

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE LETRAS
PROGRAMA DE PÓS-GRADUAÇÃO EM LETRAS
ÁREA DE CONCENTRAÇÃO: LINGÜÍSTICA APLICADA**

CELSO AUGUSTO NUNES DA CONCEIÇÃO

**UM ALGORITMO ILUSTRATIVO SOBRE O PLURAL NO PB COMO
CONTRIBUIÇÃO PARA MORFOLOGIA COMPUTACIONAL:
CAPACIDADES, LIMITAÇÕES E PERSPECTIVAS**

Porto Alegre, setembro de 2009.

CELSO AUGUSTO NUNES DA CONCEIÇÃO

**UM ALGORITMO ILUSTRATIVO SOBRE O PLURAL NO PB COMO
CONTRIBUIÇÃO PARA MORFOLOGIA COMPUTACIONAL:
CAPACIDADES, LIMITAÇÕES E PERSPECTIVAS**

Tese apresentada como requisito para obtenção do grau de Doutor, pelo Programa de Pós-Graduação da Faculdade de Letras da Pontifícia Universidade Católica do Rio Grande do Sul.

Prof. Dr. Jorge Campos da Costa
Orientador

Porto Alegre, setembro de 2009.

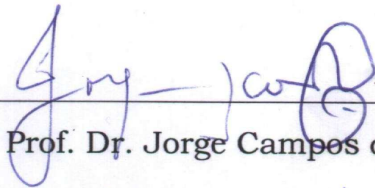
CELSO AUGUSTO NUNES DA CONCEIÇÃO

UM ALGORITMO ILUSTRATIVO SOBRE O PLURAL NO PB COMO CONTRIBUIÇÃO PARA MORFOLOGIA COMPUTACIONAL: CAPACIDADES, LIMITAÇÕES E PERSPECTIVAS

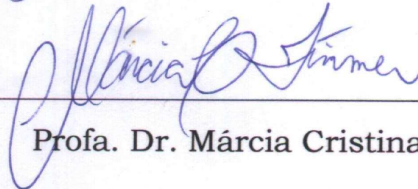
Tese apresentada como requisito para obtenção do grau de Doutor, pelo Programa de Pós-Graduação em Letras da Faculdade de Letras da Pontifícia Universidade Católica do Rio Grande do Sul.

Aprovada em 27 de março de 2009

BANCA EXAMINADORA:



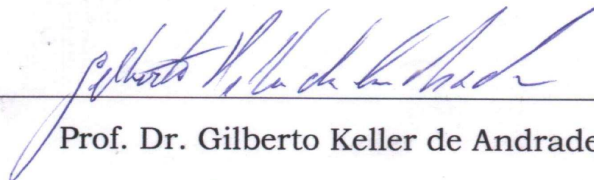
Prof. Dr. Jorge Campos da Costa - PUCRS



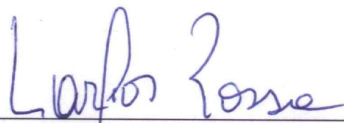
Profa. Dr. Márcia Cristina Zimmer - UCPel



Profa. Dr. Sabrina Pereira Pereira - UFRGS



Prof. Dr. Gilberto Keller de Andrade - PUCRS



Prof. Dr. Carlos Ricardo Pires Rosa - PUCRS

Dedico esta tese

Ao meu grupo de neurônios
especializados, que sempre estiveram
comigo em todas as horas de que mais
precisei.

E ao meu amor, Elizane Galvani
pela paciência inesgotável nas várias
ausências, pelo amor, pelo carinho,
compreensão e apoio durante toda essa
trajetória.

AGRADECIMENTOS

Ao professor Jorge Campos, amigo e hoje orientador desta tese, pela atenção, apoio, incentivo e, principalmente, pela acolhida em um momento crítico da caminhada acadêmica: a mudança estrutural do trabalho para a qualificação; também pela sua paciência em tolerar meu ritmo intempestivo e por sempre confiar em minha responsabilidade.

À professora Dra. Ana Mello, coordenadora do PPG, pelo apoio e pela compreensão no momento mais difícil da trajetória acadêmica: prazos. E também por sua paciência e confiança.

À Secretária do PPG, Isabel e Mara, pelo acompanhamento simpático, paciente e compreensivo das *nuances* acadêmico-administrativas.

Ao prof. Dr. José Marcelino Poersh, ex-orientador, amigo, e grande incentivador para a interface Lingüística e Computação.

À professora e amiga Ana Ibaños, pelo empréstimo e indicação de livros para ampliar os fundamentos na versão final desta tese.

Ao amigo Ricardo Bomfim, pelas aulas incansáveis de programação em IA.

À Alessandra Dahmer e Everton J. Silva pelo trabalho conjunto na elaboração do PluralRNA.

Aos meus filhos e enteados, pela ausência sentida nesses anos de estudos.

À minha mãe, pela vontade de viver e força em relação à sua enfermidade, principalmente pela espera na finalização desta tese.

À minha irmã, por tentar me deixar tranquilo quando da impossibilidade de compartilhar a total atenção com minha mãe.

A todos os meus amigos que compreenderam o meu distanciamento e afastamento.

E à família, que sem ela eu não teria o apoio e a motivação para concluir este trabalho de forma menos traumática.

SUMÁRIO

1 INTRODUÇÃO	15
2 SOBRE O PLURAL NA MORFOLOGIA LINGÜÍSTICA E O NÚMERO EM PORTUGUÊS	19
2.1 O ESTUDO DO PLURAL NA MORFOLOGIA LINGÜÍSTICA	20
2.1.1 A regularidade	23
2.1.2 As irregularidades	23
2.1.3 O processo de pluralização no Português Brasileiro	24
2.2 O NÚMERO EM PORTUGUÊS	26
2.2.1 Regras Descritivas	28
2.2.2 Regras Normativas	32
2.3 PROBLEMAS DO NÚMERO EM PORTUGUÊS	35
3 NA INTERFACE LINGÜÍSTICA/COMPUTAÇÃO	38
3.1 LINGÜÍSTICA COMPUTACIONAL	38
3.2 MORFOLOGIA COMPUTACIONAL	43
3.3 MODELAGEM COMPUTACIONAL	45
3.3.1 Das Ciências Cognitivas à Inteligência Artificial (IA)	46
3.3.2 Algoritmo em IA	49
3.3.3 O Algoritmo em linguagem natural e ilustrativo	58
4 SOBRE O COMPORTAMENTO DO NÚMERO EM PORTUGUÊS E O ALGORITMO COMPUTACIONAL	63

	8
4.1 <i>CORPUS</i>	63
4.2 DADOS	64
4.3 APLICAÇÃO METODOLÓGICA DO MODELO COMPUTACIONAL	67
4.4 IMPLEMENTAÇÃO DO MODELO	71
4.5 DESCRIÇÃO DO PROTÓTIPO	74
4.6 APLICAÇÃO DO <i>CORPUS/DADOS</i>	81
4.6.1 Treinamento <i>versus</i> aprendizado	81
4.6.2 Simulação de uso do PluralRNA	83
4.7 ANÁLISE DOS RESULTADOS: capacidades, limitações e perspectivas	92
5 CONCLUSÃO	95
REFERÊNCIAS	100
APÊNDICE A – Diagrama ER da base de conhecimento	104
APÊNDICE B – Algoritmos: Separar Sílabas e Identificar a Sílabas Tônica	105
APÊNDICE C – Algoritmo do funcionamento do PluralRNA	114

RESUMO

Esta tese é uma aplicação lingüístico-computacional baseada em Redes Neurais Artificiais (RNA) com um algoritmo ilustrativo sobre o plural na Morfologia do Português Brasileiro (PB): PluralRNA. Este é um *software* criado com o objetivo de fazer com que a máquina depreenda regras de flexão de número a partir da inserção de pares lexicais singular/plural. Essa depreensão acontece na fase de treinamento depois de a base de dados da rede estar devidamente provida desses pares. A partir daí, digitam-se outras palavras no singular para que o programa aplique a desinência depreendida pelo processamento. A seguir, analisa-se o potencial de aprendizagem do programa para constatar as suas capacidades e limitações e sugerir perspectivas, pretendendo assim contribuir cientificamente para a Morfologia Computacional do PB. O resultado é satisfatório na medida em que, por um lado, as generalizações das regras depreendidas são percebidas e cotejadas com o léxico padrão, tendo, na maioria dos casos, correspondência com normatização; por outro, alguns ciclos de treinamento reforçam a depreensão das desinências e alteram, em raríssimos casos, aquelas que já eram dadas como certas pela correlação gramatical. Portanto, os pares lexicais e a aplicabilidade do PluralRNA oferecem material suficiente para a continuidade dos estudos do processamento flexional de número para a Morfologia Computacional do PB.

Palavras-chave: Morfologia. Morfologia computacional. Algoritmo. Plural.

ABSTRACT

This dissertation is a linguistic-computational application based on Artificial Neural Nets (ANN) with an illustrative algorithm on the plural in the Morphology of Brazilian Portuguese (BP): PluralRNA. This is a software created with the objective of making the machine learn rules of number flexing by inserting lexical pairs singular/plural. This learning takes place in the net training phase as soon as the net data base is properly provided with these pairs. By now, other words are typed in the singular so that the program applies the ending learnt by the process. The next step is to analyze the program's learning potential in order to note its capacities and limitations and suggest perspectives aiming at the scientific contribution to the Computational Morphology of BP. The result is satisfactory as generalizations of inferred rules are perceived and compared to the standard lexicon, having, in most cases, correspondence with normalization. On the other hand, some training cycles reinforcing the plural learning alter, in very rare cases, those which were already taken as right according to grammatical correlation. Therefore, the lexical pairs and the applicability of PluralRNA provide enough material for the Brazilian Portuguese computational morphology studies on flexing and number processing.

Key words: Morphology, Computational morphology, Algorithm, Plural.

LISTA DE SIGLAS E ABREVIATURAS

IA	Inteligência Artificial
LMCPC	Léxico Multifuncional Computadorizado do Português Contemporâneo
MLP	<i>Multilayer Perceptron</i>
NGB	Normas gramaticais brasileiras
PB	Português Brasileiro
BP	Brazilian Portuguese
PE	Português Europeu
PLN	Processamento da Linguagem Natural
RAD	<i>Rapid Application Development</i> ou Desenvolvimento Rápido de Aplicações
RNA	Rede Neural Artificial

LISTA DE FIGURAS

Figura 1:	MorDebe, uma Base de Dados Morfológicos de Português – resultado 1.....	44
Figura 2:	MorDebe, uma Base de Dados Morfológicos de Português – resultado 2.....	44
Figura 3:	Modelo simulado de neurônio	51
Figura 4:	Modelo de McCulloch e Pitts (1943)	52
Figura 5:	O Perceptron de Rosenblat (1957)	52
Figura 6:	O Perceptron de Rosenblat (1957) e o neurônio biológico.....	53
Figura 7:	Grafo arquitetural da MLP.....	69
Figura 8:	Representação dos neurônios da camada de saída.....	69
Figura 9:	Tela de configuração da rede neural.....	75
Figura 10:	Tela do conjunto de entrada (singular x plural).....	77
Figura 11:	Tela de treinamento da rede MLP.....	78
Figura 12:	Tela de <i>log</i> do treinamento da rede MLP.....	79
Figura 13:	Tela de teste da rede MLP.....	79
Figura 14:	Tela principal do PluralRNA.....	80
Figura 15:	Tela de simulação: apresentação de erro.....	82
Figura 16:	Tela de simulação: apresentação de acerto.....	83
Figura 17:	Tela de Treinamento com 200 e 1200 ciclos.....	85
Figura 18:	Tela de digitação das mesmas palavras que entraram com seus pares no plural.....	90
Figura 19:	Tela de digitação de palavras diferentes da base de pares lexicais.....	90

LISTA DE QUADROS

Quadro 1: A formação do plural dos nomes, em inglês.....	21
Quadro 2: A formação do plural dos nomes, em português.....	25
Quadro 3: Estudos lingüístico-computacionais.....	41
Quadro 4: Algoritmo ilustrativo do PluralRNA.....	60
Quadro 5: Algoritmo ilustrativo “Separar sílabas”	61
Quadro 6: Algoritmo ilustrativo “Descobrir sílaba tônica”	62
Quadro 7: Terminações codificadas em valores numéricos.....	70
Quadro 8: Resumo das regras do plural.....	71

LISTA DE GRÁFICOS

Gráfico 1: Relação entre treinamento versus aprendido	82
Gráfico 2: <i>Log</i> de aprendizado para quatro regras	84

1 INTRODUÇÃO

Esta tese é uma aplicação lingüístico-computacional baseada em Redes Neurais Artificiais (RNA) com um algoritmo ilustrativo sobre o plural na Morfologia do Português Brasileiro (PB): **PluralRNA**¹. O tópico é a análise do potencial lingüístico desse *software*, que é capaz de depreender regras gramaticais, notadamente a flexão de número dos substantivos e dos adjetivos. A programação diferencia-se das procedimentais em seu aspecto de resultados previamente estabelecidos, pois recebe os dados na forma de *pares lexicais singular/plural*, constituindo uma base lingüística, e os processa como fase de treinamento da rede para a depreensão da desinência de plural. Depois disso, o programa está pronto para receber novas *palavras no singular* e anexar a desinência de plural correspondente a esse léxico².

O contexto em que se insere esta tese está ligado a duas áreas específicas: Lingüística e Computação. Lingüística porque são necessários conhecimentos da Gramática do PB, notadamente a Morfologia, e a Computação por ser instrumento para as pesquisas de aplicabilidade tecnológica, codificando a linguagem natural em linguagem de máquina com o aporte do Processamento da Linguagem Natural (PLN) nas Redes Neurais Artificiais (RNA).

A importância dessas áreas, que compõem o núcleo da tese, está no fato de que é necessária a interdisciplinaridade para o avanço científico de natureza conjunta. Essa relação tripartite gera material lingüístico para análise do desempenho do *software* com o propósito de contribuir cientificamente para a Morfologia Computacional por meio de suas capacidades, limitações e perspectivas. E a relevância desse estudo está evidenciada nas análises feitas a partir dos resultados processados pelo *software*, que, por um lado, gera a resposta esperada, e, por outro, pela dificuldade da própria gramática em definir suas regras, compromete a depreensão da desinência do plural.

Duas questões surgiram quando da idéia de se fazer um estudo que

¹ Nome do *software*, que significa Rede Neural Artificial para processamento do Plural. Idealizado por mim e produzido academicamente com Alessandra Dahmer, professora da disciplina de Inteligência Artificial, e Everton J. Silva, pesquisador em informática, no Unilasalle/Canoas.

² Para evitar problemas conceituais, léxico e palavra estão no mesmo nível semântico.

envolvesse essas áreas: a) Se as regras morfológicas do PB podem ser processadas computacionalmente, ou seja, é possível tratar o plural de maneira sistemática e mecânica, fundamentado em regularidades, mesmo sabendo da existência de casos específicos de exceção a regras? b) É possível a geração de um programa computacional que permita fazer esse tipo de tratamento lingüístico?

Quanto à divisão do trabalho nos capítulos que se seguem, são estudos que buscam a corroboração dessas hipóteses.

O capítulo 2 apresenta o *Plural na Morfologia Lingüística* e o *Número em Português*. São três as seções desse capítulo que fundamentam o tópico desta tese no que se refere à questão lingüística. Na primeira (2.1), *O Estudo do Plural na Morfologia Lingüística* apresenta três subseções com a finalidade demonstrar as regularidades e irregularidades na formação do plural e o processo de pluralização no PB. Na subseção 2.1.1, são apresentadas seis regras da flexão do plural possíveis de processamento computacional e aponta a regularidade na formação de palavras como forma de possibilitar ao PluralRNA o processamento, de forma satisfatória, dos pares lexicais que geram a desinência de flexão de número, a qual é afixada nas palavras que são objetos de teste; a 2.1.2 ressalta as irregularidades como dificuldades que impedirem o *software* de fazer o processamento de forma satisfatória; e a 2.1.3 mostra o processo da flexão de número somente do PB. Na seção 2.2, *O Número em Português* diferencia as abordagens descritivo-normativas, separando metodologicamente a flexão de número nas suas subseções. A 2.2.1 apresenta as regras da Gramática Descritiva, evidenciando a dupla articulação de Martinet (1971) e descreve as regras de número em PB, ou seja, apresenta as suas realizações baseadas em vários estudos sobre linguagem natural. Na 2.2.2, está o conceito de gramática pela visão normativa, juntamente com as regras de flexão de número, as quais são a base gramatical que sustenta o núcleo lingüístico do programa. E na seção 2.3, os problemas para o processo de pluralização do PB finalizam esse capítulo, destacando discordância entre os gramáticos e lingüistas: regras que aceitam várias formas desinenciais a partir do léxico aplicado.

