição, segundo as suas probabilidades de ocorrência, nomeadamente, no início e no fim.

Um exemplo explicativo para este algoritmo é a introdução da palavra "sempre" com o intuito de se gerar 5 pseudopalavras (Npp) parecidas a esta. A sua divisão silábica é dada por "sem-pre". O algoritmo procura 15 sílabas, na 1ª posição, que formem par com a sílaba "pre" e 15 sílabas, na 2ª posição, que formem par com "sem", exemplo de:

[15] - **pre**
sem - [15]

Cinco pseudopalavras através deste exemplo, são os seguintes:

Pseudopalavras	**Divisão silábica**
surpre	[sur] - **pre**
cupre	[cu] - **pre**
sembrai	**sem** - [brai]
depre	[de] - **pre**
sempas	**sem** - [pas]

4.3 Cálculos e informações lexicais

Para além dos algoritmos de geração de PP, foram desenvolvidos algoritmos que calculam e indicam informações lexicais acerca das PP para poderem ser apresentados ao utilizador. Esses algoritmos estão implementados em C++ e estão inseridos nos algoritmos em *MATLAB* através dos MATLAB executable (MEX) Files, introduzidos em 4.1.1. Como estes algoritmos foram desenvolvidos em C++, a obtenção das métricas lexicais é rápida de obter.

Uma das métricas calculadas é a Distância de *Levenshtein* que foi descrita em 2.2.1. A Distância de *Levenshtein* de, por exemplo, entre "apagar" e "pegar" é de 2, já que é necessário um apagamento do primeiro "a" e uma substituição do "a" pelo "e", para a obtenção da palavra "pegar" através da palavra "apagar". Este cálculo ajuda a explicar o OLD20 que esse sim é mostrado ao utilizador.

O OLD20 é um dos cálculos lexicais possíveis de obter e foi descrito em 2.2.1. Para uma melhor explicação recorre-se a dois exemplos. Um primeiro tendo a pseudopalavra "gara" que tem OLD20 igual 1, tem pelo menos 20 vizinhos com distância igual a 1. Os seus vizinhos são, nomeadamente, "ara", "cara", "gaba", "gafa", "gaga", "gaia", "gaja", "gala", "gama", "gana", "gare", "garoa", "garra", "garça", "gata", "gaza", "gera", "gira", "gora", "lara". Noutro segundo exemplo, com a pseudopalavra "trauterregno" o seu OLD20 é de 5.2, ou seja, em média são necessárias 5.2 operações para transformar esta pseudopalavra nos seu vizinhos ortográficos. Nestes dois casos verifica-se que a primeira pseudopalavra é muito parecida a palavras do léxico, já o mesmo não se pode afirmar da segunda. O OLD20 permite avaliar as pseudopalavras em termos de proximidade ao léxico, ou seja, quanto menor este valor, mais próxima será uma pseudopalavras às palavras do léxico.

Os vizinhos de diferentes distâncias (Dists) dividem-se em quatro informações lexicais. Nos vizinhos de distância 1 só por substituição (Dist1sub), vizinhos de distância 1 (Dist1), vizinhos de distância 2 (Dist2) e nos vizinhos de distância 3 (Dist3) que indicam quantos vizinhos (palavras do léxico), existem de distância um, dois e três, respetivamente, através de substituições, apagamentos e inserções. Para o caso de Dist1sub, um possível exemplo é, tendo a palavra "pato" possíveis vizinhos de distância 1 só por substituição são, "pata" , "gato", etc. Para um mesmo exemplo, possíveis vizinhos com Dist1, Dist2 e Dist3, com a mesma palavra ("pato") tem como vizinhos, "patos" (d=1), "patas" (d=2); "sapata" (d=3).

Por fim a última informação lexical é denominada de lista dos 20 vizinhos mais próximos (Lists). Por exemplo, a palavra "pato" tem como vizinhos mais próximos, "pata", "patos", "rato", etc.

4.4 Interface gráfico

Para o manuseamento dos algoritmos houve a necessidade de um *interface* gráfico com os consequente botões, para a fácil manipulação e geração de PP. Como os algoritmos estão implementados em *MATLAB*, decidiu-se utilizar também um *interface* com o utilizador em *MATLAB*. A aplicação foi desenvolvida através da ferramenta *Appdesigner*.

Os ficheiros que os algoritmos utilizam são logo carregados para memória quando o interface é aberto o que resulta numa melhor eficiência nos algoritmos e num menor tempo de espera de processamento.

4.4.1 Janela de apresentação e tempo de processamento

Na janela inicial há uma breve descrição do sistema, procedente pela escolha do número de pseudopalavras, que permitirá (se possível) retornar o número de pseudopalavras desejado. O interface assenta em dois algoritmos principais e assim existe a possibilidade de escolha individual dos mesmos, para a geração algorítmica de PP. Ou seja, é possível a escolha exclusiva do algoritmo "Gerador de pseudopalavras de 1-10 sílabas" ou do algoritmo "Palavra Protótipo". É possível a escolha mútua do campo "Resultados" (informações lexicais das pseudopalavras). Existem dois botões, um que se chama de "Gerar" onde é feita, efetivamente, a geração das pseudopalavra; o outro botão denominado de "Guardar como...", permite guardar a informação apresentada na tabela, para um ficheiro .txt ou .xlsx (excel), à escolha do utilizador. A janela inicial da aplicação é a seguinte:

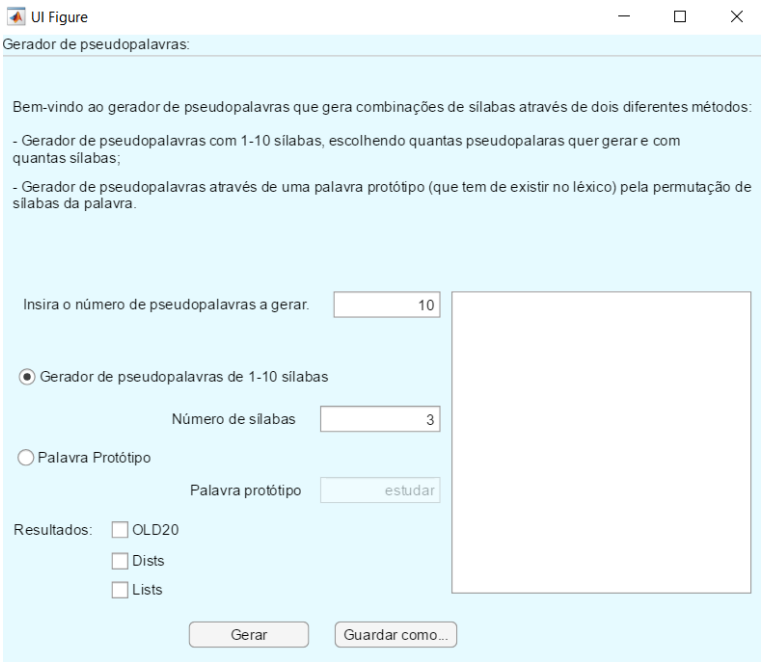

Figura 4.3: Janela de apresentação.

Se o utilizador escolher o algoritmo "Gerador de pseudopalavras de 1-10 sílabas", pode escolher quantas sílabas quer que as pseudopalavras tenham. Com esta opção selecionada, o interface não permite introduzir nenhuma palavra no algoritmo "Palavra Protótipo". Caso a escolha recaia para o segundo algoritmo "Palavra Protótipo", é dada a possibilidade de introdução de uma palavra, de forma a que as pseudopalavras sejam variações de 1 sílaba da mesma. O interface não permite, por sua vez, introduzir o número de sílabas, no algoritmo contrário (quando este algoritmo é escolhido).