No terceiro capítulo, a Interface Lingüística/Computação apresenta a relação entre Lingüística, Morfologia e Computação. Na sua subseção 3.1, o destaque para a importância da Lingüística Computacional, com um breve histórico dos avanços tecnológicos a partir do Processamento da Linguagem Natural – PLN e também com várias pesquisas apontadas sobre o assunto. No 3.2, a Morfologia Computacional

apresenta um programa on-line para consultas à base de dados da Língua Portuguesa, inclusive com possibilidades de outros idiomas ligados a esse tronco lingüístico. Em 3.3, a Modelagem Computacional faz abordagens sobre as ciências cognitivas que impulsionaram a Inteligência Artificial – IA (seção 3.3.1). Na 3.3.2, o conceito de algoritmo na IA, mostrando a relação do neurônio biológico com o artificial e lançando as bases para o algoritmo de aprendizagem. A subseção 3.3.3 apresenta, em forma de fluxograma, o algoritmo ilustrativo sobre o plural no PB, que está no título desta tese e que se constitui como seu próprio objeto.

No quarto capítulo, o foco é na relação entre o comportamento do número em PB e o algoritmo criado para verificar a correspondência de seu uso com as hipóteses geradas no início desta tese. Mostra como os pares lexicais de singular e plural são armazenados no PluralRNA, como também estudos para refutar ou corroborar a hipótese principal sobre a capacidade do *software* em simular o comportamento humano em relação às regras gramaticais previamente estabelecidas. Nas seções 4.1 e 4.2, estão o *Corpus* e os Dados, que definem o material lingüístico e o programa computacional. Na seção 4.3, tem-se a aplicação metodológica do modelo computacional para tornar possível a pretensão de dotar a máquina com um comportamento inteligente. Sua arquitetura de rede é uma das fases mais relevantes da tese porque é o núcleo de funcionamento do sistema. Para ilustrá-la, há uma figura que representa, de forma artificial, o funcionamento cerebral humano. Essa rede contém três camadas, em que a primeira, a de *entrada*, permite a separação de letras e de sílabas com algumas considerações acerca das regras gramaticais de flexão de número e aponta para a necessidade de se estudar a palavra quanto ao número de sílabas e tonicidade; a segunda, a *camada intermediária escondida*, é definida pela opção de um modelo de Rede Neural Artificial – RNA; por último, a camada de saída define a maneira como serão apresentados os resultados finais. Na seção 4.4, na Implementação do Modelo, é apresentada toda a arquitetura da rede desde a sua configuração até o resultado final. A seção 4.5, Descrição do Protótipo, define o tipo de modelo de rede neural para compor o PluralRNA. Para testar a possibilidade de execução do propósito deste trabalho, é na seção 4.6, Aplicação do *Corpus/Dados*, que ocorre o Treinamento *versus* Aprendizado, em que são apresentados os erros e os acertos na rede para fazer as simulações do uso do PluralRNA, já com as telas de interação máquina-usuário. Este capítulo é finalizado com a seção 4.7, Análise dos

Resultados, em que são explicitadas as capacidades, limitações e perspectivas para futuros trabalhos.

Na conclusão, a relevância da presente tese é destacada em função da possibilidade de implementação do próprio *software* PluralRNA para futuras pesquisas nas áreas da Linguística e da Computação.

2 SOBRE O PLURAL NA MORFOLOGIA LINGÜÍSTICA E O NÚMERO EM PORTUGUÊS

Morfologia é o estudo da formação das palavras. O termo vem do grego “morfe”, significando forma. Pode-se estudar morfologia por diferentes critérios. De acordo com Pinker (1994), assim como a sintaxe, a morfologia constitui-se de um sistema extremamente bem elaborado, na qual muitas das idiossincrasias das palavras se devem a previsíveis produtos de sua lógica interna. Os dois focos principais da **morfologia** são forma e função, dividindo seus estudos em *flexão* e *formação das palavras*, em que o primeiro trata da *flexão de número e gênero*, e o segundo, da formação *derivacional e lexical*.

Esta parte da tese baseia-se tanto na forma desinencial como na sua função. A primeira caracteriza-se pela marca lingüística própria, e a segunda pela pluralização das palavras.

Necessário destacar a delimitação do material lingüístico, que tem o propósito de fazer um estudo dirigido para um tipo de desinência, a de plural, e que fornece uma gama considerável de possibilidades flexionais, as quais serão apresentadas no decorrer desta tese .

O elemento constitutivo desse estudo morfológico é o **morfema**. E é necessário conceituar o termo a partir do estruturalismo, abordagem lingüística do séc. XX proposta por Saussure em seu Curso de Lingüística Geral - CGT. Sua concepção é de sistema como objeto primário de estudo e não de elementos particulares presentes nesse sistema. Foram idéias que lançaram as bases para que outro teórico da área, Bloomfield (1984), as especificasse através dos processos segmentais e classificatórios.

Apesar de a proposta estruturalista para a morfologia, no que se refere a concatenar as seqüências sonoras ao significado das palavras, ter recebido inúmeras críticas a respeito, a contribuição de Bloomfield foi inquestionável: definiu **morfema** como *unidade básica da morfologia*, acrescentando que é uma *unidade mínima do léxico* (Elson & Pickett, 1973).

As seções a seguir subdividem este capítulo para apresentar o estudo do plural

na morfologia lingüística, a flexão de número no PB e os problemas que essa flexão provoca quando de sua aplicação.

2.1 O ESTUDO DO PLURAL NA MORFOLOGIA LINGÜÍSTICA

Com a delimitação do morfema como marca de plural nos nomes em português, especificamente no PB, é indispensável retroceder no tempo e buscar um pouco da história. Mattos e Silva (2006) destaca o “morfema sobrevivente” da rica morfologia flexional latina: o < s >, que marca tanto o plural dos nomes como também dos elementos que com eles concordam nos sintagmas nominais. Destaca também, a exemplo das regras da flexão de gênero, que hoje a formação do plural não é muito diferente do período arcaico, variando somente as alomorfias do morfema < s > decorrentes da estrutura fonológica do lexema.

É necessário também introduzir alguns conceitos sobre alomorfia, que é um tipo de variação morfêmica, e que pode se apresentar nas flexões de plural.

Houaiss (2001) afirma que

cada uma das diferentes formas fônicas (morfes) que um morfema assume em função do contexto lingüístico; p.ex., o morfema -s marca o plural em português com as formas [-s] (depois de vogal, como em cartas) e [-es] (depois de consoantes, como em mares, cartazes); o mesmo pode ocorrer com morfemas lexicais, como os alomorfes [nóbr-] em nobre, [nôbr-] em nobreza e [nobil-], em nobilíssimo.

Nesse caso, o alomorfe está sendo apresentado como variação de morfema lexical na constituição da palavra, em oposição aos morfemas que marcam o plural.

Nos conceitos de Lopes (1993), essas variações são apresentadas como noções.

A primeira noção de alomorfes coteja as palavras *barganha/breganha* e *pergunta/pregunta* para explicitar a coexistência, em português, desses pares no plano do conteúdo, embora nos planos da expressão cada um deles seja diferente. Essas quatro palavras constituem formas divergentes do mesmo morfema por serem **alomorfes** umas das outras.

Outro exemplo de alomorfia, especificamente o plural dos nomes em inglês, que é importante para reforçar a noção de morfe, está no quadro a seguir.

I	II	III
<i>Plano do Conteúdo invariante</i>	<i>Contextualização: regras de combinação sintagmática</i>	<i>Plano de Expressão (variável) correspondente às diferentes regras</i>
s = semas s ₁ — plural s ₂ — dos nomes	s ₁ — palavras terminadas por fonemas sonoros (exceto [z] e [ʒ]) REGRA: + [-z]	+ [z] { [dɔg] “dog” — [dɔgz] “dogs” — [kaw] “cow” — [kawz] “cows” — [hen] “hen” — [henz] “hens” —
	s ₂ — palavras terminadas pelos fonemas [s], [ʃ], [č], [z], [ʒ], REGRA: + [iz]	+ [iz] { [ˈhɔrs] “horse” [hɔrsiz] [praɪz] “price” [praɪziz] [rʌʃ] “rush” [rʌʃiz] [tʃɜ:ç] “church” [ˈtʃɜ:çiz] [ˈdʒɜ:dʒ] “judge” [ˈdʒɜ:dʒiz]
	s ₃ — palavras terminadas por fonemas surdos (exceto [s], [ʃ], [č]) REGRA: + [-s]	+ [s] { [kaet] “cat” — [kaets] “cats” — [kaep] “cap” — [kaeps] “caps” — [lɒk] “lock” — [lɒks] “locks” —

Quadro 1: A formação do plural dos nomes, em inglês.
Fonte: Lopes (1993)

Na primeira coluna, o *Plano de Conteúdo invariante* compõe-se de traços mínimos: semas; na segunda, são as *transcrições das regras de contextualização sintagmática* que, por serem diferentes em cada contexto, geram diversos modos de expressão de significado quando manifestadas discursivamente; por último, a coluna III transcreve o *plano da expressão* (Lopes, 1993).

Uma analogia em relação ao exemplo anterior é necessária para destacar o plano da expressão e do conteúdo, salientando a afirmação de Lopes (1993): “Por menor unidade entender-se-á uma unidade que não possa ser dividida sem que se destrua ou se altere drasticamente o seu sentido original”. Em seu exemplo em “patas” há duas formas mínimas portadores de significação:

1. “pat-“
2. “-a-”
3. “-s”

Ele explica que em 1. existe uma cadeia de significantes dotada de um plano de conteúdo próprio (“extremidades”, “relativa aos membros inferiores”, “de animais”); em 2. “-a-”, existe um plano de conteúdo “gênero feminino” que enquadra os conceitos anteriores, pertencentes a 1., no sistema gramatical do português; em 3. “-s”, há o “número plural” no plano de conteúdo. Assim, 1., 2. e 3. *possuem um significado*, que seria destruído se tentasse subdividi-los ainda mais, construindo

4. “pa-“
5. “-t-”

Esses exemplos provam que o sentido original desaparece e dá lugar ao significado de *instrumento de cavar* em 4., ficando o 5. sem qualquer significado, reforçando o que Hockett (1973) afirma: “tratam-se de morfemas, que são os menores elementos individualmente significativos nos enunciados de uma língua”.

Elson & Pickett (1973) apresentam alguns problemas especiais: os alomorfes zero. São palavras em inglês em que não há diferença de grafia, como é o caso de *sheep* (singular) e *sheep* (plural). Comparando essas duas palavras com os pares *book/books*, *cat/cats* e *foot/feet*, em que há diferenças expressas entre as formas de singular e plural, nota-se que *sheep* não carrega a marca de plural. Isso se constitui como um problema morfológico, pois a maioria dos substantivos em inglês exibe diferenças. Sugere-se então usar o símbolo –∅ quando não houver marca distintiva de plural, identificando-a como morfema zero.

As subseções a seguir subdividem este capítulo para apresentar os aspectos de regularidade e irregularidade da flexão do plural dos nomes, ou seja, das desinências de plural em relação à normatização das regras do PB, como também o seu próprio processo de pluralização. Estes dois tópicos anteciparão as possibilidades e as dificuldades que o software PluralRNA terá quando de seu processamento, tanto na entrada e fase de treinamento como na inserção da desinência de plural nas palavras que serão digitadas para serem pluralizadas.

2.1.1 A regularidade

Para que o *software* receba os pares lexicais, processe-os e gere os resultados quando da afixação da desinência nas palavras digitadas depois do treinamento, como forma de verificar se o programa depreendeu adequadamente a desinência, é necessário que as regras de formação do plural obedçam a uma regularidade. Essas regras estão bem definidas nas seções 4.1 e 4.2 (CORPUS E DADOS) e alguns exemplos na subseção 2.2.2 (Regras Normativas).

2.1.2 As irregularidades

Tudo que foge à regra gera ônus para o que se pretende desenvolver. E, nesse ponto, cabe o alerta de que o PluralRNA foi desenvolvido com o propósito de depreender a desinência de plural a partir da entrada dos pares lexicais vinculados às seis regras selecionadas em que a regularidade flexional não é comprometida.

Salienta-se aqui que o propósito do programa computacional é simular o comportamento humano utilizando um dos casos lingüísticos: o plural.

Um dos teóricos das ciências cognitivas, Steven Pinker (1999) trata desse assunto de forma conceitual. Afirma que o conceito de regularidade associa-se aos padrões obedecidos pela maior parte de palavras e que o mesmo não acontece com a irregularidade, constituindo-se esta como exceção à regra. Como a cognição humana está diretamente relacionada às atividades cerebrais, Pinker (1999) está buscando evidências no mapeamento dessa atividade com a hipótese de que essas formas de regularidade e irregularidade ocorrem diferente e separadamente no cérebro humano, inclusive já tendo recolhido boas evidências que comprovam essa hipótese.

Diante disso, é oportuno destacar a delimitação das entradas lexicais para somente os casos em que não havia dúvida em relação à correta pluralização. Ou

seja, os casos em que havia mais de uma possibilidade de flexão, por razões custo/benefício, não foram objeto deste estudo, mas, como todo trabalho científico, aponta com perspectivas de solução.

Essas irregularidades estão melhor definidas na seção 2.3 (PROBLEMAS DO NÚMERO EM PORTUGUÊS).

A subseção a seguir apresenta o processo de pluralização no PB, a fim de propiciar o cotejamento dos resultados do PluralRNA com os exemplos gramaticais da flexão do plural dos nomes.

2.1.3 O processo de pluralização no PB

A morfologia brasileira começa a sua primeira fase de desenvolvimento sob a influência do estruturalismo norte-americano e os estudos morfológicos começam a despertar interesse da comunidade científica deste país. Um dos cientistas da linguagem, o brasileiro J. Mattoso Câmara Jr. (1976), fez estudos sobre a língua portuguesa e estudos descritivos de morfologia de línguas indígenas.

Muitos problemas originaram-se daí. Um deles foi com a definição do próprio objeto da Morfologia, o morfema, pois eram múltiplas as pertinências da palavra como unidade lexical, gramatical, fonológica, semântica, etc. Com a complexidade dessas questões, estudos e propostas metodológicas para a sua resolução. Especificamente ao morfema sufixal de número, Câmara Jr. (1985) distingue o contraste das formas singular e plural a partir da presença do sufixo flexional, ou a desinência /S/, aplicado na última sílaba da palavra, o nome. É claro que as variações flexionais dessa natureza não se restringem à desinência /S/ e que dependem de conceitos particulares para inúmeros casos não regulares, o que é melhor exposto na seção 2.2 (O NÚMERO EM PORTUGUÊS) e na 2.3 (PROBLEMAS DO NÚMERO EM PORTUGUÊS).

Outro lingüista que também se notabilizou no âmbito brasileiro é Celso Pedro Luft (2002). Conceitua Morfologia como parte da gramática que se ocupa do sistema mórfico da língua, do aspecto formal das palavras. Parece nesse conceito que a abordagem é descritiva, pois a palavra *mórfico* leva a essa implicação. No entanto, a

distingue com duas terminologias: gramatical e lexical. A primeira refere-se ao sentido restrito, que se ocupa da classificação das palavras, categorias gramaticais (gênero, número, grau, pessoa, modo, tempo, aspecto), paradigmas flexionais, etc. e que segue as normas da NGB; a segunda, no amplo, a que trata de problemas como origem, formação e estrutura das palavras, famílias de palavras, etc. e a gramatical.

O quadro 2 abaixo, a exemplo do que foi exposto anteriormente com as palavras em inglês, apresenta, na primeira coluna, o Plano de Conteúdo invariante, que compõem-se de traços mínimos: semas; na segunda, são as transcrições das regras de contextualização sintagmática que, por serem diferentes em cada contexto, geram diversos modos de expressão de significado quando manifestadas discursivamente; por último, a coluna III transcreve o plano da expressão (Lopes, 1993).

<i>Plano do Conteúdo invariante</i>	<i>Contextualização: regras de combinação sintagmática</i>	<i>Plano de Expressão (variável) correspondente às difs. regras</i>
s = semas s ₁ = plural s ₂ = dos nomes	s ₁ — palavras terminadas por fonemas sonoros vocálicos ou semivocálicos (com exceção dos terminados nos ditongos [aw], [ɛw], [ɔw]) REGRA: + [-Z] (49)	+ [-Z] { <ul style="list-style-type: none"> ['meza] — ['mezaZ] [is'tāti] "estante" [is'tātiZ] [sīzejru] "cinzeiro" [sīzejruZ] ka'fɛ] "café" [ka'fɛZ] [pɔ] "pó" [pɔZ] [pi'ru] "peru" [pi'ruZ] [bɛj] "bem" [bɛjZ] zar'dī] "jardim" [zar'dīZ] [sōw] "som" [sōwZ] [sida'dāw] "cidadão" [sida'dāwZ] (50) kriztāw] "cristão" [kriztāwZ] (50)
	s ₂ — palavras terminadas por fonemas sonoros consonantais [z], grafado "s" ("português"), ou grafado "z" ("rapaz"), /ʒ/ ("mal" pronunciado [maʒ] e [r]) REGRA: + [-iZ]	+ [-iZ] { <ul style="list-style-type: none"> [ana'naz] "ananás" [ana'anaziZ] [pa'iz] "país" [pa'iziZ] [maʒ] "mal" [maliZ] [a'sukar] "açúcar" [a'sukariZ] [ko'ʒɛr] "colher" [ko'ʒɛriZ] [ra'paz] "rapaz" [ra'paziZ]
	s ₃ — palavras terminadas por fonemas sonoros (semivocálicos), [aw], [ɛw], [ɔw] REGRA: [w] ⇒ [j] + [-Z]	{ <ul style="list-style-type: none"> [ani'maw] "animal" — [ani'majZ] [pa'pɛw] "papel" — [pa'pɛjZ] [kruɛw] "cruel" — [kruɛjZ] [sɔw] "sol" — [sɔjZ] [āzɔw] "anzol" — [ā'zɔjZ]

Quadro 2: A formação do plural dos nomes, em português.
 Fonte: Lopes (1993)

Explicando de outra forma, Lopes (1993), em 1., a significação é lexical e diz respeito ao vocabulário da língua, ao dicionário; em 2. e 3., possuem significação gramatical, que diz respeito, não ao dicionário, mas à gramática de uma língua.