O último campo é denominado de "Resultados", composto pelo OLD20, Dists e Lists. Se nenhum destes parâmetros for selecionado, quando se gera pseudopalavras, só é criada uma tabela com 1 coluna com uma lista das mesmas. Caso contrário são criadas as colunas correspondentes.

É indicado ao utilizador o tempo de processamento, através de uma *msgbox*. Exemplo ilustrativo é o seguinte:

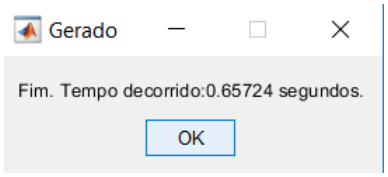

Figura 4.4: *Msgbox* com informação acerca do tempo de geração de pseudopalavras.

4.4.2 Parâmetros por omissão

De modo a garantir uma melhor experiência com o utilizador, existem certos parâmetros por omissão, casos do número de PP, número de sílabas e palavra protótipo.

Na inserção do número de PP, o valor predefinido é o número 10. O algoritmo selecionado, por omissão, é o primeiro algoritmo, definido como "Gerador de pseudopalavras de 1-10 sílabas", que por omissão gera PP de 3 sílabas. Caso o utilizador mude para o algoritmo "Palavra Protótipo", a palavra protótipo por omissão é "estudar". Inicialmente os "Resultados" não estão selecionados.

Capítulo 5

Resultados

Neste capítulo são apresentados possíveis resultados, aquando da experimentação e do contacto do *interface* com o utilizador e os consequentes resultados, consoante certas especificações.

Como já foi enunciado e descrito ao longo deste trabalho, existem dois algoritmos principais e dois auxiliares. Os resultados com estes algoritmos serão apresentados e descritos agora de seguida.

5.1 Resultados com o algoritmo Gerador de Pseudopalavras de 1-10 sílabas

Possíveis resultados com o algoritmo "Gerador de pseudopalavras de 1-10 sílabas", utilizando os parâmetros por omissão (10 pseudopalavras e 3 sílabas) e todos os "Resultados" selecionados, são apresentados na tabela 5.1.

	Pseudopalavras	**OLD20**	**Dist1sub**	**Dist1**	**Dist2**	**Dist3**	**Lists**
1	inconte	2.15	0	0	17	197	arconte,conte,...
2	ressora	1.85	1	3	34	328	ressoa,ressona,...
3	sublecer	2.85	0	0	3	29	sublevar,sublocar,...
4	demargar	2.2	1	1	14	142	demarcar,amargar,...
5	chilreite	2.9	0	0	5	12	chilreante,chilreia,...
6	engalha	1.8	4	4	36	274	encalha,engelha,...
7	demarga	2	1	1	18	305	demarca,amarga,...
8	lobrica	1.85	3	3	21	239	lobriga,lubrica,...
9	internal	1.7	5	6	27	150	infernal,interna,...
10	saltaba	1.85	3	3	33	364	saltada,saltara,...

Tabela 5.1: 10 resultados para a geração de 10 pseudopalavras de 3 sílabas.

Verifica-se através destes resultados, que as pseudopalavras são sempre possíveis de ler. A pseudopalavra mais próxima do léxico é a que tem o valor de OLD20 mais baixo. Neste caso é a pseudopalavra "internal" que por sua vez é a pseudopalavra que tem mais vizinhos tanto de distância 1 por substituição como de distância 1, por substituição, apagamento ou inserções de uma letra. Possíveis vizinhos desta palavra são, nomeadamente,

"infernal" e "interna".

O tempo médio de geração das 10 PP foi de 0.69927 segundos, o que equivale a um tempo de processamento de 69.927 ms por pseudopalavra.

Outro possível exemplo, com o intuito de gerar 100,000 PP de 3 sílabas, os primeiros 3 resultados e os 2 últimos foram os seguintes:

	Pseudopalavras	OLD20	Dist1sub	Dist1	Dist2	Dist3	Lists
1	riana	1.7	4	6	146	1699	ariana,diana,...
2	riado	1.3	5	14	240	1971	criado,fiado,...
3	menteca	2.1	0	0	18	288	centena,enoteca,...
4	óscane	2.6	0	0	8	156	escale,escame,...
5	óscaro	1.95	0	1	19	325	óscar,caro,...

Tabela 5.2: As primeiras 5 PP na geração de 1 milhão de PP de 3 sílabas.

O tempo médio de geração foi de 679.981 segundos (cerca de 11 minutos) e só foi possível gerar, no máximo, 22,186 PP, o que equivale a um tempo de geração por pseudopalavra de cerca de 30.65 ms.

Com o intuitivo de gerar 10 PP de 8 sílabas os resultados foram os seguintes:

	Pseudopalavras	OLD20	Dist1sub	Dist1	Dist2	Dist3	Lists
1	flautabotoziguacheste	13	0	0	0	0	farmacologicamente,...
2	deladotaipazinado	8.55	0	0	0	0	desacompanhado,...
3	desentranamengasteirar	12.65	0	0	0	0	desgovernamentalizar,...

Tabela 5.3: As 3 primeiras PP na geração de 10 PP de 8 sílabas.

O tempo médio de processamento foi de 0.70161 segundos, o que equivale a um tempo médio por pseudopalavra de 71.61 ms.

Um dos algoritmos denominado de gera_pp_1sil, gera PP de 1 sílaba tal como o nome indica. Este algoritmo é chamado quando ou no algoritmo "Palavra-protótipo" é introduzida uma palavra de 1 sílaba ou quando no algoritmo "Gerador de pseudopalavras de 1-10 sílabas" pretende-se gerar PP de 1 sílaba. Para a geração de 50 PP de 1 sílaba os resultados foram os seguintes:

	Pseudopalavras	OLD20	Dist1sub	Dist1	Dist2	Dist3	Lists
1	méns	1.8	2	4	82	922	améns,mins,...
2	hís	1.95	1	1	100	846	hás,ais,...
3	xou	1.8	3	4	137	1288	dou,ou,...
4	déis	1.7	6	6	87	1041	dais,deis,...
5	ãos	1.35	7	13	233	1671	aos,dos,...

Tabela 5.4: As primeiras 5 PP na geração de 50 PP de 1 sílaba.

O tempo médio de geração foi de 0.25131 segundos e com tempo médio por pseudopalavra de cerca de 5ms.

5.2 Resultados com o algoritmo Palavra Protótipo

Para o algoritmo Palavra protótipo, possíveis resultados para a palavra predefinida, "estudar", quando se pretende 100 PP.

	Pseudopalavras	OLD20	Dist1sub	Dist1	Dist2	Dist3	Lists
1	sultudar	2.95	0	0	1	58	suturar,aculturar,..
2	futudar	2.4	1	1	10	1399	futurar,autuar,...
3	esture	1.7	5	6	70	473	estere,estire,...
4	leitudar	2.8	0	0	4	50	estudar,leituga,...
5	esladar	2	0	0	28	429	deslaçar,enfadar,...

Tabela 5.5: As primeiras 5 PP através de derivações da palavra "estudar".

O tempo médio de geração foi de 4.4376 segundos e foi possível gerar as 100 PP derivadas de "estudar". O tempo de médio de geração por pseudopalavra foi de cerca 44.376 ms.