Acrescente-se a isso, o conceito de morfema no dicionário Houaiss (2001):

a menor unidade lingüística que possui significado, abarcando raízes e afixos, formas livres (p.ex.: mar) e formas presas (p.ex.: sapat-, -o-, -s) e vocábulos gramaticais (preposições, conjunções) Para o estruturalismo norte-americano, pode ter ainda outras manifestações, como a ordem das palavras na frase, indicando as funções sintáticas dos constituintes, ou a entonação sozinha, que pode mudar o sentido de um enunciado: *Você vai. Você vai?*

Houaiss define de forma resumida que essa unidade morfêmica tem sentido e que uma depende de outra por suas formas: *livres* e *presas*. A primeira se constitui como o **radical** e a outra, por seu aspecto dependente, como os **afixos**. No exemplo (*sapat-, -o-, -s*), em que *sapat* é o radical, *-o-* é a desinência de gênero e o *-s* a desinência de número.

2.2 O NÚMERO EM PORTUGUÊS

A *morfologia*, hoje, é estudada em níveis diferenciados, em que um deles trata do aspecto descritivo e o outro do normativo. Essas duas abordagens são apresentadas para fundamentar o *número em português brasileiro*. Margarida Basílio (2002, p. 17-18) expõe de maneira clara a distinção desses dois aspectos.

No descritivo, Basílio faz alusão ao período da Lingüística Estruturalista:

O termo “estruturalista” é bastante vago em sua referência. De um modo geral, entendemos por estruturalismo em Lingüística a idéia de que as línguas são, sobretudo, estruturas. A questão da referência é, no entanto, mais complexa. Por exemplo, é comum opor o estruturalismo à teoria gerativa transformacional, embora esta última seja obviamente uma teoria estruturalista. Aqui, estamos fazendo referência sobretudo ao período estruturalista da lingüística descritiva americana, onde encontramos um desenvolvimento maior de processos de análise morfológica. Na abordagem estruturalista, a noção de morfema é básica; o morfema é definido como a unidade significativa mínima numa língua. Em síntese, a análise morfológica consiste na apreensão de morfemas e de (Haykin, 1994).suas possíveis combinações na formação de palavras. Naturalmente, morfemas podem constituir diferentes classes, de acordo com suas propriedades de combinação.

E acrescenta que essa abordagem se preocupa com as determinações da estrutura das palavras já formadas, em que a análise morfológica estruturalista seria apenas a de estabelecer as formulações gerais que já fazem parte das formações definidas pela língua. Observa também que o problema maior desse tipo de análise está no conceito básico de morfema, que é definido em relação ao significado e que gera a gravidade da análise, porque as palavras apresentam significado global no léxico de uma língua e não necessariamente no significado das partes. Isso resulta na inviabilidade de isolar o significado das partes do significado global. Dizendo de outra forma, há elementos constituintes das palavras que não podem ser definidos em termos de significado.

No aspecto normativo, Basílio (2002, p. 15) define a Gramática Normativa como um modelo clássico que não se ocupa muito da estrutura em relação à formação de palavras, o que faz é enumerar processos e listar exemplos. Continua seu esclarecimento mencionando que

a preocupação da exaustividade é freqüente nas gramáticas tradicionais normativas; no que concerne à formação de palavras, essa preocupação se traduz na tentativa de dar conta do significado final de todas as palavras nas quais entre em jogo um dado afixo. Um outro aspecto da abordagem das gramáticas normativas no fenômeno de formação de palavras se refere ao entendimento do termo “formação”. De fato, podemos observar que “formação” tem duas interpretações: uma interpretação é ativa, em que o termo se refere ao processo de formar palavras; e uma interpretação mais passiva, em que o termo se refere à maneira como as palavras estão constituídas. As gramáticas normativas seguem, via de regra, a segunda interpretação; em conseqüência disso, procuram dar conta apenas das características das formas já construídas.

E conclui que esse posicionamento pode estar ligado à função normativa e que não caberiam novas formas na língua por se tratar de objeto de prescrição.

Um dos objetivos desta tese é estudar, predominantemente, a flexão de número do léxico do PB, notadamente dos substantivos e dos adjetivos na relação singular e plural. Esse recorte compõe o *corpus* lingüístico para abastecer o *software* criado com o propósito de processar os pares lexicais para depreender, por cálculo estatístico, as desinências de plural e aplicá-las às palavras digitadas posteriormente a esse processamento.

As duas seções a seguir fazem a distinção das duas regras gramaticais, em que uma descreve os fenômenos lingüísticos e a outra normatiza/prescreve o seu uso.

2.2.1 Regras Descritivas

As regras descritivas, ao contrário do conhecimento popular, são as que não dependem de normatização ou prescrição. E quem trata dessas regras é a Gramática Descritiva, em que teóricos descrevem os fenômenos da linguagem através da observação do comportamento humano em relação à língua falada. É essa gramática que norteia o estudo dos lingüistas. Essas regras estão ligadas a comunidades lingüísticas específicas, em que a língua sofre mudanças substanciais influenciadas pelo meio. Logicamente, ela não tem a pretensão de apontar erros, mas identificar as formas existentes de expressão como também verificar o tempo e os sujeitos responsáveis por essa alteração.

Como a linguagem humana é articulada, há uma forte motivação lingüística no sentido de definir exatamente o que significa essa articulação. Martinet (1971, p. 9-11), em relação a isso, concorda em função de que cada língua possui os seus sons, mas afirma que convém explicar pelo aspecto nocional dessa articulação. Observa que ela se manifesta em dois planos diferentes, ou seja, a dupla articulação.

A **primeira articulação** transmite as experiências e as necessidades que pretende revelar ao interlocutor, representada pela forma vocal e de seu sentido. O exemplo é a frase *Tenho uma dor de cabeça*, em que revela lingüisticamente que tem dores de cabeça, mas que poderia manifestar seu sofrimento por meio de gritos, os quais são inalisáveis em termos de comunicação lingüística. Essas unidades separadas podem exercer funções e ter significados conforme sua posição dentro de uma sentença. Por exemplo, *tenho* em *Tenho livros*; *dor* em *Dor de cotovelo*; *cabeça* em *Cabeça de prego*, etc. Se for considerado o significado do grito em várias situações, ele teria um significado para cada sentença, o que seria praticamente impossível se consideradas as várias situações contextuais possíveis de um indivíduo em uma sociedade. E se for considerada, por exemplo, a palavra *cabeça* em seu aspecto seqüencial *ca-*, *-be-* e *-ça*, não seria possível a análise de sentido dessas unidades sucessivas porque cada elemento não tem significado próprio, precisando estarem ligados entre si para compor o conjunto *cabeça*, diferenciando-se assim de outras unidades, como *cobiça*, *cabeço*, etc.

A análise dessas unidades separadamente pode ser feita com o estudo da chamada **segunda articulação**, que corresponde aos fonemas. A noção da primeira articulação independe lingüisticamente da representação fonológica em seu aspecto seqüencial. Não há dúvida alguma de que a isso tenta explicar a denominação de monema para designar a forma lingüística mínima, abrangendo o lexema, que corresponde ao semantema³, e o morfema. Martinet evidencia que, segundo esta teoria, as unidades mínimas significativas são os monemas, sendo os fonemas unidades dessa segunda articulação.

Gleason (1978) faz referência a duas unidades básicas do sistema de expressão que caracterizam a Lingüística Descritiva: o fonema e o morfema. Este último é uma unidade que entra em relação com o nível do conteúdo. O morfema compõe-se por fonemas, mas diferencia-se deles por manter relação com o conteúdo por ser este provido de significado, o que não acontece com os fonemas. Ele afirma da impossibilidade em conceituar de forma exata o morfema, mas tenta defini-lo como unidade mínima gramaticalmente pertinente. Porém, o próprio Gleason (1978) afirma que essa definição gera outro problema e seria necessário definir gramática como o estudo dos morfemas e suas combinações, entrando aí em uma circularidade. No item seguinte, tenta conceituá-lo como unidade significativa mínima, em que não pode ser dividido sem que destrua ou altere drasticamente o significado. O exemplo citado é uma palavra da língua inglesa: /streynj/, na forma *strange* é um morfema dotado de significado. Se for segmentado, obtêm-se fragmentos como /str/ ou /eynj/, que não possuem significado, ou ainda /strey/, como em *stray*, ou /strey/, como em *strain*, os quais têm significados sem qualquer relação aparente com o de *strange*. Concluindo, /streynj/ é um morfema que preenche as condições descritivas como unidade significativa mínima da estrutura da linguagem.

Outra distinção que se faz necessária é do morfema com a sílaba. O morfema pode ser constituído de somente uma sílaba, mas que contém vários fonemas. Isso acontece com outros morfemas, tanto da língua inglesa como também em outras. Para exemplificar em PB, a palavra *estranho*, traduzida do inglês *strange*, é um morfema, mas constituído de três sílabas /es/, /tra/ e /nho/.

Assim, ficou evidenciado que a separação silábica depende da concepção

³ Nome dado aos elementos formais que simbolizam na língua o ambiente bio-social, correspondendo a um conjunto de segmentos fônicos distintos de uma língua.

estrutural de cada língua.

Para Mattoso Câmara (1985), a primeira articulação da língua é a morfologia, na qual o segmento fônico está ligado à significação léxica ou gramatical. E a composição das palavras, através de seus morfemas, tem nessa derivação inúmeras questões que vão desde a apresentação de regras bem definidas até casos em que a lógica dessa composicionalidade não acontece. Especificamente no que se refere à flexão de número, esse lingüista, abordando a questão do mecanismo da flexão nominal, introduz o assunto afirmando que esse tipo de flexão é um mecanismo simples e praticamente uniforme. Acrescenta que:

A forma singular ou masculina, sem marca própria dessa sua categoria, opõe-se, respectivamente, à forma plural ou a feminina, caracterizada por um sufixo flexional específico. Para o plural – /z/, escrito sempre –s; para o feminino – /a/ átono final. Assim, as formas de singular, e que o morfema é zero (0), lobo, mestre, pastor, peru etc. se opõem as de plural lobos, mestres, pastores, perus etc. Da mesma sorte, a essas formas de singular e plural, que também são masculinas e com terminação vária, se opõem as de feminino com o sufixo flexional –a: loba (plural de lobas), mestra (pl. mestras, pastora (pl. pastoras), perua (pl. peruas) etc. (MATTOSO CÂMARA, 1985)

Essa citação apresenta a questão do gênero como classificação obrigatória de substantivos. Essa questão não será discutida neste estudo porque os pares lexicais estão restritos à flexão de plural. O gênero certamente fará parte da palavra por se tratar de um morfema anterior às desinências de número. Em alguns casos a desinência de gênero será mencionada para explicar o porquê da desinência de plural ser um no feminino e outro no masculino.

Esses autores definem os morfemas flexionais a partir de seu significado ‘fletir’, pois fletem ou alteram os morfemas lexicais adaptando-os à expressão das categorias gramaticais que a sua classe admite: nos *nomes*, gênero e número; nos *verbos*, modo e tempo, número e pessoa. Eles, os morfemas flexionais em português, são classificados em cinco tipos: *aditivos*, *subtrativos*, *alternativos*, *morfema zero* e *morfema latente*.

Os morfemas *aditivos* são aqueles que resultam do acréscimo de um ou mais fonemas ao morfema lexical de gênero e número, que aqui se faz nova restrição: somente o de número. Esses estudiosos apresentam o par *rapaz – rapazes*, em que o segmento /-es/ indica a noção gramatical de número, composto do singular e do plural. Os *subtrativos* suprimem um segmento fônico do morfema lexical. É o caso do par *órfão - órfã*, em que a noção de feminino decorre não da adição, mas da

supressão de um morfema masculino. Essa outra forma de flexão desvia-se do processo básico de formação do gênero em português. A exemplificação do gênero, contrariando aqui o objeto fundamental que é o número, deve-se à dificuldade de restringir a noção de morfema lexical somente para o número, pois a desinência de plural será diferente dependendo do gênero do léxico apresentado: *órfãos* - *órfãs*. Os *alternativos* são resultado da permuta ou alternância de um fonema no interior do vocábulo, em que a vogal tônica /-ô/ do masculino singular pode alternar com um /-ó/ no feminino e no plural, como é o caso do par *povo* - *povos*. Essa alternância se caracteriza como um traço morfológico secundário, porque ela complementa as flexões de gênero e número, enquadrando-o como uma subclasse dos morfemas alternativos. Diferenciando-se dos demais apresentados até aqui, o *morfema-zero* resulta da ausência de marca lingüística que expresse determinada categoria gramatical, ou seja, o morfema lexical isolado assume uma significação gramatical pela ausência do outro que expressa significação oposta. Exemplo disso é o morfema lexical *mar*, no qual a ausência da marca de plural /-es/ indica a noção de singular. Dentre os morfemas anteriores, os aditivos sustentam o mecanismo gramatical de gênero e número e ficam em oposição a morfemas-zero. Por último, o *morfema latente* ou *alomorfe* mantém semelhança com o morfema-zero pela ausência da marca, mas distingue-se do outro, porque não apresenta morfema gramatical que indique qualquer categoria, ou seja, não se opõe a outros morfemas para contrastar categorias gramaticais. Exemplo são os vocábulos *lápiz* e *ônibus*. Não há como definir o seu tipo de flexão numeral, seja singular ou plural.

Dito de outra forma, o morfe é uma forma (em oposição a sentido) de um morfema e representa um dado morfema, mas nem todo morfe corresponde a ele, o morfema. Um exemplo é a palavra *foi*, que é uma combinação de mais de um morfema, não pode ser segmentada em morfes. Já em *gatos* podem-se depreender dois morfes: *gato* e *-s*. Tem-se aí a relação de número que compreende a forma singular e a desinência de plural.

Para evitar problemas de entendimento em relação à expressão *desinência de número*, cabe salientar que se adota *desinência de plural*, porque é o plural que é depreendido no algoritmo de aprendizagem. Como na forma singular não há morfema que o identifique, acredita-se que essa explicação é pertinente na medida em que o uso da terminologia morfológica *número* possa dificultar o entendimento quando em oposição ao termo *plural*.

2.2.2 Regras Normativas

Diferentemente das gramáticas descritivas, as normativas estão vinculadas a prescrições gramaticais, em que o objetivo é estabelecer padrões lingüísticos imutáveis em termos lexicais e estruturais. Especificamente quanto à morfologia do plural de Língua Portuguesa, notadamente o PB, torna-se indispensável destacar que são leis gramaticais e como tais são invariáveis.

Na abordagem pela ótica da Gramática Tradicional, é o estudo das palavras quanto à sua estrutura, formação, flexão e classificação. Cada um deles trata de seus objetos específicos: a) da estrutura, *raiz, radical, tema, afixo, etc.*; b) da formação das palavras, *derivação, composição e hibridismo*; c) da flexão das palavras, *gênero, número, voz, grau, etc.*; d) da classificação das palavras, *substantivo, adjetivo, artigo, numeral, pronome, verbo, advérbio, preposição, conjunção e interjeição*.

A tradição gramatical trata das palavras independentemente de suas relações na frase. Os gramáticos Cunha/Cintra (2001) e Bechara (2006) a conceituam na Língua Portuguesa como a parte da gramática que estuda a estrutura, a formação, a flexão e a classificação das palavras. As palavras são formadas de unidades ou elementos mórfico-estruturais de natureza diversa. Como a proposta é a flexão de número dos substantivos e dos adjetivos, também chamada de plural, fez-se necessária essa delimitação.

Um dos problemas lingüísticos que suscita vários estudos é o conceito de **substantivo** e de **adjetivo**. Os gramáticos esclarecem a diferença com o apoio sintático: *o substantivo é a palavra que se usa para designar seres, coisas, idéias, podendo, mas dependendo da estrutura frasal, a palavra ser classificada também como adjetivo, conforme abaixo:*

O **azul** triste de seus olhos fascinava-me (substantivo)

O céu é **azul**. (adjetivo).

A **flexão do substantivo** quanto ao *número* pode estar no singular ou no plural. Há, porém, alguns substantivos flexionados que só se usam no singular ou no plural. São exemplos disso, as palavras *fé, caridade, parabéns, pêsames*.

Referindo-se ao plural desses substantivos, algumas regras e exemplos mantêm regularidades:

- (a) Substantivos simples, a regra geral é feita pelo acréscimo do -s à forma do singular.