Outro algoritmo de nome de palavra_prot_2sil, gera PP de 2 sílabas. Este algoritmo é chamado dentro do algoritmo "Palavra Protótipo" quando é introduzida uma palavra de 2 sílabas. Possíveis resultados para a introdução da palavra, "porta" e com o intuito de serem geradas 1000 PP, serão mostrados apenas cinco resultados, 3 PP que mudassem a primeira sílaba e 2 PP que mudassem a 2ª sílaba.

	Pseudopalavras	OLD20	Dist1sub	Dist1	Dist2	Dist3	Lists
1	enta	1.3	4	14	272	2346	anta,benta,...
2	nista	1.55	8	9	131	1214	cista,dista,...
3	mita	1	18	27	397	2450	cita,dita,...
232	porme	1.8	4	4	113	1068	dorme,forme,...
236	porquei	1.95	0	1	24	168	porque,aparquei,...

Tabela 5.6: 5 derivações da palavra "porta".

O tempo de geração total foi de 5.6606 segundos e só foi possível gerar, no máximo, 242 PP parecidas a "porta". O tempo médio de geração por pseudopalavra foi de 23.39 ms.

O objetivo foi atingido, na medida em que é possível, em tempo real, gerar pseudopalavras pronunciáveis e com um tempo de geração (processamento) bastante rápido.

Capítulo 6

Conclusões e trabalho futuro

Foi possível obter um *corpus* lexical suficientemente grande, sem palavras compostas, sem estrangeirismos e sem siglas o que é altamente benéfico para a formação de pseudopalavras, já que tendo um bom alicerce, os algoritmos retornam bons resultados.

Também foi necessário criar um vocabulário de sílabas e um vocabulário de sílabas que não contém sílabas que sejam palavras no léxico, não contém dígrafos, nem palavras começadas por "nh" e "ç"'. Também foi criada uma bigrama de sílabas com diferentes parâmetros associados a cada par de sílabas, de modo a ser possível a geração de pseudopalavras.

Os algoritmos de geração de pseudopalavras foram concluídos e com o devido interface gráfico.

Os resultados, que foram apresentados no capítulo 5 mostraram-se satisfatórios, já que, foi possível criar pseudopalavras que podem ser lidas, seguindo as regras fonotáticas da língua e as mesmas são formadas pelos mesmo encontros silábicos encontrados no léxico.

Apesar dos resultados terem sido satisfatórios, não é possível afirmar, com certeza, que não exista nenhuma palavra escondida no meio das pseudopalavras, isto porque não foi feito nenhum algoritmo que verifique um possível lema das pseudopalavras geradas.

O primeiro algoritmo descrito foi o denominado de "Gerador de PP 1-10 sílabas", descrito em 4.1.1, dá origem a muitas pseudopalavras acabadas por consoante seguidas por i, tanto no singular como no plural, o que não é muito provável de acontecer na língua portuguesa, já que, as palavras acabadas dessa maneira, são geralmente formas verbais que indicam passado.

O segundo algoritmo foi o algoritmo "Palavra Protótipo", descrito em 4.1.2, que retorna pseudopalavras com uma sílaba diferente, em cada posição da palavra protótipo. Um melhoramento futuro passaria pela escolha, por parte do utilizador, do número de sílabas diferentes da palavra protótipo. Esse valor teria que ser maior do que 0 e menor ou igual ao número de sílabas dessa palavra protótipo. Caso o utilizador quisesse tantas sílabas diferentes como o número de sílabas da palavra protótipo, o algoritmo palavra protótipo chamaria o algoritmo "Gerador de 1-10 sílabas".

Não é feita uma verificação de plebeísmos, o que pode resultar, por vezes, em encontros silábicos que induzam a palavrões.

As verificações de singular e plural foram feitas nos casos mais simples, ou seja os algoritmos apenas verificam se as pseudopalavras acabam com a letra "s". Se não acabar com "s", adiciona-se o "s" e verifica-se se assim existe, se existir não é pseudopalavra.

O trabalho futuro passará pela criação de um algoritmo que permita verificar se a pseudopalavra gerada pode ser uma flexão válida do lema identificado mas que não exista no vocabulário (caso mais provável na flexão verbal). Esta verificação garante uma maior probabilidade de não existirem palavras entre as pseudopalavras. Este algoritmo também permitiria mostrar ao utilizador por forma de curiosidade nos resultados, um "pseudolema" para as PP geradas.

Outro objetivo futuro passará pelo desenvolvimento de todos estes algoritmos em C++ e interface com o utilizador em HTML, de modo a que os algoritmos sejam rápidos e que o gerador de PP funcione a partir de um *browser*.

Bibliografia

[1] Diana Raquel Silva de Sá Coutinho. Processamento fonológico de pseudopalavras linguisticamente motivadas em crianças com dislexia, 2014.

[2] Portal da Língua Portuguesa. Acordo ortográfico da língua portuguesa de 1990. *Online.] Accessed: http://www. portaldalinguaportuguesa. org/acordo. php*, 2014.

[3] Instituto de Linguística Teórica e Computacional. Lince - conversor para a nova ortografia. `http://www.portaldalinguaportuguesa.org/lince.php/` [Online: acesso a 1 de Dezembro, 2017].

[4] Maria Dias. *O papel da consciência fonológica nas dificuldades específicas de leitura e escrita (DELE): na perspetiva dos docentes do 1º CEB*. PhD thesis, 2013.

[5] Wouter Duyck, Timothy Desmet, Lieven PC Verbeke, and Marc Brysbaert. Wordgen: A tool for word selection and nonword generation in dutch, english, german, and french. *Behavior Research Methods, Instruments, & Computers*, 36(3):488–499, 2004.

[6] Emmanuel Keuleers and Marc Brysbaert. Wuggy: A multilingual pseudoword generator. *Behavior research methods*, 42(3):627–633, 2010.

[7] DA Medler and JR Binder. Mcword: An on-line orthographic database of the english language, 2005.

[8] Paulo Alexandre Rocha and Diana Santos. Cetempúblico: Um corpus de grandes dimensões de linguagem jornalística portuguesa. *quot; In Maria das Graças Volpe Nunes (ed) V Encontro para o processamento computacional da língua portuguesa escrita e falada (PROPOR 2000)(Atibaia SP 19-22 de Novembro de 2000) São Paulo: ICMC/USP*, 2000.

[9] L.; Mata L; Rosa M. Silva, I. ; Marques. Orientações curriculares para a educação pré-escolar, 2016.

[10] I. Sim-Sim. Desenvolvimento da linguagem, 1998.

[11] AP Soares, M Comesaña, A Iriarte, JJ Almeida, A Simões, A Costa, and J Machado. Procura-palavras (p-pal): A web application for a new european portuguese lexical database. In *Poster presented at the 17th meeting of European Society of Cognitive Psychology, San Sebastián, Spain*, 2011.