Exemplos: *aluno – alunos*
 casa – casas
 urubu – urubus

- (b) Nos substantivos terminados em *al, el, ol* e *ul*, plural troca o *l* por *is*.

Exemplos: *jornal – jornais*
 pastel – pastéis
 anzol – anzóis
 azul – azuis

- (c) Nos substantivos terminados em *il* fazem o plural de duas formas:

- Quando são oxítonos, trocam o *l* por -s.

Exemplo: *funil – funis*
 projétil - projetis

- Quando são paroxítonos, trocam o *l* por -eis.

Exemplo: *fóssil - fósseis*
 projétil – projéteis

Obs.: As palavras projétil e projetil apresentam o mesmo léxico, mas isso não se constitui como irregularidade porque suas regras flexionais são bem definidas. É uma convenção lingüística que permite a dupla grafia, dependendo tanto da região como da área específica que o emprega.

- (d) Nos substantivos terminados em *r* e *z*, fazem o plural pelo acréscimo de -es

Exemplos: *pastor - pastores*
 juiz – juízes
 doutor – doutores

- (e) Nos substantivos monossílabos e os oxítonos terminados em *s* fazem o plural pelo acréscimo de -es.

Exemplos: *país – países*
 mês – meses

francês – franceses

(f) Quando não são oxítonos, os substantivos terminados em s não mudam de forma no plural.

Exemplos: o tênis – os tênis
o lápis – os lápis

A flexão do adjetivo quanto ao número segue as mesmas regras aplicadas ao substantivo. Fica claro que a diferença é somente de classificação e está vinculada à Sintaxe, disciplina que estuda as estruturas e a função do léxico que as compõem.

A lingüista Maria Helena de Moura Neves (2002) observa que a mesma forma lexical pode ter categorias substantivas (**N**) e adjetivas (**Adj**), mas que podem ser definidas pela estrutura sintática. Exemplos: (1) *assassino* (**N**) impiedoso; (2) *instinto assassino* (**Adj**).

Importante destacar que existem palavras que são usadas somente na forma singular e outras na plural, conhecidas como expressões latinas *singularia tantum* e *pluralia tantum*. São exemplos do primeiro caso: *fé, preguiça, sensatez, arroz, ouro, oxigênio*, etc.; do segundo, *anais, esponsais, exéquias, parabéns, pêsames, víveres*, etc. (NEVES, 2003)

Outro autor da gramática normativa, Almeida (1986, p. 110) descreve as regras de número, destacando problemas de entendimento em relação a singular e plural, conforme abaixo:

Os substantivos, tal qual se encontram nos dicionários, indicam um só elemento, uma única unidade, ou seja, encontram-se sempre na forma *singular*. Se tivermos necessidade de indicar mais de um ser, *flexionaremos* numericamente o substantivo e diremos então que o substantivo passou para o plural. Isto de poder o substantivo indicar um ou mais objetos é o que em gramática se chama número. Número gramatical é, pois, a propriedade que têm os substantivos de indicar um ou mais objetos.

E conclui, na citada explicação, haver em português dois números, o singular e o plural: *casa* (singular), *casas* (plural); *homem* (singular), *homens* (plural). No grego, há um terceiro, o dual, com desinência especial, assim denominado por indicar apenas duas unidades. Essa terceira foi citada por ser definida como oposição às outras duas e por seu caráter puramente classificatório, não constitui aqui objeto de estudo.

Com tais concepções da gramática normativa, parte-se para o estudo a ser

desenvolvido, pois se entende serem essas noções importantes para a fundamentação desta tese. Com isso, será possível fornecer os dados gramaticais para que o PluralRNA os processe e que gere os resultados para serem analisados e cotejados com as regularidades e as irregularidades do plural do PB, contribuindo assim com o seu potencial inferencial para contribuição à Morfologia Computacional.

A seção seguinte apresentará alguns problemas flexionais de número com o propósito de definir os tipos de pares lexicais que existem e que podem gerar problemas para o PluralRNA por se constituírem como irregularidades não processáveis segundo delimitação do *corpus* lingüístico nesta tese.

2.3 PROBLEMAS DO NÚMERO EM PORTUGUÊS

Para uma breve demonstração dos problemas flexionais de número do PB, a seguir a continuação das regras estabelecidas da seção anterior.

- (g) Nos substantivos terminados em *n* fazem o plural pelo acréscimo de -es ou -s (a segunda forma é mais usada)

Exemplos: pólen – polens (pólenes)
hífen – hifens (hífenes)
abdômen – abdomens (abdomens)
Próton – prótons (prótenes)

- (h) Os substantivos terminados em *x* formam plural sem a desinência.

Exemplo: tórax – tórax
látex - látex

Esses exemplos não são correspondidos para Houaiss (2001), pois os flexiona assim: tóraces e látices.

- (i) Os substantivos terminados em *ão* podem formar o plural de três maneiras:

- Pelo acréscimo de *s*.

Exemplo: irmão – irmãos

- Pela transformação do ão em ães

Exemplo: alemão - alemães

- Pela transformação do ão em ões

Exemplo: leão – leões

(j) Vários substantivos em ão ainda não encontraram uma forma definitiva para o plural, se bem que a flexão ões, por ser mais eufônica, impõe-se na prática. Eis alguns exemplos:

<i>aldeão</i>	aldeões	aldeãos	aldeães
<i>castelão</i>	castelões	castelãos	
<i>ermitão</i>	ermitões	ermitãos	ermitães
<i>faisão</i>	faisões		faisães
<i>hortelão</i>	hortelões	hortelãos	
<i>sultão</i>	sultões	sultãos	sultães
<i>verão</i>	verões	verãos	
<i>vilão</i>	vilões	vilãos	

Esta seção trata de alguns problemas pontuais que implicam dificuldades no desempenho do PluralRNA em função de algumas irregularidades na aplicação do plural.

No item (g), palavras terminadas em *n* aceitam ambas as formas: a inclusão da desinência –s ou –es. Mas, no primeiro caso, o resultado da flexão do plural altera a grafia da sua forma primitiva quando não mantém a acentuação (*hífen – hifens*) e a mantém se a desinência for –es (*hífen – hífenes*), constituindo-se em uma palavra grafada como proparoxítona. Por que a convenção gramatical faz essa distinção? Não se vê razão para isso e também não se acredita que o argumento seja plausível.

Em relação ao item (h), com as terminações em *x*, há uma clara discordância entre os estudiosos do PB. Apesar de se ter usado um dicionário nesta tese, o que parece que não é adequado para trabalhos científicos, mas que se entende como necessário porque o seu núcleo lingüístico é o léxico, acredita-se que a discussão não é de fácil compreensão. Por um lado, têm-se os gramáticos, e, por outro, um lingüista renomado e que trabalhou por muitos anos como lexicógrafo da Academia Brasileira de Letras. Houaiss define tóracas como plural de tórax.

Como a flexão do léxico *tórax* obedece, para Houaiss, outra regra diferente das apresentadas nas gramáticas de PB, não seria difícil de se entender que as flexões derivadas de sua forma como elemento de composição pospositivo –*tórax* também teria o mesmo tratamento. As palavras derivadas a partir de sua inserção aos outros léxicos: *alvitórax*, *cefalotórax*, etc.

Para finalizar, dentre os problemas citados nesta seção, o item (j) expõe de forma clara as três possibilidades (*ães*, *ãos* e *ões*) desinenciais para a terminação –*ão*, limitando-se a sugerir que a melhor forma é aquela em que a própria comunidade lingüística do PB estabelece como a mais eufônica. O interessante é que não existe regra geral para se definir quais as palavras que permitem somente uma, duas ou, até mesmo as três situações.

Na constituição do *corpus* desta tese, as regras que compõem este capítulo são apresentadas de forma correlata com a arquitetura de representação neuronal do PluralRNA na seção 4.6 (APLICAÇÃO DO CORPUS/DADOS).

3 NA INTERFACE LINGÜÍSTICA/COMPUTAÇÃO

O tratamento computacional das línguas naturais tem sido objeto de muitas investigações lingüísticas. Fazer com que o processo de comunicação humana seja simulado em uma máquina instiga os pesquisadores das áreas afins a modelarem seus objetos de estudo. E esse desafio tem o nome de Processamento Automático de Línguas Naturais⁴ (PLN).

Tanto a Lingüística como a Morfologia Computacionais constituem-se hoje como subdisciplinas da Lingüística e da Morfologia, respectivamente. Sistema computacional de descrição morfológica. A seguir, através da sua importância, a seção faz um breve histórico para se ter um melhor entendimento.

3.1 LINGÜÍSTICA COMPUTACIONAL

Um dos estudos que buscam a história e a metodologia desse domínio do conhecimento está no Relatório do Núcleo Interinstitucional de Lingüística Computacional NILC-ICMC-USP, com o título *Introdução ao Processamento das Línguas Naturais e Algumas Aplicações*, conforme o site do Instituto de Ciências Matemáticas e de Computação - ICMC-USP⁵.

Desde a sua introdução em nossa cultura, no início dos anos 40, os computadores digitais não só vêm contribuindo para avanços substantivos nos diversos campos do conhecimento científico, como também têm sido responsáveis pelo desenvolvimento e pela abertura de novas frentes de pesquisa que, sem eles, nunca teriam sido cogitadas. Destacam-se, por exemplo, a teoria dos autômatos, a teoria das linguagens formais, a teoria dos algoritmos, a teoria da complexidade, as teorias das lógicas não-clássicas, entre outras. Essas máquinas, que cada vez mais vão fazendo parte de nosso cotidiano e nos auxiliando na construção de conhecimentos sofisticados, colocaram seus idealizadores diante de um primeiro enigma:

⁴ Com o propósito de padronizar a expressão, doravante Processamento da Linguagem Natural.

⁵ Ver site <http://www.icmc.usp.br/~tasparado/NILCTR0710-DiasDaSilvaEtAl.pdf>.

como fazê-las “entender” instruções, necessárias para a execução de tarefas? A criação de linguagens de programação foi a resposta imediata que os cientistas encontraram para esse enigma: a comunicação homem-máquina poderia ser estabelecida por meio da “desajeitada” linguagem da máquina.

É claro que seu estágio inicial envolvia a programação procedimental, que originou vários tipos de linguagem, mas nenhuma delas tinha ainda uma perspectiva humana em relação a respostas probabilísticas. **Lisp** e **Prolog** foram as linguagens de programação que tentaram, sem êxito, chegar a esse nível, conforme excerto do Relatório citado no parágrafo anterior:

Embora a instrução codificada em Prolog seja indiscutivelmente muito mais inteligível que as seqüências enigmáticas da linguagem de máquina, ela evidentemente não é uma instrução codificada em língua natural. Se não digitarmos a instrução exatamente da forma prescrita pela linguagem Prolog, isto é, **Y is 2 + 4.**, com a variável **Y** escrita em maiúscula, a seqüência **is** com letras minúsculas e o característico ponto final, receberemos – frustrados – um **no** ou um **syntax error** como resposta. Uma outra possibilidade, cuja realização é sem dúvida muito mais complexa, continua sendo um desafio: criar programas capazes de interpretar mensagens codificadas em línguas naturais. Por que não investigar meios que façam com que as máquinas “aprendam” as línguas naturais e sejam capazes de decifrá-las? Com efeito, essa preocupação com a comunicação “mais natural” entre o homem e a máquina já se instalava, desde o momento da própria criação dos primeiros computadores. As preocupações, porém, foram muito mais além. Por que não ousar? Por que não criar meios que instrua o computador a “traduzir” frases e textos de uma língua para a outra? Questões como essas motivaram os pesquisadores a investigar o processamento automático das línguas naturais (PLN). A partir delas, inúmeros “aventureiros” se dispuseram a criar meios para decifrá-lo. Desde então, criar programas computacionais “inteligentes”, até mesmo capazes de “compreender” as línguas e, por meio delas, simular uma interação verbal com o usuário, tem se revelado um empreendimento polêmico, complexo e desafiador, porém, fascinante.

O problema dessas linguagens está na relação direta com as instruções procedimentais, em que se elaboram instruções com um objetivo bem específico: prever o resultado a partir do que se programa. Mas a linguagem humana não é previsível, por isso os estudos continuaram e ainda estão desafiando a capacidade humana de simular o seu comportamento em linguagem de máquina.

Nas décadas de 1970 e 1980, desenvolveram-se os sistemas TAUM-METEO, SYSTRAN, ATLAS II, EUROTRA e KBMT, complexos processos de tradução automática e da própria sofisticação do código lingüístico. O citado Relatório ainda contribui com as especificidades de cada um:

Tradução automática – Os sistemas de tradução automática podem ser classificados de acordo com a metodologia de tradução empregada: sistemas diretos, sistemas transferenciais e sistemas interlinguais. Os sistemas diretos buscam correspondências diretas entre as unidades lexicais da língua de partida e da língua de chegada como, por exemplo, o sistema SYSTRAN, criado para traduzir relatórios sobre a missão espacial Apollo-Soyuz. Os sistemas de transferência já são mais sofisticados como, por exemplo, o sistema TAUM-METEO, que até hoje traduz relatórios meteorológicos do inglês para o francês, e o projeto EUROTRA, que pretende traduzir as línguas dos países pertencentes ao Mercado Comum Europeu. Estes sistemas efetuam a análise sintática da frase da língua de partida e, através de regras de transferência sintática, constroem a representação sintática da frase da língua de chegada. Os sistemas interlinguais são os mais sofisticados dos três como, por exemplo, os sistemas ATLAS-II, PIVOT, ULTRA e KBMT-89, nos quais a língua de partida e a língua de chegada são intermediadas por uma interlíngua, isto é, uma representação abstrata do significado para a qual a língua de partida é “traduzida” e, a partir da qual, a língua de chegada é “gerada”.

Apesar de as tentativas atingirem melhores resultados, ainda o problema continuava: a previsibilidade de resultado em função de a construção do programa ainda ser procedimental.

Seguiram-se mais estudos com o propósito de desenvolver pesquisas acadêmicas, e foi em 1972 que o pesquisador Winograd, em sua tese de doutorado no Massachusetts Institute of Technology (MIT), criou um sistema computacional que passou a ser o marco dos estudos acadêmicos sobre o PLN: o sistema SHRDLU, em que apresentava para a comunidade científica a possibilidade da interação homem-máquina por meio de línguas naturais. Acrescenta também a justificativa para a não recorrência à Teoria Lingüística em função da pouca formalização das descrições de fatos lingüísticos, que são apontadas como agravantes para a execução de seus estudos. No *site* Revistas Eletrônicas da PUCRS⁶, Silva (2006) faz a seguinte menção sobre isso:

Quando surgiram os primeiros trabalhos de descrição de línguas naturais com o auxílio do computador, não havia teorias sintáticas suficientemente explícitas e prontas para serem computacionalmente codificadas. Os primeiros projetistas [de sistemas de PLN] que ousaram propor os sistemas de tradução automática pioneiros foram forçados a construir seus próprios modelos de descrição lingüística... Como decorrência, [esses modelos] eram muito precários e imediatistas.

Mas as experiências em PLN continuaram e uma multiplicidade de pesquisas acadêmicas passou a se somar às pesquisas comerciais, que aconteciam

⁶ Ver *site* <http://revistaseletronicas.pucrs.br/ojs/index.php/fale/article/viewFile/597/428>.

simultaneamente. Com isso, foi possível associar as experiências e manter o incentivo a novas pesquisas.

Com um programa para a área comercial, em que simulava o braço de um robô movimentando-se na tela de um computador, movimento de manipulação de um conjunto de blocos que se encontravam sobre uma mesa. Essa simulação acontecia por meio de instruções em língua inglesa digitadas no teclado de um computador. Foi com essa experiência que Winograd demonstrou para a comunidade acadêmica que, mesmo de modo primitivo, a máquina poderia ser programada para processar uma interação homem-máquina por meio de uma língua natural.

Especificamente quanto às pesquisas na área da Lingüística, o quadro abaixo apresenta uma síntese da evolução desses estudos em termos do grau de sofisticação alcançado.

<p>Década de 50: A Tradução automática</p> <ul style="list-style-type: none"> ▪ sistematização computacional das classes de palavras da gramática tradicional ▪ identificação computacional de poucos tipos de constituintes oracionais <p>Década de 60: Novas aplicações e criação de formalismos</p> <ul style="list-style-type: none"> ▪ primeiros tratamentos computacionais das gramáticas livres de contexto ▪ criação dos primeiros analisadores sintáticos ▪ primeiras formalizações do significado em termos de redes semânticas <p>Década de 70: Consolidação dos estudos do PLN</p> <ul style="list-style-type: none"> ▪ implementação de parcelas das primeiras gramáticas e analisadores sintáticos ▪ busca de formalização de fatores pragmáticos e discursivos <p>Década de 80: Sofisticação dos sistemas</p> <ul style="list-style-type: none"> ▪ desenvolvimento de teorias lingüísticas motivadas pelos estudos do PLN <p>Década de 90: Sistemas baseados em “representações do conhecimento”</p> <ul style="list-style-type: none"> ▪ desenvolvimento de projetos de sistemas de PLN complexos que buscam a integração dos vários tipos de conhecimentos lingüísticos e extralingüísticos e das estratégias de inferência envolvidos nos processos de produção, manipulação e interpretação de objetos lingüísticos
--

Quadro 3: Estudos lingüístico-computacionais
Fonte: Winograd.

Esse quadro traz, de forma resumida, os estudos mais relevantes que aconteceram desde a década de 50, com a tradução automática. De 90 em diante, o crescente volume de trabalhos decorrentes dessa evolução tecnológica gera uma dificuldade maior ainda em termos de acompanhar esse ritmo extraordinário de estudos científicos. Costa (2003) escreve, em seu artigo *A Lingüística e os estudos*

de linguagem rumo ao século XXI, as perspectivas da Lingüística para o século XXI. Para não deixar esta tese com informações que transcendem o seu foco e ao mesmo tempo não prescindir das informações de seu artigo, entende-se que a sua inserção neste trabalho é uma maneira de compartilhar conhecimentos dessa natureza. Apresenta ainda a contribuição de Fiorin (apud COSTA, 2003) em relação à interface da Lingüística com a computação:

[...] entende que, na agenda lingüística do século XXI, um ponto importantíssimo é essa interface que, para avançar, vai ter de investir na pesquisa do que é e do como se constitui a faculdade da linguagem humana, inclusive para permitir a produção de softwares para facilitar o próprio trabalho de pesquisa.

Quanto mais a produção de *softwares* aumenta, mais aumentam os *softwares* derivados dele mesmo, isto é, produzem-se programas computacionais para aumentar a sua capacidade de criar outros.

Outro nome que se destaca nessa área, e com uma das participações mais relevantes internacionalmente, é Diana Santos (1987), que trabalha com PLN, relatando em seu site. Pelo *site* dela mesmo, relata

Em 1998 tive a oportunidade, através do projecto "Processamento computacional do português" (1998-2000) de lançar as bases da Linguateca (2000-), um centro de recursos (distribuído) para o processamento computacional da língua portuguesa, cujo razão de ser é armar o português de todas os recursos e ferramentas necessárias para que o seu processamento não seja discriminado, e para que a investigação sobre a nossa língua seja tão sedutora como a sobre o inglês.

O acesso ao *site* Linguateca está disponibilizado com o propósito de compartilhar 1709 referências de trabalhos científicos específicos de PLN.

Tendo a necessidade de parar com a pesquisa na área da Lingüística Computacional, sob o ônus de transformar esta tese em um trabalho bibliográfico, não se pode deixar de citar um dos eventos que se notabilizou, e que continua, em reunir a comunidade científica mundial para avançar cada vez mais no tratamento dos PLNs: Encontro para Processamento Computacional da Língua Portuguesa Falada e Escrita – PROPOR.

Pesquisar sobre PLN é uma tarefa que exige muita leitura, paciência e testagens do material disponível na internet. Com certeza o material aqui indicado não se aproxima da grande quantidade de estudos feitos até hoje, mas conduz, de certa forma, a um caminho garantido de que a tecnologia chegou e se integrou da

Linguística de tal forma que não há como estudar linguagem prescindindo da computação.

Várias e complexas áreas do conhecimento são contempladas com o PLN, mas em função desta tese, exige-se a adoção de uma restrição: a morfologia computacional no processamento de linguagem natural, pois parte do princípio de que é necessário estabelecer um domínio de aplicação específico e criar uma gramática que possa ser depreendida por um componente processador, no caso o PluralRNA.

3.2 MORFOLOGIA COMPUTACIONAL

São inúmeros os trabalhos de interface entre a morfologia linguística e a computacional. No *site* do Instituto de Linguística Teórica e Computacional⁷ há uma fonte lexicográfica *on-line* chamada **MorDebe**, base de dados da Língua Portuguesa de Portugal, doravante Português Europeu - PE. Contém mais de 135.000 entradas e cerca de 1,5 milhões de forma flexionadas. Também oferece informações sobre a ortografia e a flexão de um grande número lexical da Língua Portuguesa. Semântica e Etimologia não são citadas no seu escopo.

Primeiramente propõe-se a descrever todas as palavras do PE, mas salienta-se que podem ser incluídas nos textos-fonte do **MorDebe** palavras pertencentes ao Português do Brasil, de Moçambique, de Angola e de Cabo Verde.

Para ilustrar o referido *software*, algumas telas foram recortadas com o propósito de exemplificar alguns léxicos da base de dados do PluralRNA para posterior análise. A tela inicial desse *software on-line*, figura 1, contém quatro campos, em que se definem os tipos de entrada e qual a palavra a ser pesquisada, mostrando no resultado as possibilidades do plural do substantivo 'aldeão', constituindo-se assim como irregularidades na flexão do plural.

⁷ O link é <http://www.portaldalinguaportuguesa.org/index.php?action=mordebecontent>

MorDebe
uma Base de Dados Morfológica de Português

Forma Pesquisa pesquisa avançada

aldeão - substantivo
masculino

Singular	aldeão
Plural	aldeães / aldeãos / aldeões

feminino : aldeã

comentários e sugestões

Figura 1: MorDebe, uma Base de Dados Morfológicos de Português – resultado1
Fonte: Ilttec.

A figura 2 apresenta o mesmo resultado para a mesma palavra, mas com a variação morfológica adjetiva, indicando também a pluralidade de opções de flexão de número, aceita pela comunidade lingüística, tanto portuguesa como brasileira.

MorDebe
uma Base de Dados Morfológica de Português

Forma Pesquisa pesquisa avançada

aldeão - adjetivo

	Masculino	Feminino
Singular	aldeão	aldeã
Plural	aldeãos / aldeães / aldeões	aldeãs

comentários e sugestões

Figura 2: MorDebe, uma Base de Dados Morfológicos de Português – Resultado 2
Fonte: Ilttec.

São muitos os recursos que esse *software* oferece e não cabe aqui explorá-lo em todas as suas potencialidades, mas com certeza o resultado do programa, se for executada a pesquisa avançada, com a digitação das palavras *aldeã* e *aldeão*, é possível constatar as três possibilidades de resposta, subsidiando assim o PluralRNA no cotejamento do resultado de seu treinamento. Mesmo sendo o

MorDebe um software do PE, tem na sua base de dados as mesmas regras normativas da formação do plural em Língua Portuguesa do PB (Linguateca, 2009).

Nesta seção, pode-se incluir o software PluralRNA como mais um instrumento computacional para tratamento morfológico, especificamente a flexão de número.

Esse programa foi criado com o propósito de armazenar pares lexicais, processá-los para depreender as desinências de plural particulares de cada regra de flexão de número do PB. A partir daí, léxicos na sua forma singular são digitados no programa para que essas inferências desinenciais, também chamadas de morfemas sufixais, sejam anexadas ao léxico digitado no programa. Mas para que isso ocorra, é necessário que a fase de treinamento da rede seja executada em suas várias iterações até que a generalização aconteça. Os resultados desse processo compõem o material desta tese com o objetivo avaliar as suas capacidades e limitações e gerar perspectivas como contribuição à Morfologia Computacional.

Para que fosse possível essa proposta, a seção 3.3, que trata da modelagem computacional, busca os fundamentos nas ciências cognitivas e na IA

3.3 MODELAGEM COMPUTACIONAL

As Ciências Cognitivas nesse processo tecnológico justifica-se pela possibilidade de simulação computacional de alguns aspectos do comportamento humano através de softwares específicos.

Nesta tese, criou-se um *software* com o objetivo de contribuir para a área da Morfologia Computacional através do estudo hibridizado entre a lingüística e a computação, através da criação de um algoritmo em linguagem natural que seja representado em linguagem de máquina: linguagem de programação, estabelecendo assim correspondência dos fatores biológicos com os artificiais.

3.3.1 Das Ciências Cognitivas à Inteligência Artificial (IA)

A organização cerebral da cognição humana é feita através de padrões de combinações nas redes neuronais de acordo com o sentido que captou o estímulo de entrada, podendo ser visual, olfativo, gustativo ou tátil (YOUNG & CONCAR, 1992).

Também é importante ressaltar que o processamento no cérebro humano ocorre em distribuição paralela, ou seja, várias sinapses de vários neurônios são ativadas ao mesmo tempo para processar diferentes informações. É o processamento em paralelo que nos permite a execução de duas ou mais tarefas ao mesmo tempo, como por exemplo dirigir um carro, prestar atenção ao trânsito, pensar no trajeto a ser feito e mexer no rádio, tudo isso ao mesmo tempo.

Na base neuronal, o cérebro é um órgão com mecanismo extremamente complexo e com muitos estudos a serem feitos para ser conhecido mais detalhadamente. O cérebro dos seres humanos, em especial, permanece rodeado de mistérios que aos poucos são revelados conforme o avanço dos estudos e pesquisas sobre o sistema nervoso.

Esse sistema nervoso tem como unidade básica o neurônio (célula nervosa) e isso já define bem a complexidade do estudo em questão. Os neurônios podem ser de vários tipos de acordo com seu tamanho, forma e função. Eles têm basicamente três elementos: o corpo, o axônio e os dendritos.

O soma é o nome do corpo celular do neurônio e contém as estruturas especializadas que garantem a sustentação do funcionamento da célula. A substância cinzenta do cérebro é formada de somas. Já a função do axônio é conduzir a informação processada e transportar a informação vinda do soma de outros neurônios. A camada interior branca do cérebro é formada pelos axônios que são revestidos por uma substância daquela cor. Os axônios podem variar amplamente em seu comprimento: de milésimos de milímetros a um metro ou mais, possibilitando a comunicação interneuronal a longas distâncias. As fibras que se estendem a partir do soma, são os dendritos – sua função é receber as informações vindas através do axônio de outros neurônios, são a porta de entrada das mensagens mandadas por outras células.

As possibilidades de conexão entre os neurônios são imensas, pois, no

mínimo, cada neurônio se conecta com outro neurônio. Estima-se que existam cem bilhões de neurônio no cérebro e que cada um se comunique com até dez mil outros neurônios, o que torna possível receber dez mil mensagens simultaneamente para formular um único *output* a ser comunicado a milhares de outros neurônios. O número aproximado de conexões neuronais é de mil trilhões.

Uma célula nervosa nunca toca diretamente em outra, há entre elas um minúsculo espaço vazio chamado sinapse. Os neurônios se comunicam através de proteínas sintetizadas na própria célula e que são liberadas pelos axônios responsáveis pela condução do output daquele neurônio. Essas proteínas são denominadas neurotransmissores produzidos por cada neurônio em número de um ou mais, conforme o tipo de estímulo a ser transmitido, segundo Caudill e Butler (1991).

A construção das conexões neuronais ocorre em sua maior parte, no período compreendido entre o nascimento e o quarto ano de vida. Essa alta atividade sináptica ocorre através dos estímulos recebidos pelo organismo enquanto interagindo com o meio, enfrentando problemas, tentando se adaptar. A atividade mental intensificada após o nascimento acelera o nível metabólico do cérebro que consegue ainda aos vinte anos de idade alterar sua estrutura e formar novas sinapses. Entretanto, são os primeiros anos de vida os mais cruciais porque nos primeiros meses surgem as sinapses básicas de percepção, e nos três anos seguintes desenvolvem-se as sinapses nas áreas secundárias que servem para interpretar estímulos com maior detalhamento. Também criam-se sinapses em áreas de associação onde as informações são organizadas.

As interações com o meio, as experiências, são essenciais para o desenvolvimento das funções cerebrais. O cérebro precisa delas para aprender. Os neurônios estimulados e usados servem como instrumento do pensamento, caso não sejam estimulados e utilizados no período crítico (cada área cerebral tem o seu) eles nunca se desenvolverão.

A plasticidade do cérebro humano, sua flexibilidade e expansão da inteligência são o resultado do seu funcionamento (processamento) simultâneo como um enorme número de informações. Esse trabalho com informações e a necessidade de armazená-las pode favorecer a criação de novas sinapses. Assim, a aprendizagem não só reforça as sinapses como também causa mudanças na arquitetura cerebral, provocando “reajustes” das redes neuronais que estão ligados ao desenvolvimento

cerebral dos primeiros anos de idade. Os reajustes de função modelam o órgão, quer dizer: a função faz o órgão.

Plásticas ou adaptáveis (flexíveis), as sinapses aumentam ou diminuem seu peso, dadas condições adequadas e causam um efeito inibitório ou excitatório sobre a atividade nervosa conectada a elas.

Com relação ao armazenamento de informações, as conexões neuronais sofrem mudanças sutis e parecem estar ligadas à aprendizagem e ao armazenamento de informações, como sugerem pesquisas na área. Há padrões de atividade elétrica nas sinapses que podem ser comparados a um código, utilizado pelo cérebro para armazenar conhecimento, pois, durante o processamento da informação os neurônios ajustam a força de suas sinapses “marcando” uma determinada rede. Possivelmente, o resgate de uma informação específica ativa aquela mesma rede “marcada” durante seu processamento.

A constituição da memória dá-se através dos padrões de atividades elétricas que resgatam as informações no momento desejado. Quanto mais utilizadas forem as informações, mais tempo elas estarão disponíveis na memória de longa duração. As informações que são utilizadas por pouco tempo, são armazenadas na memória de curta duração, sendo que esta é constantemente atualizada com informações novas, que também serão utilizadas por um tempo determinado e, posteriormente, eliminadas.

A linguagem humana permite a comunicação compartilhada da nossa língua. As propriedades que caracterizam a linguagem nos seres humanos são descritas de acordo com pontos de vistas distintos. Os lingüistas não oferecem as mesmas respostas, porém parece haver um consenso que ela seja arbitrariamente simbólica e regularmente estruturada. Quanto à primeira característica – a linguagem cria uma relação arbitrária entre um símbolo e seu referente: um objeto, um processo, uma relação ou descrição. Quanto ao fato de ser regularmente estruturada - a linguagem tem uma estrutura, (dependendo da teoria ela tem aspectos universais); cada língua tem sua estrutura própria: determinadas combinações de símbolos especialmente padronizados têm significado, combinações diferentes formam outros significados distintos.

A linguagem também pode ser explorada e descrita em múltiplos níveis: em termos de seus sons, unidades de significado, palavras, frases, discurso, contexto de produção e recepção, dentre outros.

O nível dos sons, por exemplo, é um dos mais descritos pelos lingüistas da área da fonologia e fonética. Historicamente a unidade básica de análise fonológica passou pelos níveis da sílaba, do segmento, e, por último no nível dos traços. A riqueza de detalhe das teorias fonológicas que tratam dos traços como unidade básica é espantosa, praticamente não há área da lingüística que ofereça tamanho desenvolvimento teórico.

Diante do acima exposto, esses códigos podem ser simulados via Inteligência Artificial, que é o ramo da ciência da computação que trata dos meios de representar o conhecimento através desses símbolos e que utiliza métodos heurísticos para o processamento da informação. O principal objetivo dessa disciplina é compreender a inteligência por meio da construção e programação de máquinas que procuram imitar o raciocínio humano. A subseção a seguir trata do algoritmo para a resolução do problema proposto.

3.3.2 Algoritmo em IA

Em geral, os problemas que se procuram resolver com métodos de inteligência artificial não são bem estruturados, isto é, a simples descrição do problema não determina de antemão o melhor método de solucioná-lo: não se dispõem de algoritmos — ou processos simples e sistemáticos — de resolução. A inteligência artificial substitui os métodos algorítmicos tradicionais mediante uma busca exploratória da solução.

A história moderna sobre inteligência artificial iniciou-se na década de 50. Nesses anos, surgiram vários programas de computador sobre o assunto: um demonstrava teoremas da lógica proposicional, outro servia para o jogo de damas e aprendia com seus próprios erros; um terceiro demonstrava teoremas de geometria plana, e outro mais reconhecia caracteres manuscritos. Além disso, concebeu-se uma série de linguagens de programação para a manipulação geral de símbolos.

A inteligência artificial é um ramo da ciência que elabora programas para computadores de maneira simbólica, no lugar da forma numérica usual; a "tartaruga"

da ilustração, projetada na Universidade de Edimburgo, por exemplo, responde a cada ordem do teclado com um movimento que se apresenta na tela e se converte em ação na bolha.

Pioneiros da computação previram o uso de máquinas eletrônicas para a resolução de problemas de maneira inteligente. Na década de 1940 e início da de 1950, considerava-se o xadrez e outros jogos protótipos de muitos problemas de raciocínio, já que a informação pertinente não podia ser apresentada, em geral, matematicamente e a solução dos problemas requeria um método heurístico, pois a busca de todos as possíveis seqüências de movimentos exigia muito tempo. Hoje, esses jogos são muito populares e ajudam na investigação da inteligência artificial.

Além da teoria dos jogos, grande parte da investigação inicial nesse terreno foi dedicada à resolução de pequenos problemas de raciocínio simbólico, à construção de robôs com sistemas eficientes de visão e manipulação, e a compreensão de instruções em um idioma natural humano. Realizaram-se, também, grandes esforços para a criação de máquinas que se auto-aperfeiçoassem, reconhecessem padrões e traduzissem idiomas.