Apêndice A

Conjunto de todas as sílabas

a	beis	blés	brir	bál	car	chiu	clas	cons
ab	bel	blí	bris	bár	cas	cho	clau	cop
abs	bem	blói	briu	bárc	cau	chol	claus	cor
ac	ben	bo	bro	bás	caus	chon	cle	cos
ad	bens	bod	bron	bâ	caz	chor	clea	cou
ads	beo	boi	bros	bân	ce	chos	clec	cra
af	ber	bois	brou	bã	cea	chou	clei	crais
ag	berg	bol	bru	bães	cec	chu	clem	cral
ah	bes	bom	brum	bão	cei	chua	cleo	cram
ai	beu	bon	bruns	bãs	ceis	chui	clep	cran
ais	beus	bons	brup	bé	cel	chuis	cles	crar
al	bex	bop	brus	béis	cem	chum	cli	cras
am	bi	bor	brá	bél	cen	chun	clin	crau
an	bia	bos	brác	bém	ceo	chus	clip	craus
ao	big	bot	brás	béns	cep	chá	clis	craz
aos	bil	bou	brâ	bér	ceps	chás	clo	cre
ap	bim	boz	brân	bés	cer	chã	clos	crei
ar	bin	bra	brã	béu	ces	chão	clou	crem
arc	bins	brahm	brão	béus	ceu	chãos	clu	cren
as	bio	brai	brãos	bê	ceus	chãs	clui	crer
at	bir	brais	brãs	bên	cha	ché	cluis	cres
au	bis	bral	bré	bês	chai	chéis	clá	creu
aus	biu	bram	brés	bí	chais	chém	clás	cri
az	bla	bran	bréus	bís	chal	chés	cláu	crim
ba	blam	brar	brês	bó	cham	chéu	clâ	crins
bac	blan	bras	brí	bói	chan	chí	clâmp	crip
bad	blar	braz	bró	bóis	char	chões	clã	cris
bag	blas	bre	brôn	bós	chas	chú	clãs	cro
bah	ble	brech	brões	bô	chau	ci	clé	croi
bai	blei	brei	bser	bôs	che	cia	cléc	cror
bais	bles	breim	bsor	bões	chei	cie	clí	cros
bal	bli	breis	bu	bú	cheis	cil	clíp	crou
bam	blia	brem	bua	búl	chel	cim	clís	cru
ban	blin	bren	bue	bún	chem	cin	cló	crus
bap	blis	bres	bui	búr	chen	cio	clós	cruz
bar	blo	breu	buis	ca	cher	cir	cni	crá
bas	bloi	breus	bul	cac	ches	cis	co	crás
bau	blon	bri	bum	cai	cheu	ciu	coa	crâ
baus	blos	bria	bun	cais	chi	cius	coc	crã
baz	blou	bril	buns	cal	chil	cla	cog	crãs
be	blu	brim	bur	cam	chim	clam	coi	cré
bea	blá	brin	bus	can	chin	clamp	col	crés
bec	blás	brins	buz	caos	chins	clan	com	crê
bei	blé	brio	bá	cap	chis	clar	con	crês

crí cés daz dor drês dêu fal fleg fri
críp céu de dorm drí dí fam fleu fric
crís céus dea dos dró díc fan fli frim
cró cê dei dou dróis díp far flic frin
crói cên deis doz drões dís fas flir frins
crós cês del dra drú dó fau flo fro
crú cêu dem drac ds dói faus flor froi
csi cí den drai du dóis fax flou frol
ctó cím dens drais dua dól faz flu fron
cu cín deo dral duc dóp fe flua fros
cua cír dep dram dui dós fec flui frou
cui cís der dran dul dô fei fluo fru
cul có des drar dum dôn fel flá frui
cum cóg deu dras dun dõe felds flâ frus
cun cói deus drau duns dões fem flã frá
cuns cóis dex draus duo dú fen flé frás
cuo cóp dez dre dur dúc fer fló frâ
cur cór dhis drei dus dúl fes flú frân
cus cós di drem duz dún feu fo frães
cuz cô dia dren dá dús feus foi frão
cuí côm dic dres dác e fez fol frãos
cá côn dies drez dáf ec fi fom fré
cád côns dif dri dái ed fia fon frém
cál côr dig dria dál eh fic for fréns
cáp côs dil dril dás ei fif fos frê
cár cõe dim drim dâ eins fig fou frí
cás cões din drins dâm eis fil foz fró
cáu cú dins drio dân el fim fra frões
cáus cúm dio dris dã em fin frac fta
câ cún dir dro dão emp finc frag ftá
câm cúr dis droi dãos en fins fral fu
cân cús disp drol dãs ens fio fram fuc
cã da diu dron dé er fir fran fui
cãe dac diz drop déis ers fis frar fuis
cães dai do dros dél es fiz fras ful
cãi dais doa drou dém et fla frau fun
cão dal dog dru déns eu flac fre func
cãs dam doi drun dér eus flam frei fur
cé dan dois drá dés ex flan freis fus
céis dap dol drás déu ez flar frem fut
cél dar dom dráu déus fa flas fren fá
cém das don drân dê fac flau frer fác
cép dau dons drão dên fai fle fres fál
cér daus dop dré dês fais flec freu fár

fás
fáus
fâ
fân
fã
fães
fão
fãos
fãs
fé
fém
féns
fér
fés
féu
féus
fê
fí
fíl
fím
fín
fínc
fís
fó
fói
fór
fós
fô
fõe
fões
fú
fúc
fúl
fún
fúr
ga
gai
gais
gal
gam
gan
gar
gas
gau
gaz
ge
gea
gei
geis
gel
gem
gen
gens
geo
ger
gers
ges
geu
geus
gi
gia
gie
gil
gim
gin
gio
gip
gir
gis
giu
giz
gla
glai
glan
glas
glau
gle
gles
gli
glio
glis
glo
glos
glu
glus
glá
glân
glês
glí
glís
gló
glú
gno
gnos
gnu
gnus
gnó
gnós
go
goa
goi
gol
gon
gop
gor
gos
gou
goz
gra
grai
grais
gral
gram
gran
grar
gras
grau
graus
gre
grei
greis
grel
grem
gren
gres
gri
grim
grin
grir
gris
griu
gro
groi
gros
grou
grous
gru
grua
grun
grá
grâ
grân
grã
grão
grãos
grãs
gré
grés
grí
gró
grói
gu
gua
guai
guais
gual
guam
guan
guar
guas
gue
guei
gueis
guel
guem
guen
guer
gues
gueu
guez
gui
guia
guil
guim
guin
guins
guir
guis
guiu
guiz
gul
gum
gun
guns
guo
guos
guou
gur
gus
guá
guém
guéns
guê
guês
guí
guís
gá
gál
gár
gás
gáu
gâ
gâm
gân
gã
gão
gãos
gãs
gé
géis
gér
gés
gê
gên
gês
gí
gím
gín
gíp
gís
gó
gói
góis
gór
gós
gô
gôn
gõe
gões
gú
gúi
gún
gús
ha
hai
hal
ham
han
har
has
hau
haus
he
heb
hec
hei
heis
hel
hem
hen
hep
her
hes
heu
hi
hil
him
hin
hip
hir
his
hit
ho
hom
hon
hor
hos
hou
hu
hui
hum
hun
há
háp
hás
hã
hão
hé
hél
hér
hí
hís
hó
hós
hú
hún
i
ic
ig
ih
il
im
in
ins
io
ip
ir
is
ist
iu
iz
iô
ja
jac
jai
jais
jal
jam
jan
jar
jas
jau
jaus
jax
jaz
je
jea
jec
jei
jeis
jem
jen
jens
jes
ji
jim
jin
jis
jo
joa
joi
jon
jor
jos
jou
joz
ju
jul
jum
jun
juns
jur
jus
já
jás
jâm
jã
jães
jão
jãs
jé
jéc
jês
jí
jó
jói
jós
jô
jões
jú
la
lac
lag
lai
lais
lam
lan
lans
lap
lar
las
lau
laus
laz
le
lea
lec
lei
leis
lem
len
leo
lep
ler
les
leu
leus
lex
lez
lha
lhai
lhais
lhal
lham
lhan
lhar

lhas lip lér mem mur nab nhen niz néu
lhau lir lés men mus nac nhes no néus
lhaus lis léu mens muz nad nhez nob nê
lhe liu léus meo má naf nhi noc nên
lhei lius lê mer mál nai nhin noi nês
lheis lix lên mers már nais nhir nol nêu
lhem liz lês mes más nal nhis non ní
lhen lo lêu meu mâ nam nhiu nop ním
lher loa lí meus mân nan nho nor nín
lhes loi lím mez mã nap nhol nos nís
lheu lom lín mi mãe nar nhor nou nó
lhi lon líp mia mães nas nhos noz nói
lhis lor lís mic mão nau nhou nu nóis
lho los ló mig mãos naus nhoz nua nóp
lhor lou lói mil mãs naz nhu nue nór
lhos loui lós mim mé ne nhum nui nós
lhou lour lô min méis nea nhun nuis nô
lhu loz lôm mins mém nec nhuns nul nõe
lhum lu lôs mio méns nei nhá num nões
lhus lui lõe mip mér neis nhã nun nú
lhá luis lões mir més nel nhão nuns núl
lhã lum lú mis mê nem nhãs nuo nún
lhães lun lúm miu mêi nen nhé nup núp
lhão lup lúr mne mên neo nhéis nur nús
lhé lur ma mné mês nep nhés nus o
lhéu lus mad mo mí ner nhês ná oas
lhéus luz mag mog míg nes nhí nál ob
lhê lá mai moi míl neu nhó náp obs
lhês lác mais mol mín neus nhóis nár oc
lhí lás mal mom míp nex nhões nás of
lhís láu mam mon mís nez ni náu og
lhó lâ man mons mó nha nia nâ oh
lhós lâm mar mop mói nhai nic nâm oi
lhões lân mas mor móis nhais nig nân ois
li lã mau mos mór nhal nil nã ol
lia lães maus mou mós nham nim não om
lib lão max moz môn nhan nin nãos on
lic lãos maz mu mões nhar nins nãs op
lig lãs me muf mú nhas nio né or
lim lé mea mui múl nhe nir néc os
limp léc mec mul mún nhei nis néis ou
lin lém mei mum múr nheis niu nép ov
lins léns meis mun mús nhel nius nér ox
lio lép mel muns na nhem nix nés oz

pa
pac
pag
pai
pais
pal
pam
pan
par
part
pas
pau
paus
pav
paz
pe
pec
pei
peis
pel
pem
pen
pep
per
pers
pes
peu
peus
pez
pi
pia
pic
pig
pil
pim
pin
pins
pio
pir
pis
piu
pla
plai
plais
plam
plan
planc
plar
plas
plau
ple
plec
plei
pleis
plem
plen
ples
pleu
plex
pli
plia
plin
plis
plo
ploi
plos
plou
plu
plui
plum
plá
plás
plé
pléc
plêi
plên
plí
pló
plói
plô
plúm
pneu
pneus
po
poi
pois
pol
pom
pon
pons
pop
por
port
pos
pou
pra
prag
prai
pram
pran
prar
pras
praz
pre
preg
prei
prem
pren
pres
pri
pria
prin
prio
prir
pris
priu
pro
proc
prog
prol
pron
pros
prou
prous
pru
prá
prân
pré
prés
prí
prín
prís
pró
prós
prú
pseu
psi
psiu
pso
psí
psó
pte
pti
pto
pu
pug
pui
pul
pum
pun
punc
puns
pur
pus
puz
pá
pál
pár
pás
páu
pâ
pâm
pân
pãe
pães
pão
pé
péc
péis
pél
pép
pér
pérs
pés
péu
péus
pê
pên
pêu
pí
píc
pín
pís
pó
pói
pól
pór
pós
pô
pôn
pôr
pôs
põe
põem
pões
pú
púl
púr
pús
qa
qu
qua
quais
qual
quam
quan
quar
quart
quas
quaz
que
quea
quei
queis
quel
quem
quen
queo
quer
ques
quet
queu
queus
qui
quia
quid
quim
quin
quins
quio
quir
quis
quiu
quo
quoi
quos
quou
quá
quâ
quân
qué
quéc
quéis
quém
qués
quê
quês
quí
quín
quó
quói
quós
ra
rac
rad
rai
rais
ral
ram
ran
rap
rar
ras
rau
raus
rax
raz
re
rea
rec
rei
reis
rel
rem
remp
ren
reo
rep
rer
res
reu
reus
rex
rez
ri
ria
ric
rie
rif
rig
ril
rim
rin
rins
rio
rip
rir
ris
rit
riu
riz
ro
rob
rof
roi
rol
rom
ron
ros
rou
roz
rop
ror
rra
rrai
rral
rram
rran
rrar
rras
rre
rrea
rrec
rreg
rrei
rreis
rrel
rrem
rren
rreo
rrer
rres
rreu
rri
rria
rril
rrim
rrin
rrio
rrir
rris
rrit
rriu
rro
rroi
rrom
rron
rror
rros
rrou
rroz
rru
rrui
rrum
rrun
rrup
rrus
rrá
rrais
rrás
rrâ
rrân
rrã
rrão
rrãs
rré
rrê
rrên
rrí
rrít
rró
rrói
rrós
rrôm
rrões
ru
ruc
rui
ruis
rul
rum
run
runs
rup
rur
rus
rá
rác
rál
rár
rás
râ
râm

rân	san	som	ssio	subs	tac	tins	trei	trão
rã	sar	son	ssir	suc	tag	tio	treis	tré
rães	sas	sons	ssis	sul	tai	tip	trel	tréis
rão	sau	sor	ssiu	sump	tais	tir	trem	trê
rãos	saus	sos	sso	sun	tal	tis	tren	três
rãs	se	sou	ssoi	sur	tam	tiu	trens	trí
ré	sea	spi	ssol	sus	tan	tiz	treo	trín
réc	sec	spon	ssom	sá	tap	tlan	trep	tríp
réis	seg	spor	sson	sál	tar	tlas	tres	trís
rém	sei	sprin	ssons	sáu	tas	tle	treu	tró
réns	seis	spí	ssor	sâ	tats	tlim	treus	trói
rép	sel	ssa	ssos	sân	tau	tlo	tri	trói s
rés	sem	ssai	ssou	sâns	taus	tlos	tria	tróp
rét	sen	ssais	ssu	sã	taz	tlân	tric	trós
réu	seo	ssal	ssuc	são	tche	tlé	tril	trô
réus	sep	ssam	ssui	sãos	tchim	to	trim	trôn
rê	ser	ssan	ssuis	sãs	tché	toc	trin	trõe
rên	ses	ssar	ssul	sé	te	tog	trio	trões
rês	seu	ssas	ssum	sép	tea	toi	trip	tson
rí	seus	ssau	ssump	sér	teau	tol	trir	tsé
rín	sex	ssaz	ssun	sés	tec	tols	tris	tu
rís	shan	sse	ssuns	sê	tech	tom	triu	tua
rít	shi	ssea	ssur	sên	tei	ton	triz	tui
ró	si	ssec	ssus	sês	teis	tons	tro	tuis
rói	sia	sseg	ssá	sêx	tel	top	troi	tul
róis	sig	ssei	sséu	sí	tem	tor	trom	tum
róp	sil	sseis	ssâ	síg	temp	tos	tron	tun
rós	sim	ssel	ssão	síl	ten	tou	trons	tungs
rô	sin	ssem	ssé	sím	tens	tra	trop	tuns
rôm	sins	ssen	sséis	sín	teo	trac	tros	tuo
rôn	sio	sseo	ssép	sís	ter	trai	trou	tur
rõe	sir	ssep	ssên	só	ters	trair	troz	tus
rões	sis	sser	ssí	sói	tes	trais	tru	tut
rú	siu	sses	ssín	sóis	teu	tral	truc	tá
rún	sius	ssex	ssís	sór	teus	tram	trui	tác
rúr	sni	ssez	ssó	sós	tex	tran	trun	tál
rús	sno	ssi	ssóis	sô	tez	trans	trus	táp
sa	so	ssia	ssõe	sôr	ti	trap	truz	tár
sac	sob	ssig	ssões	sõe	tia	trar	trá	tárc
sai	sobs	ssil	sti	sões	ties	tras	trác	tás
sais	soi	ssim	stre	sú	tig	trau	trás	táu
sal	sois	ssimp	su	súb	til	traz	trâ	tâ
salz	sol	ssin	sua	súl	tim	tre	trân	tâm
sam	sols	ssins	sub	ta	tin	trea	trâns	tân