Outra grande parte das pesquisas era, então, como hoje, voltada para a modelação psicológica. Os estudos sobre como o homem resolve problemas proporcionaram informação para a preparação de programas "inteligentes" de computação, que ajudaram a testar modelos psicológicos da inteligência humana. As primeiras pesquisas sobre a inteligência artificial deram lugar à "psicologia do processamento da informação", enquanto a disciplina que combina a psicologia e a investigação sobre a inteligência artificial, que também tem ligações com a filosofia, a lingüística e outras especialidades, é conhecida hoje como "ciência cognitiva".

Dentre alguns tipos de simulador, destaca-se o chamado de *sistemas de larga escala*, pois tenta tornar o programa suficientemente flexível para que a maioria dos usuários possa utilizá-lo para propósitoS específicos, sendo os mais conhecidos: o **T-Learn**, de Elman e Plunkett (1997); o **Leabra**, de Randall O'Reilly e Yuko Munakata (2000)⁸, e o **LENS** de Douglas Rohde (1999), que vem respaldado por nomes como Seidenberg, Plaut, McClelland e Patterson.

A seguir, uma breve exposição para melhor entendimento sobre Rede Neural

⁸ A fonte de estudo utilizada é a versão 2.1 do PDP++ neural network simulation software. (2001) do mesmo programa. A parte operacional básica não foi alterada, apenas foram corrigidos alguns *bugs* existentes na primeira versão. O modelo citado (2000) é o que consta no livro O'Reilly e Munakata (2000), a nova versão foi indicada pelo autor e está disponível na página dele.

Artificial no processo de correlação do neurônio artificial (figura 3) com o neurônio biológico.

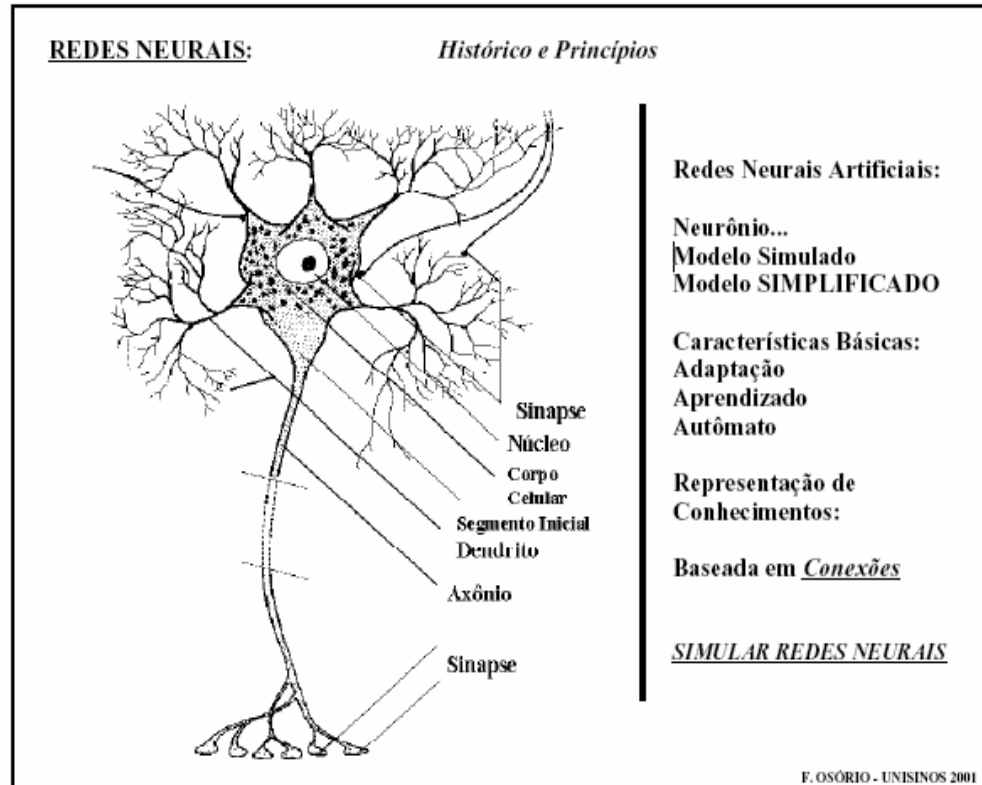


Figura 3: Modelo simulado de neurônio
Fonte: Osório⁹ – Unisinos 2001

Importante ressaltar o que Kovacz (2002) garante a respeito da RNA:

o cérebro humano é composto por aproximadamente 100 bilhões de neurônios e é considerado o mais fascinante processador baseado em carbono existente. Suas funções e movimentos estão relacionados ao funcionamento dessas pequenas células. Os neurônios estão conectados uns aos outros através de sinapses e juntos formam uma grande rede, chamada Rede Neural. Essa grande rede proporciona uma fabulosa capacidade de processamento e armazenamento de informação

Em 1943, McCulloch e Pitts projetaram a primeira estrutura conhecida como rede neural, com representação e formalização matemática dos neurônios biológicos em uma formatação artificial, o que fez surgirem os primeiros modelos de Redes Neurais Artificiais (figura 04).

⁹ Fernando Osório, pesquisador do Centro de Ciências Exatas e Tecnológicas da Unisinos/São Leopoldo

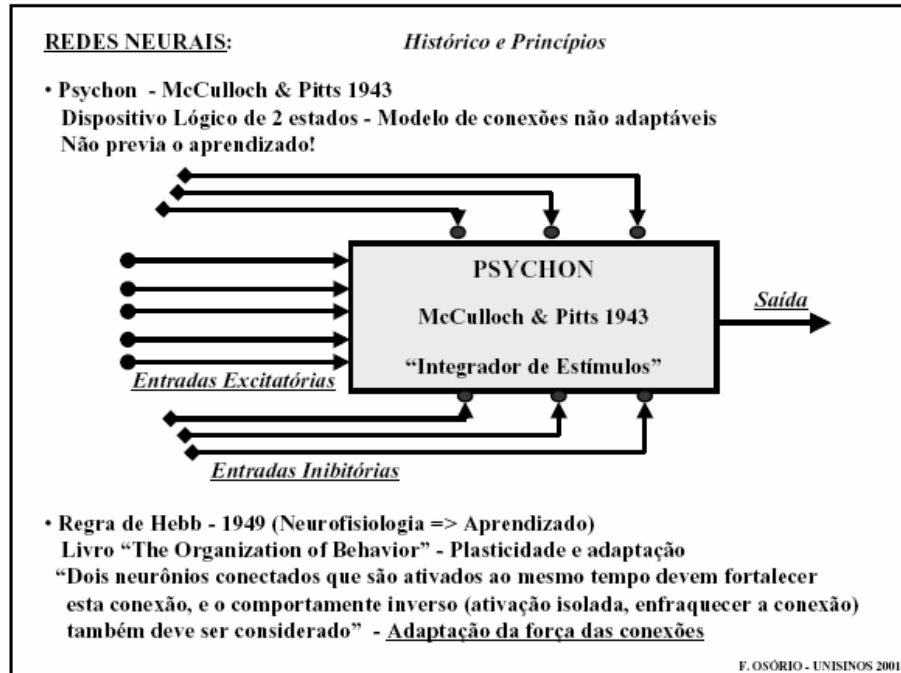


Figura 4: Modelo de McCulloch e Pitts (1943)
Fonte: Osório – Unisinos 2001

Em 1957, Rosenblat desenvolveu o **perceptron**, introduzindo assim uma nova abordagem para o problema de reconhecimento de padrões, conforme figura xx. Também propôs um algoritmo para o ajuste dos pesos do perceptron, provando sua convergência quando os padrões são linearmente separáveis.

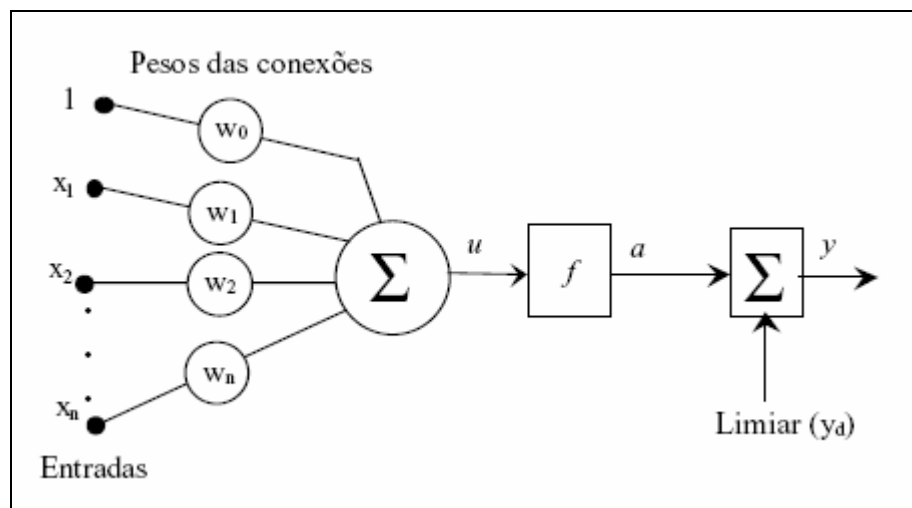


Figura 5: O Perceptron de Rosenblat (1957)
Fonte: Bilobrovec, 2004.

Apresentados o neurônio biológico e o artificial, separadamente, é importante demonstrar a sua relação de forma mais detalhada, conforme figura 06.

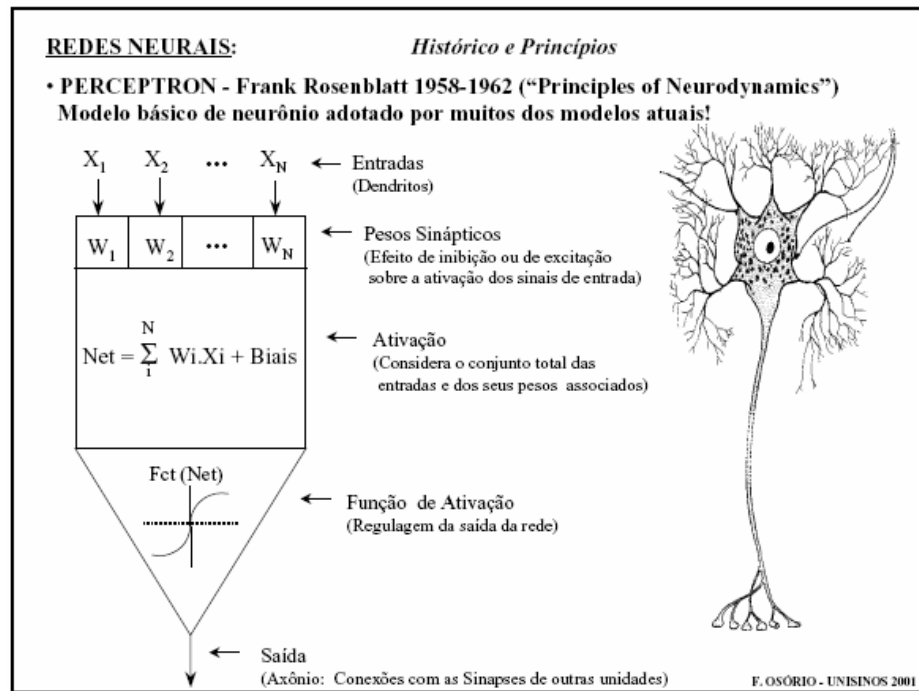


Figura 6: O Perceptron de Rosenblat (1957) e o neurônio biológico
Fonte: Bilobrovec, 2004.

Sobre aplicações de redes neurais artificiais em simulações, Bilobrovec, Marçal e Pilatti (2004), afirmam que

a idéia básica é a partir do modelo biológico de um neurônio e gerar o modelo matemático (artificial). Nesse modelo, cada sinal positivo ou negativo que entra pelo sistema é multiplicado por um número, ou peso, que indica a sua influência na saída. Caso a soma ponderada dos sinais exceda certo limite, é gerada uma resposta na saída. No modelo geral de neurônio, as entradas $W_i X_i$ são combinadas usando uma função f para produzir um estado de ativação do neurônio correspondente à frequência de descarga do neurônio biológico. As entradas chegam através dos dendritos e têm um peso atribuído pela sinapse. A função básica de um neurônio é somar as entradas e retornar uma saída. Caso esse valor seja maior que o valor de soma limite “threshold”

Entretanto, foi com Rumelhart e McClelland (1986), que as redes neurais efetivamente se desenvolveram mais. Colocaram a área de redes neurais como uma das prioritárias na obtenção de recursos em função de um método de ajuste de parâmetros de redes não-recorrentes de múltiplas camadas (MLP). Este é método baseado em um algoritmo denominado retro-propagação (backpropagation). Isso gerou uma corrida de pesquisadores das mais diferentes áreas para aplicações diversas em redes neurais artificiais.

Segundo Haykin (1994),

essas redes do tipo MLP têm sido utilizadas com sucesso para a solução de vários problemas envolvendo alto grau de não-linearidade. Seu treinamento é do tipo supervisionado e utiliza um algoritmo muito popular chamado retropropagação do erro 16 (*error backpropagation*, o mesmo que

retropropagação). Esse algoritmo é baseado numa regra de aprendizagem que “corrige” o erro durante o treinamento. Esse processo de retropropagação do erro é constituído de duas fases: uma fase de propagação do sinal funcional (*feedforward*) e uma de retropropagação do erro (*r*). Na fase positiva, os vetores de dados são aplicados às unidades de entrada, e seu efeito se propaga pela rede, camada a camada. Essas RNAs do tipo MLP com *Backpropagation* são bem utilizadas na prática. Finalmente, um conjunto de saídas é produzido como resposta da rede. Durante a fase positiva, os pesos das conexões são mantidos fixos. Na retropropagação do erro, por outro lado, os pesos são ajustados de acordo com uma regra de correção do erro. Especificamente, a resposta da rede em um instante de tempo é subtraída da saída desejada (*target*) para produzir um sinal de erro. Esse sinal é propagado da saída para a entrada, camada a camada, originando o nome “retropropagação do erro”. Os pesos são ajustados de forma que a “distância” entre a resposta da rede e a resposta desejada seja reduzida.

Como já visto, existe a possibilidade de a máquina aprender. Esse aprendizado é chamado de conexionista¹⁰, pois é um processo gradual e iterado, e que é realizado utilizando-se um conjunto de dados de aprendizado disponível. Cada iteração deste processo gradativo de adaptação dos pesos de uma rede neural é chamada de aprendizado (LUGER, 2004).

De todas as propriedades interessantes das Redes Neurais Artificiais, nenhuma “captura” tão bem a característica humana como a capacidade de aprender.

Ao invés de especificar os detalhes computacionais de processamento, tem-se a possibilidade de treinar essa rede e aguardar os resultados. Isso significa que se podem tratar problemas em que regras apropriadas são muito difíceis de se conhecer *a priori*. O objetivo do treinamento de uma RNA é fazer com que a aplicação de um conjunto de entradas produza um conjunto de saídas desejado ou no mínimo um conjunto de saídas consistente. Cada conjunto de entrada ou saída é chamado de vetor.

Segundo um método determinado previamente, esses vetores de entrada são treinados pela sua aplicação seqüencial, podendo, em alguns casos, também ocorrer com os de saída enquanto os pesos da rede são ajustados. Esses pesos da rede convergem, gradualmente, para determinados valores enquanto o treinamento estiver ocorrendo, a fim de que a aplicação desses vetores de entrada produzam as saídas esperadas. Esses procedimentos de treinamento levam as RNAs a aprender determinadas tarefas, podendo acontecer de duas formas: classe supervisionada e não Supervisionado.

¹⁰ É um termo da área das Ciências Cognitivas que trata da relação neurônio natural e artificial, conforme seção X, mas que não será objeto de estudo em função de a neurociência não ser topicalizada nesta tese.

Osório e Bittencourt (2000), com a sua afirmação

No aprendizado não-supervisionado os pesos da rede são modificados em função de critérios internos, tais como, a repetição de padrões de ativação em paralelo de vários neurônios. O comportamento resultante desse tipo de aprendizado é usualmente comparado com técnicas de análise de dados empregadas na estatística,

entende-se que, por ser uma técnica que se utiliza de estatística, o que não acontece na flexão em número das palavras, esse modelo não é o mais apropriado.

No aprendizado supervisionado o usuário ensina a rede. Sendo assim, a rede deve ser capaz de medir a diferença entre seu comportamento atual e o comportamento de referência e então corrigir os pesos de maneira a reduzir o desvio de comportamento em relação aos exemplos de referência. Esse modelo de aprendizado foi escolhido porque, como existem divergências entre gramáticos do PB em relação a algumas regras do plural, em determinados momentos a intervenção de um especialista será necessária.

E no que diz respeito aos algoritmos de treinamento utilizados, existe uma grande variedade, tanto para o treinamento supervisionado como para o não supervisionado. Entre estes, um dos mais difundidos com certeza é o algoritmo *backpropagation*, aplicado nesta tese.

Como o material que compõe o núcleo lingüístico do PluralRNA é muito complexo e com distorções no estabelecimento de determinadas regras, regularidades e irregularidades, optou-se por não utilizar nenhum dos simuladores acima e partir para a construção de um algoritmo específico, mas baseado em Rede Neural Artificial (RNA) do tipo supervisionado.

Existem vários conceitos de algoritmo, mas todos eles têm em seu núcleo significativo a rotina para alcançar um resultado. Na visão etimológica, segundo Houaiss (2001),

[...] do antr. ár. al-Khuwarizmi (matemático ár. do sIX) formou-se o ár. al-Khuwarizmi 'numeração decimal em arábicos' que passou ao lat. medieval *algorismus* com infl. do gr. *arithmós* 'número'; ver *algarismo*; f.hist. 1871 *algorithmo*.

Na abordagem matemática, em que está a origem do número, é uma seqüência finita de regras, raciocínios ou operações que possibilitam solucionar problemas a partir de um número finito de dados. Por exemplo, algoritmo para a extração de uma raiz quadrada.

Como a geração desse algoritmo situa-se no âmbito computacional e é aplicado ao tópico deste trabalho, conceitua-se como conjunto de procedimentos e

regras lógicas que levam à solução de um problema em um número finito de etapas, desde que esse conjunto esteja definido de forma precisa e sem exceção. Por extensão de sentido, a palavra método é comumente utilizada como sinônimo de algoritmo por ser um conjunto de instruções para se atingir um objetivo.

É conveniente o que Marconi e Lakatos (2001) expressam a respeito da configuração do método científico,

[...] o conjunto das atividades sistemáticas e racionais que, com maior segurança e economia, permite alcançar o objetivo (conhecimentos válidos e verdadeiros), traçando o caminho a ser seguido, detectando erros e auxiliando as decisões do pesquisador. (MARCONI; LAKATOS, 2001).

A proposta desta tese é utilizar um modelo computacional da área da Inteligência Artificial para tratar a questão da geração dos plurais e responder algumas perguntas em relação à capacidade de aprendizagem pelo computador, a partir do algoritmo de aprendizagem. Para viabilizar esse objetivo, foi elaborado o *software PluralRNA* (CONCEIÇÃO, DAHMER & SILVA, 2008), baseado no modelo de Redes Neurais Artificiais (RNA), que é capaz de reconhecer os padrões desinenciais da formação das palavras. A seguir são apresentados os passos executados para montagem e definição de um modelo de RNA, conforme Ludwig e Montgomery (2007).

Com uma arquitetura adequada, é possível evitar problemas inerentes à memorização, porque, quando a rede neural decora um conjunto de dados, não aprende e isso gera o problema de não ter resultados corretos quando a entrada é diferente do conjunto de dados de treinamento. Para garantir a sua boa funcionalidade, é necessária a execução de cinco fases distintas: *coleta e seleção de dados*); *configuração da rede*; *treinamento*; *teste e integração*, descritas a seguir:

A *configuração* de rede é dividida em quatro etapas:

- a) Seleção do paradigma neural apropriado à aplicação, isto é, a configuração da rede neural a ser utilizada apresenta boa resposta com o *Perceptron* Simples, conforme será mostrado mais adiante;
- b) Determinação da topologia da rede, ou seja, o número de camadas e o número de neurônios ou nós em cada camada;
- c) Determinação do algoritmo de treinamento, taxa de aprendizagem e demais parâmetros de treinamento;
- d) Escolha do tipo de função de transferência.

Na fase de *treinamento*, são ajustados os pesos das conexões. A tarefa aqui é a determinação dos valores iniciais dos pesos sinápticos na inicialização da rede, no algoritmo de aprendizagem e no tempo de treinamento para o aprendizado da rede.

Os valores iniciais dos pesos da rede são, normalmente, números aleatórios uniformemente distribuídos em um intervalo definido.

O algoritmo de aprendizagem é escolhido em função do tipo de rede utilizada e das características dos dados que serão ajustados. Entretanto, o processo é basicamente empírico.

Quanto ao tempo de treinamento, podem ser adotados alguns indicadores, dentre os quais um número máximo de ciclos, a taxa de erro médio por ciclo, ou ainda a capacidade de generalização da rede. Pode ocorrer que, em um determinado instante do treinamento, a generalização comece a degenerar, causando o problema de *over-training*, ou seja, a rede se especializa no conjunto de dados do treinamento e perde a capacidade de generalização: memoriza os dados e não mais consegue definir uma saída correta para outros dados que estejam fora do padrão de entrada. Nesse caso a rede decora.

O ideal é que o treinamento venha a ser interrompido quando a rede apresentar uma boa capacidade de generalização e quando a taxa de erro for admissível. Assim, deve-se encontrar um ponto ótimo de parada com erro mínimo e capacidade de generalização máxima.

Durante a fase de teste, o conjunto de validação é utilizado para determinar o desempenho da rede com dados que não foram apresentados a ela. Também é nessa fase que ocorre a verificação de aprendizagem: se a rede não decorou ou memorizou os dados de entrada, então a valida para a aplicação desejada.

São considerados ainda outros testes, como a análise dos pesos sinápticos e níveis de bias, pois, quando existem valores muito pequenos, as conexões associadas são consideradas insignificantes e assim são eliminadas (*prunning*). De modo inverso, valores muito maiores que os outros indicam a possibilidade de *over-training* da rede.

A *integração* é o processo final do projeto de uma rede neural para sua aplicação ao objetivo desejado. Assim, com a rede treinada e validada, é possível a sua integração a um sistema que contenha facilidades de utilização, como interface conveniente e facilidades de aquisição de dados através de planilhas eletrônicas, interfaces com unidades de processamento de sinais ou arquivos padronizados.

Além disso, o sistema deve periodicamente monitorar seu desempenho e fazer a manutenção da rede quando for necessário ou indicar aos projetistas a necessidade de novas seções de treinamento.

Este trabalho apresenta um modelo computacional de RNA, um *software* específico para testes e avaliações do modelo, como também os resultados esperados e os que exigem a interferência do usuário. Esse *software* obtém o plural das palavras usando RNA, já reconhecendo o padrão da flexão de número na formação desse plural, e tendo a capacidade “inteligente” de pluralizar palavras.

Com relação à aplicação metodológica do modelo computacional, é descrito juntamente com a arquitetura da RNA e os procedimentos adotados para a sua execução.

Na subseção a seguir, a relação do algoritmo em linguagem natural e o algoritmo ilustrativo.

3.3.3 O Algoritmo em linguagem natural e ilustrativo

Por que algoritmo em linguagem natural? E por que ilustrativo?

Primeiramente, é necessário destacar a diferença entre linguagem natural e linguagem de programação. Essa última é mais difícil em função de estar ligada diretamente à máquina, ou seja, são programações de baixo nível porque se aproximam da linguagem de máquina. Pode parecer estranho, mas, quanto maior for o nível da linguagem de programação, melhor será o entendimento do usuário, porque a linguagem desse tipo exige maior complexidade por parte do programador. Mesmo sendo mais fácil, o problema é que as pessoas não estão acostumadas a produzir algoritmos nem em linguagem natural.

Quanto ao algoritmo em si, não existe consenso em relação ao seu conceito. O termo é largamente associado à computação, mas sua etimologia remonta o século XVII, quando o matemático iraniano Abu Abdullah Mohammad Ibn Musa al-Khwarizmi, que nasceu em Khawarizm (Kheva), ao sul do mar Aral, cunhou esse nome. Khawarizmi também influenciou no crescimento da ciência em geral,

especificamente na matemática, astronomia e geografia. Considerado o fundador da álgebra, nome que foi motivado pelo livro *Al-Jabr wa-al-Maqabilah*, escrito por ele. (MEDINA e FERTIG, 2005)

Medina e Fertig (2005) estendem a aplicação do algoritmo às áreas da engenharia, administração, ente outras e sugerem algumas definições de algoritmo: a) um procedimento passo a passo para a solução de um problema; b) uma seqüência detalhada de ações a serem executadas para realizar alguma tarefa. Exemplos de aplicação podem ser encontrados em balancetes, receitas culinárias, itinerários, etc.

Diferentemente da programação, que exige *rigidez sintática* para que a ação seja executada sem problemas, a linguagem natural não tem esse rigor. *Rigidez semântica* também tem de ser obedecida na programação porque a sentença, sintaticamente, pode gerar ambigüidades e complicar o resultado. Exemplo disso é o que Medina e Fertig (2005) apresentam:

- *A velhinha ouviu o barulho produzido pela janela.*

Podem-se gerar três interpretações a partir dessa sentença:

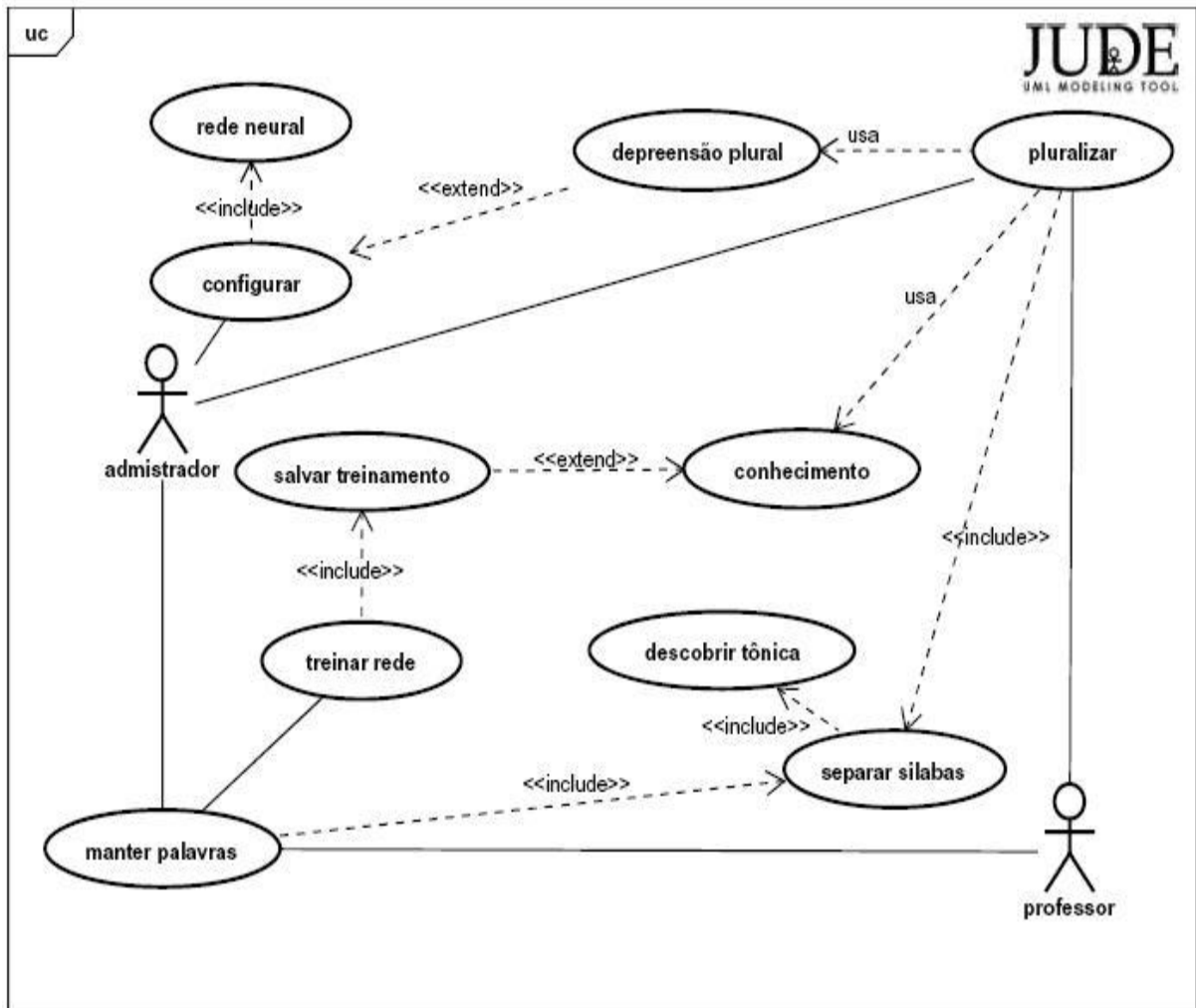
1. A velhinha ouviu o barulho produzido pela janela.
2. A velhinha estava junto à janela e ouviu o barulho.
3. A velhinha ouviu o barulho que veio através da janela.

Nenhuma máquina seria capaz de interpretar o que a sentença inicial produziu. Não adianta o programador ter a intenção ou conhecimento do que a sentença produz: a máquina não entende o que pode provocar, pois não é dotada de componentes interpretativos. Como o computador é simplesmente um executante de ordens, fica clara a necessidade da rigidez semântica quando se está programando.

Em linguagem natural seria diferente: o algoritmo para solução desse caso agregaria contextos específicos para cada situação.

Como já se tem a resposta para o porquê do algoritmo em linguagem natural, mas ainda não em relação ao termo “ilustrativo”, adjetivo contido no título da tese e que tem o objetivo de restringir o significado desse algoritmo, faz-se necessário explicitar os passos para que tanto algoritmo em linguagem natural como o ilustrativo atinjam o objetivo proposto.

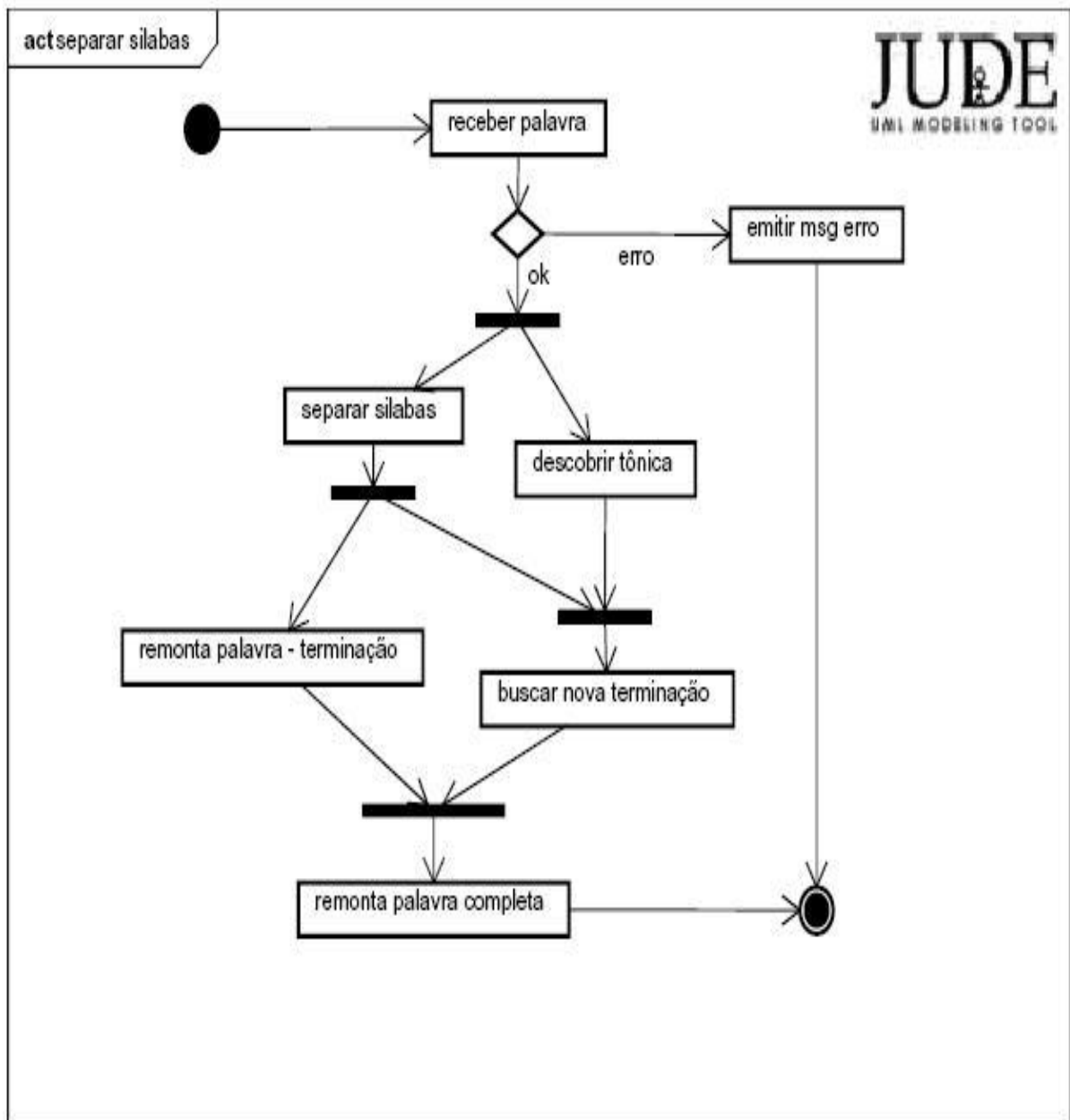
A seguir, o primeiro fluxograma (Quadro 4) que ilustra o processo pelo qual o PluralRNA foi gerado.



Quadro 4: Algoritmo ilustrativo do PluralRNA
Fonte: Celso Augusto Conceição

Esse quadro foi produzido pela ferramenta JUDE (2007), que permite criar um fluxograma a partir da linguagem de programação, que não é objeto de análise, mas que propicia ilustrar de forma gráfica o caminho computacional desde a entrada dos pares lexicais, passando pelo treinamento da rede para a depreensão da desinência, até a finalização do processo: a inserção da desinência adequada à palavra singular digitada. A fim de não gerar redundância para explicitar os passos desse algoritmo, a seção 4.4 (IMPLEMENTAÇÃO DO MODELO) os apresenta com detalhes.