tã
tãe
tães
tão
tãos
tãs
té
téc
téis
tém
téns
tér
tés
téu
téus
tê
têm
tên
tês
têu
têx
tí
tím
típ
tís
tó
tóc
tói
tóis
tóp
tós
tô
tôm
tõe
tões
tú
túr
u
uh
ui
uis
ul
um
un
uns
ups
ur
us
uz
uís
va
vai
vais
val
vam
van
var
vas
vau
vaus
vaz
ve
vea
vec
vei
veis
vel
vem
ven
vens
veo
ver
vers
ves
veu
vez
vi
via
vic
vil
vim
vin
vins
vio
vir
vis
viu
viz
vla
vo
voa
vod
voi
vol
von
vor
vos
vou
voz
vra
vrai
vrais
vral
vram
vran
vrar
vras
vre
vrei
vrem
vres
vri
vro
vros
vrou
vrá
vrál
vrão
vró
vrões
vu
vul
vur
vá
vál
vár
vás
vâ
vân
vã
vães
vão
vãos
vãs
vé
véis
vém
véns
vér
vés
véu
véus
vê
vêm
vên
vês
ví
víl
vín
vír
vís
vó
vói
vól
vór
vós
vô
vôs
võe
vões
xa
xac
xai
xais
xal
xam
xan
xar
xas
xau
xaus
xe
xei
xeis
xel
xem
xen
xeo
xer
xes
xeu
xi
xia
xil
xim
xin
xins
xio
xir
xis
xo
xoi
xor
xos
xou
xu
xul
xun
xá
xár
xás
xâ
xã
xão
xé
xéis
xér
xés
xí
xís
xó
xói
xór
xós
xô
xõe
xões
xú
xún
za
zai
zais
zal
zam
zan
zar
zas
ze
zea
zei
zeis
zel
zem
zen
zeo
zer
zes
zeu
zeug
zeus
zi
zia
zil
zim
zin
zio
zir
zis
ziu
zo
zoi
zol
zom
zon
zoo
zor
zos
zou
zu
zuis
zul
zum
zun
zuns
zur
zá
zás
zâ
zâm
zân
zã
zão
zãs
zé
zéis
zém
zéns
zés
zê
zês
zí
zín
zó
zói
zóis
zós
zôi
zões
zú
à
às
á
ál
ár
árc
ás
áu
áus
â
âm
ân
ã
ães
ão
ãos
ãs
ça
çai
çais
çal
çam
çan
çar
ças
ço
çoi
çol
çons
çor
ços
çou
çu
çul
çuz
çá
çás
çâ
çã
ção
ções
çãos
çãs
ró
çóis
çõe
ções
çú
é
éis
ér
és
ét
ê
êi
êm
ên
ês
êu
êx
í
íc
íg
íl
ím
ín
ís
ó
ób
óc
ói
óis
óp
ór
ós
ô
ôn
ôs
õe
ões
ú
úl
ún
úr
ús

Apêndice B

Pseudopalavras de 3 sílabas e OLD20

Pseudopalavras	OLD20	Pseudopalavras	OLD20
escoutos	2	firado	1,15
baldoa	1,7	despeitais	2,25
cobarca	1,95	beijanda	2,45
enfrasta	2,55	angloba	2,3
canhantes	2,3	manguelhos	3,05
forcava	1,6	nigrossa	2,9
pivexo	2,8	lapinhal	2
arminga	1,95	quentia	1,95
gincani	2,75	mistina	1,95
trambera	2,85	buero	1,85
ninantal	3	sanguejo	2,75
placanta	2,65	mériza	1,95
lagrinhos	1,9	rabispal	2,65
dessoural	2,75	gaspara	1,85
horrolha	3	mulia	1,7
baldeire	2,4	tripato	2,1
baorem	1,95	nocturpei	3,45
precia	1,65	trouxelês	2,95
primornou	3,15	jureci	2,35
pelega	1,8	retoura	1,95
braderi	2,45	rascanta	2,55
pampino	1,95	cipita	1,95
sessenti	2,35	dorzidas	1,9
varreçam	2,2	buriza	1,95
piorde	1,9	omoro	1,85
mateiral	2	guilhora	2,8
pródica	1,8	tímiga	1,9
poceita	2,1	cegueja	2,4
ódicar	2	moscarte	2,8
hemissão	2,35	fraseca	2
ozoa	1,95	conspita	1,95
herdardo	1,95	transduca	2,85
sabriga	1,9	poisagra	2,45
teimolo	2,45	simonto	2,45
pangora	1,9	tetiza	1,9
desproa	2	moscotar	2,95
baudere	2,85	orvaler	2,75
sincado	1,85	grasnito	2,35
chonica	2,3	reinazi	2,25
guloja	2,55	moldanda	1,95
encorda	1,9	xadreja	2,65
protoco	2,3	funçado	1,95
henrides	2,9	desgate	1,8
marceslau	3,15	vertesta	2,4

Pseudopalavras	OLD20	Pseudopalavras	OLD20
hispanhei	3,4	variza	1,8
telhaco	2,25	mieira	1,6
arritmo	2,1	impoça	1,9
roada	1,05	transcolhe	3,1
desvanta	1,9	evora	1,75
sopeita	2,05	guiatu	2,05
encarpa	1,75	perneirem	2,5
mesteira	1,75	discote	1,95
trigodões	3,4	nónua	2,15
gambora	2,1	colmeixas	2,75
penidos	1,7	épode	1,95
prolori	2,6	cortesta	1,9
mexelês	2,85	provali	2,15
fartuma	2,15	piquebra	2,75
lisboja	2,8	rotango	2
catarda	1,9	fenome	2,2
comtili	2,85	flutuja	2,35
maltono	2,45	besanda	1,95
incando	1,8	abandas	1,75
castolo	1,95	sorriscas	2,7
charanta	2,6	alperta	1,85
folida	1,75	mónita	1,9
roldago	2,5	judaia	1,85
airaco	1,95	malhofa	1,9
refeudar	2,8	gozantes	2
prostifa	2,7	cantire	2
neutria	1,9	traçante	2,2
bolseios	2,3	sainega	2,6
tonalda	2,15	hostili	2,7
matofo	2,05	mureci	2,05
trogloba	3,75	samuseu	2,9
biesses	1,95	rombura	2,45
fífina	2,05	guilhorais	3,85
reinte	1,85	fatalga	2,5
abrilha	1,95	frenove	2,45
senio	1,55	louisira	2,85
consanca	2,8	pristili	2,85
berente	1,95	hosarões	2,95
seturei	1,9	fragodes	2,55
tintanol	2,8	inqueça	2,7
braviza	2	deforta	1,9
chadora	1,95	áurece	2,45
sesmeidas	2,8	lisure	2,05
gritarses	2,5	párolo	1,85

Pseudopalavras	OLD20	Pseudopalavras	OLD20
enundam	1,95	infreram	2,65
vitila	1,85	cínima	1,9
carreda	1,65	típinal	2,85
recursa	1,8	lazeitão	2,85
caieta	1,85	hiberba	2,45
pavina	1,75	engoe	1,8
belzeja	2,7	triplaca	2,4
zipassa	2,85	singrada	1,85
hindusto	2,7	manascem	2,55
gomato	1,95	forcede	2,3
vistoou	2,3	coarque	1,95
judadei	2,85	pacota	1,8
ousegui	2,85	prospecta	2,25
tirasca	2,3	besura	1,85
terrigue	2,9	sambece	2,95
mortago	2,15	progesto	2,1
preclusos	2,6	ousastre	2,65
torvales	2,8	quebreirais	3,55
antado	1,65	prosaba	2,2
veleirem	2,4	rãzigua	3,2
símplica	2,6	beirulho	2,95
gozague	2,9	pãozinha	1,95
consteri	2,25	esbeira	1,95
renderna	1,95	barrasa	1,65
prolisa	1,95	chícharéis	3,7
fentomo	2,75	cadadão	1,95
macara	1,6	insura	1,85
garnima	2,45	dosio	2
meteçam	1,9	enchega	1,95
sacrava	1,75	fúrcuta	2,75
piejo	1,9	papalpa	2,1
canturi	2,05	travita	1,85
mácuba	2,5	mundica	2
dumiti	2,4	cutiva	1,85
sucura	1,8	meterna	1,8
archoa	2	rançoar	2,4
compinzais	2,9	tiverna	1,95
pocia	1,8	sonida	1,9
versaba	1,85	revulga	2,35
empito	1,8	frestiva	2,15
largorar	2,65	progela	1,9
segrece	2,4	desfulo	2,1
tralhantam	3,45	garraia	1,75
fizernas	2,6	sisuras	1,95

Pseudopalavras	OLD20	Pseudopalavras	OLD20
bimbares	2,55	toqueci	2,45
foqueceu	2,85	escordo	1,7
firabal	2,85	afessa	1,95
sadisca	2,3	pingueiro	2,75
botancha	2,85	oirada	1,35
adica	1,6	translia	1,9
cistiga	1,95	elvidi	2,1
pentela	1,9	visumi	2,6
essenso	2,45	traqueiras	2,3
durarma	1,9	queirava	1,85
febresa	1,9	chiança	1,85
someia	1,75	craveita	2,5
bímarães	3	inglotas	2,85
nortago	2,65	nozima	2,1
bombalha	2,7	fidece	2,7
provarda	1,85	vasola	2
hégio	1,9	coutiva	1,95
moreitar	2,65	centeri	2,2
garouca	2,15	épie	1,9
deseste	1,8	côncado	1,95
trissola	2,9	ernesci	2,9
bijuve	3	subsexo	2,8
xácado	2	apolis	1,9
prédita	1,9	frigica	2,35
ciendro	2,5	hipiste	2,9
bargancham	3,45	engule	1,85
praiendi	2,75	graçoa	1,9
mourena	2	melgarra	2,95
musgueirões	3,95	esquire	1,85
galhios	1,95	linhices	2,85
enxalta	1,95	abruça	1,95
obrande	1,85	gritio	1,95
nimiti	2,35	princede	2,75
troveita	2,6	chovenceu	3,4
choupaga	2,55	feiibi	2,9
morfixa	2,3	destranger	2,95
fartastro	2,95	reinastro	2,95
jantandra	2,85	caldio	1,8
tasquilo	2,5	alhio	1,85
chumbeio	2,65	ergoli	2,45
gorjeita	2,35	saltiça	2
oferi	1,9	frutalei	3
piasma	1,85	eurani	2,65
carosa	1,55	margali	2,45

Pseudopalavras	OLD20	Pseudopalavras	OLD20
chegarça	1,95	talistas	1,8
colesta	1,8	cachuli	2,8
pútrirão	2,85	caiadi	1,9
froixame	3,05	cartino	1,95
reterva	1,9	cabouçou	2,65
manjada	1,8	branduzi	2,85
lúriza	2,15	basbara	1,85
gestudais	2,7	oponga	1,95
désseca	2,6	ternua	1,9
clériza	2,7	lojismos	2,7
tranquiza	2,45	brigare	1,85
guionei	2,7	soltarda	1,85
roraiva	2,25	perfia	1,7
ferrilha	2	redestro	2,8
ferata	2	blogosta	2,95
suinda	1,95	empete	1,8
afruí	2,4	ecosta	1,8
excreve	2,05	substruo	2,9
laminis	1,95	farneiza	2,9
glaucono	3,25	sístoo	2,7
pinedo	1,95	nictorre	3
caristo	1,9	gelabo	1,95
satismos	2,05	marcobro	2,9
fraseja	2,05	ralheses	2,6
poline	2	cimiti	2,15
concepto	1,9	poiside	2,75
caucheira	2,55	pezinha	1,6
caracção	2,2	anzotes	2,4
sóbone	2,85	fulgula	2,5
poucari	2,05	calhoa	1,8
daguescem	3,4	cemento	1,75
perdeces	1,95	roletrar	2,55
pragmata	2,8	vestirpe	2,1
herante	1,95	peixovais	3,4
bondadei	2,7	cacua	1,8
privedo	1,95	boralha	1,95
fracora	2,35	vinhala	1,9
vulgardes	2,65	ventoque	2,8
prostranca	2,8	castita	1,75
proconta	2,7	brinquie	2,45
dorsalgar	3,6	plistonos	3
euritmo	2,7	aldolo	2
censuca	2,35	substanga	3,15
lingualo	2,6	lectica	1,95

Pseudopalavras	OLD20	Pseudopalavras	OLD20
roubanquem	3,75	báltiram	2,85
tingiza	1,85	fareci	1,8
polegi	2,1	cóliza	1,95
roufere	2,8	baboca	1,85
trimestes	2,5	cerasca	2,4
batesta	1,85	proende	1,9
plebiscas	3	xiloa	1,95
septica	1,95	equire	1,95
mapira	2	jainiscos	3,25
memolo	2	leitota	1,85
vínhala	2,85	charlançou	3,75
fitante	1,95	trofolem	3
fracava	2	rédia	1,75
tramate	1,95	comboro	1,9
pulcrito	2,9	blástua	2,9
peturi	2,6	penua	1,65
coiteio	1,95	amalis	1,95
mostrastam	2,7	víncusa	2,9
subculca	3,35	lacorre	2
baface	2,6	neutralto	2,95
efeie	2	encursa	1,9
jeitorei	2,75	monforta	2,45
seiscendi	3,15	voterra	2,3
gazula	1,85	singrande	2,7
aroupa	1,85	oiteira	1,85
livora	1,85	incerca	1,95
vistope	2,55	finante	2
floreitar	2,45	agula	1,4
plebispos	3,4	mocio	1,7
luzigua	2,7	mercanta	2,1
zibera	1,9	henderi	2
teguça	2,5	barreca	1,7
pireci	1,9	lufase	2
lambalho	2,85	rampeja	2,3
lavardar	2,1	cursistir	2,8
sirguiou	3,2	subfazei	3,3
igreda	1,9	pudernos	2,3
bufandro	2,8	virarma	1,9
ascosa	1,95	manguala	2,3
mezunda	2,55	dispunem	2,3
morboga	2,6	numinho	1,75
pedrece	2,2	mucharel	2,85
râgueses	2,8	jupio	1,95
refregão	2,4	gambida	1,95

Pseudopalavras	OLD20	Pseudopalavras	OLD20
espingar	1,9	chapero	2
iadi	1,95	pectia	1,95
jogança	2,5	sérgiva	2,6
romeiga	2,15	franguida	2,8
lebrasa	2,05	fartolo	2,3
lancheiros	2,3	esfete	1,95
chimpandir	3,85	cromona	2,05
vexarou	2,25	leriza	1,95
famina	1,8	framboli	3,4
fereita	2	cártava	1,95
bicheios	2,45	dolentar	2,4
conculo	2	laurendou	2,8
predonei	3	prostada	1,85
tempicão	2,8	zigueda	2,95
mastoico	2,8	hemostra	2,6
baixala	1,8	chuchuli	3,2
tutura	1,8	marquie	1,9
lambrigais	2,9	uzbele	2,75
obstruí	2,05	manume	2
muriza	2	alfica	2
orlante	1,95	pórfica	2,25
castrasei	2,9	leiona	1,9
aztela	1,9	mineias	1,95
chadoca	2,6	encrusta	2,35
prepora	1,95	vendiga	1,8
mezelo	2,1	brevivi	2,6
zurita	2	fincare	1,8
mentela	2	forcia	1,9
vivanta	2,5	tintua	1,9
prestiza	2,4	saudera	1,95
pintarso	2,2	letrapa	2,1
trançantes	2,85	grunhala	2,9
sirani	2,3	donjura	2,2
puranos	1,95	sevence	2,65
serestar	2,4	curdiste	2,4
olfactor	2,6	isqueima	2,75
rebriga	2	predictos	2,25
sonila	2	bórirei	2,7
cúspira	2,15	linhotes	2,35
bedula	1,95	mirmios	2,55
perandou	2,1	requindo	1,95
tressuís	3,45	júdinos	2,6
clorosa	1,8	efama	1,9
condua	1,85	giestou	2,1

Pseudopalavras	OLD20	Pseudopalavras	OLD20
deixano	1,9	gotempla	3,25
luteri	2,15	fechega	2,4
picanta	1,9	caçoilos	2,5
cheirora	2,25	dementa	1,85
porrega	1,95	lisonrou	2,9
melgado	1,85	extaque	2,35
desaura	1,95	punçosas	2,85
jantarda	1,85	vidrata	1,8
belchichas	3,4	trezentam	2,7
mastoco	2,5	bífitas	2,45
fueiras	1,75	verdaluz	3,25
gardua	2	pilia	1,8
opanhóis	3,2	incompra	2,55
jacarna	2,15	nulapa	2,05
prelectro	2,8	mutufas	2,25
toartar	1,95	bicontro	2,9
lápata	2,5	cânove	2,6
faceitas	1,8	lareita	1,95
filmiza	2,45	vírguda	2,8
síriza	2,15	bancari	1,9
sempinhei	2,9	pantacos	1,95
molata	1,85	pernalda	2,3
trovelei	2,75	bigara	1,8
toarte	1,9	rinova	1,9
pãoziguar	3,85	galguida	2,6
parango	1,95	mirasques	2,95
otera	1,9	trauliza	2,95
apaica	1,95	tiaçam	1,95
sinado	1,8	bauniza	2,45
meiguitos	2,85	auganha	2,35
cénize	2,85	flirtanca	3,4
fidalgue	2,55	duzendo	2,2
achofra	2,65	pineas	1,75
cáfica	1,9	camerge	2,8
optima	1,9	publio	2,05
ondeixas	2,45	bazoei	2,25
pandere	2,15	emplastos	2,85
cofunda	1,95	laurida	1,9
genome	2,1	tomendes	2,4
trepasto	2,3	ulcedem	2,75
fertimo	2,3	jogarra	1,9
sorterres	2,9	sequeci	2,35
recristos	2,55	empaste	1,8
relvação	1,9	inquia	1,9

Pseudopalavras	OLD20	Pseudopalavras	OLD20
pacentra	2,6	taibi	2
cheletra	2,85	burloca	2,35
adrinha	1,9	boquipa	2,7
queiroa	1,9	valiza	1,7
ceiona	2	industo	1,9
imargue	2,5	nadangam	2,8
bolchega	2,9	clamoro	1,95
modavas	1,85	gandura	1,85
cábrenha	2,8	pessina	1,95
dulçaima	3,75	houverte	2,4
gasismos	2,25	mochicha	2,45
dignante	2,45	prazinhar	2,6
vilmenta	2,65	queimascos	2,85
brilhanta	2,3	âncorre	2
tripina	1,95	filtrançam	3,2
creriza	2,6	vaguiçar	2,85
logotou	2,75	argueira	1,9
golabo	2	futriba	2,55
mestranca	2,15	armismos	2,6
almassam	2,55	musinai	2,35
burliste	2,6	fincanta	2,55
clamode	2,6	grumere	2,8
ptologa	2,65	puludi	2,45
jotia	1,9	ontoa	1,95
preonei	2,55	valgueirar	2,8
sénetra	2,6	lázinhei	3,45
foguico	2,7	esmaiar	1,9
ervora	1,95	eunura	1,95
dextroa	2,15	obumbas	2,5
somancar	2,85	himani	2,7
viantei	2,3	moiraço	2,05
prenderdes	2,7	liguerras	2,85
pôntino	2,7	centiga	1,95
famude	2,3	ileci	2,45
desacções	2,55	gasnece	2,85
expreça	2,35	sarrondou	2,9
antropo	2,85	trâmina	2,35
chazeitão	3,05	boicorri	2,9
vergajar	2,45	roubarra	2,2
pedernas	2,3	passio	1,75
mosteimar	3	acari	2
baixardes	2,4	curvere	2,4
curvardo	1,9	bolanço	1,95
piantar	1,9	regrande	2,3

Pseudopalavras	OLD20	Pseudopalavras	OLD20
apente	1,55	bebescer	2,75
macanha	1,85	sangrinha	2,45
dríado	2	tâmine	2,25
calação	1,6	sobesco	2,8
taumata	2,95	inforiu	2,4
ousancam	2,8	turfeite	2,95
fugaça	1,9	planala	1,95
duzenses	2,65	polistar	2,25
latima	1,75	áladar	2
luxame	2,3	bechacho	2,9
termali	2,7	incontas	2,35
húngalei	3,3	pregora	1,9
afronham	2,35	dicanda	1,95
medato	1,9	algura	1,75
marrica	1,9	nóniza	2,3
apulo	1,65	faduncas	2,55
seriza	1,9	albuca	2
bitono	2,3	urtino	1,9
constridor	2,85	brincarta	2,15
aqueta	1,5	intante	1,85
entonga	2,2	teclismo	2,75
charoca	1,85	etelmo	2,4
florintos	2,7	rejua	1,85
repola	1,85	ossina	1,95
esola	1,6	recaptos	1,95
leiloja	2,3	dessenti	2
impanças	2,8	fleumana	2,95
suniste	1,9	reinada	1,65
disponta	1,85	inconstrói	3,5
desfragmas	2,95	acroma	1,9
borguistão	3,75	leucio	2,6
emprano	2,25	gastrosar	3
transfunda	2,55	dobarro	2,45
coiona	1,85	acresça	1,95
fagurões	2,7	púnhala	2,8
virardo	1,85	ducarra	2,6
rédena	2,5	câmana	1,95
trompeli	2,75	chocora	1,95
autardar	2,4	panora	1,85
palhoa	1,8	transformers	2,45
odata	1,95	sácubais	2,95
subvoos	2,75	claustronou	3,95
mourajo	2,4	consenta	1,7
arritmos	2,2	fabriza	2