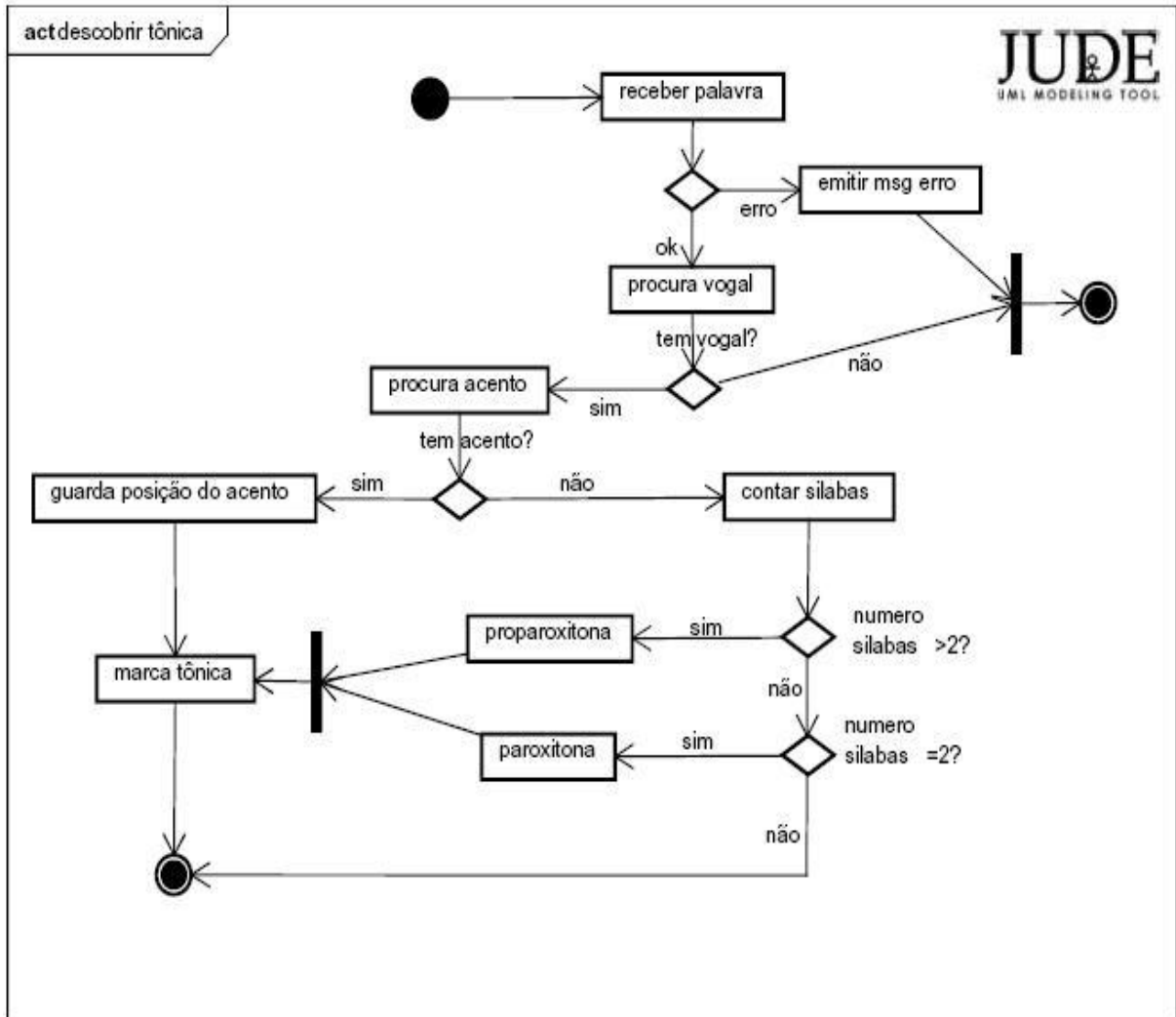
Algoritmos também podem ser ativados por outros. E é o que aconteceu no PluralRNA, que necessitou de dois algoritmos específicos: um para separar as sílabas e outro para identificar a tonicidade na palavra. O quadro 5, abaixo, ilustra os passos para essa separação.



Quadro 5: Algoritmo ilustrativo “Separar sílabas”
 Fonte: Celso Augusto Conceição

A separação de sílabas é um processo gramatical baseado na fonologia e é a tonicidade da sílaba na palavra que define a sua pronúncia.

Esse quadro ilustra o primeiro passo para definir se a palavra é oxítone, paroxítone ou proparoxítone. Isto porque as letras iguais no final de cada palavra podem provocar a geração de plural de forma inadequada. O quadro 6 esclarece melhor essa explicação e completa o processamento do PluralRNA com o algoritmo da identificação da sílaba tônica.



Quadro 6: Algoritmo ilustrativo “Descobrir sílaba tônica”

Fonte: Celso Augusto Conceição

A necessidade desse algoritmo foi imprescindível em função de que algumas palavras têm suas terminações iguais, mesmo que se diferenciem em tonicidade. É o caso, para exemplificar, de *sutil* e *fútil*. A primeira tem o seu plural em *sutis* porque é oxítona, e a segunda, em *fúteis*, porque é paroxítona terminada em I.

No capítulo a seguir são apresentados os procedimentos para aplicação da morfologia estudada e as regras gramaticais utilizadas na inserção dos pares lexicais no *software* para a geração do plural das palavras. Esse programa será melhor definido na seção 4.5 (DESCRIÇÃO DO PROTÓTIPO).

4 SOBRE O COMPORTAMENTO DO NÚMERO EM PORTUGUÊS E O ALGORITMO – RESULTADO E APLICAÇÕES

Este capítulo dedica-se a fazer a relação entre o comportamento do número em PB e o algoritmo criado para verificar se ele corresponde às hipóteses geradas no início desta tese. Convém destacar que esse número compreende o singular e o plural, e o desempenho do programa está ligado a esse último em função de ser ele o objeto-tópico do algoritmo. A entrada no PluralRNA são os **pares lexicais singular/plural** que compõem o *corpus* desta tese, mas é o resultado da desinência *plural* que deve ser considerado para efeitos de refutação ou corroboração das hipóteses: a) As regras morfológicas do PB podem ser processadas computacionalmente, ou seja, é possível tratar o plural de maneira sistemática e mecânica, fundamentado em regularidades, mesmo sabendo da existência de casos específicos de exceção a regras? b) É possível a geração de um programa computacional que permita fazer esse tipo de tratamento lingüístico?

4.1 CORPUS

Seis regras de flexão de número juntamente com o PluralRNA, que é o software para as simulações neuronais do processamento da linguagem natural, constituem o objeto desta tese. A coleta dos dados para a composição do *corpus* lingüístico está direcionada para as simulações no software PluralRNA. E como tal, a seleção restringe-se aos pares lexicais que seguem os padrões dessas seis regras, a fim de municiar esse instrumento computacional para a análise de seus resultados, ou seja, esses pares correspondem ao número de neurônios da camada de saída da rede.

Com o propósito de tornar mais compreensível o processo de simulação, resultado e análise, as regras apreendidas pelo processamento e anexadas aos respectivos neurônios serão representadas por R_n , em que n será cada uma dessas

regras. As R1 a R6 serão dispostas abaixo de cada regra de flexão do plural. Os exemplos a seguir são de Bechara (2006):

a) Formação do plural com acréscimo de **-s**

R1 (*S +1*), em que *-s* é a desinência de plural acrescentada à palavra digitada.

b) Formação do plural com acréscimo de **-es**

R2: *ES +2*, em que *-es* é a desinência de plural acrescentada à palavra digitada.

c) Plural dos nomes em **x**

R3: *igual*, em que não há modificação flexional.

d) Plural dos nomes terminados em **-l**

R4: *lS -1*, em que é reduzida a última letra da palavra digitada e adiciona-se *-is*.

e) Plural dos nomes terminados em **-l**

R5: *S -1*, em que é reduzida a última letra da palavra digitada e adiciona-se *-s*.

f) Plural dos nomes terminados em **-l**

R6: *lIS -2*, em que são reduzidas as últimas duas letras e adiciona-se *-eis* à palavra digitada.

A partir dessas regras, a próxima seção especifica os pares lexicais que contemplam as regras definidas para o processamento computacional.

4.2 DADOS

Diferentemente do *corpus*, que se caracteriza pelo conjunto PluralRNA e a definição das regras de flexão do plural, esta seção define especificamente os pares lexicais que compõem esse *corpus*.

Os dados são numerados para facilitar a sua referência:

Dados1

Os pares lexicais que compõem a digitação no PluralRNA:

Porta – portas

Jornal – jornais

Fusca – fusca

Tórax – tórax

Pardal – pardais

Doutor – doutores

Funil – funis

Juiz – juízes

Dados 2

Os pares lexicais que compõem a base de dados do PluralRNA para o treinamento da rede:

Porta – portas

Jornal – jornais

Fusca – fusca

Tórax – tórax

Pardal – pardais

Doutor – doutores

Funil – funis

Juiz – juízes

Plural – plurais

Bombril - bombris

Dados 3

As R1 a R6 serão dispostas abaixo de cada regra de flexão do plural. Os exemplos a seguir são de Bechara (2006):

a) Formação do plural com acréscimo de -s

R1 (*S +1*), em que -s é a desinência de plural acrescentada à palavra digitada.

a1) terminadas por vogal ou ditongo oral: lei-leis, cajá-cajás, casa-casas

a2) vogal nasal tônica ou átona: imã-imãs

a3) ditongos nasais -ãe (tônicos ou átonos) e -ão (átono): mãe-mães; bênção-bênçãos, mão-mãos

a4) terminadas em "n": hífen-hifens

b) Formação do plural com acréscimo de -es

R2: *ES +2*, em que -es é a desinência de plural acrescentada à palavra digitada.

b1) Palavras terminadas em -z (em sílaba tônica): luz-luzes

b2) Palavras terminadas em -r: cor-cores; elixir-elixires

c) Plural dos nomes em x

R3: *igual*, em que não há modificação flexional.

c1) palavras terminadas em -s: lápis

c2) palavras terminadas em -r: tórax

d) Plural dos nomes terminados em -l

R4: *IS -1*, em que é reduzida a última letra da palavra digitada e adiciona-se -is.

d1) nomes terminados em -al: carnaval - carnavais

d2) nomes terminados em –el: papel - papéis

d3) nomes terminados em –ol: roxinol - roxinóis

d4) nomes terminados em –ul: azul – azuis

e) Plural dos nomes terminados em –l

R5: S -1, em que é reduzida a última letra da palavra digitada e adiciona-se -s.

- palavras terminadas em –il (oxítona): funil - funis

f) Plural dos nomes terminados em –l

R6: EIS -2, em que são reduzidas as últimas duas letras e adiciona-se -eis à palavra digitada.

- nomes terminados em –il (paroxítona): fóssil-fósseis;

g) O par lexical *limão-limões* não tem correspondência nas seis regras por não terem sido consideradas todas as possibilidades de pluralização, mas é oferecido para análise do comportamento da rede.

Os dados aqui selecionados são uma amostragem das seis regras contempladas para compor o PluralRNA.

Delimitadas as regras e selecionados os dados, agora é possível partir para a aplicação metodológica do modelo computacional, o que é explicitado a seguir.

4.3 APLICAÇÃO METODOLÓGICA DO MODELO COMPUTACIONAL

Esta seção apresenta a metodologia desenvolvida para a relação do *software* e as seis regras de flexão do plural, como uma pesquisa de natureza aplicada, tendo uma abordagem quantitativa, com objetivos exploratórios e procedimentos comparativos.

Esta seção apresenta a metodologia desenvolvida para a relação do *software* e as seis regras de flexão do plural, como uma pesquisa de natureza aplicada, tendo uma abordagem quantitativa, com objetivos exploratórios e procedimentos comparativos.

De acordo com o que foi exposto na introdução e apresentado como objetivo deste trabalho, foi pesquisado e definido um modelo computacional de RNA, que torna possível ao computador o comportamento inteligente de gerar o plural das palavras.

A intervenção humana no programa é prevista para se ter uma referência do certo e do errado quando da utilização do algoritmo de aprendizagem.

Nesse programa computacional, cada palavra é separada e classificada quanto ao número de sílabas (monossílabo, dissílabo, trissílabo e polissílabo), quanto à tonicidade e quanto à acentuação gráfica (oxítona, paroxítona e proparoxítona).

Para a classificação da palavra, foi necessário fazer a separação das sílabas, em que foi utilizado o algoritmo de Divisão Silábica de Texto Escrito e Falado (GOUVEIA, 2000) e como solução para o caso da fonética foi aplicado o algoritmo utilizado no projeto do Sintetizador de Voz de Qualidade LianeTTS (BORGES, 2007) em *software* livre desenvolvido na Universidade Federal do Rio de Janeiro (UFRJ).

Essas classificações são importantes porque são informações que definem a maneira de pluralizar uma palavra. No processo de treinamento da RNA, foram mostradas as regras dos plurais e o porquê dessas classificações.

Como solução do problema de classificação de padrões para o algoritmo proposto, foi escolhida uma rede neural do tipo multicamadas (MLP), devido à sua fácil implementação e também pelo fato de bons resultados alcançados segundo literatura que trata dessa abordagem (CONRADS & ROEHL, 1999 e YONG & LIM, 2001).

Então, parte-se para a **arquitetura de rede**, que tem sua estrutura composta por uma camada sensorial, também chamada de **camada de entrada**, que recebe três sinais de entrada (Penúltima Letra - PL, Última Letra – UL, Classe de Palavra - CL), uma **camada oculta**, contendo quatro neurônios para a média ponderada entre entradas e saídas (KOVACS, 2002), e uma **camada de saída**, constituída de seis neurônios, cada um representando uma regra de flexão (R1 a R6), conforme o grafo arquitetural apresentado na figura 7. Sua alimentação é do tipo *feedforward*, função

de transferência sigmóide, e o algoritmo de treinamento escolhido é o *backpropagation*.

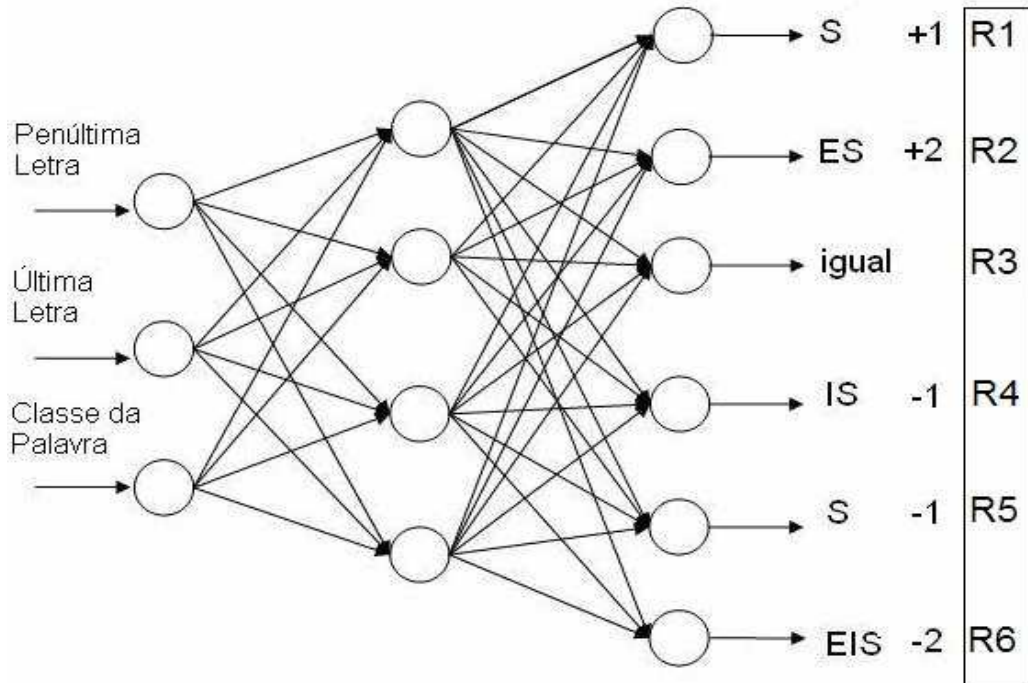


Figura 7: Grafo arquitetural da MLP
Fonte: Conceição, Dahmer & Silva.

Como são seis as camadas de saída, que correspondem a seis regras flexionais do plural inferidas pelo PluralRNA, cada uma dessas regras terá a representação de R1 a R6, conforme figura 8.

- R1:** S +1, em que -s é a desinência de plural acrescida à palavra digitada.
- R2:** ES +2, em que -es é a desinência de plural acrescida à palavra digitada.
- R3:** igual, em que não modificação flexional.
- R4:** IS -1, em que é reduzida a última letra da palavra digitada e adiciona-se -is.
- R5:** S -1, em que é reduzida a última letra da palavra digitada e adiciona-se -s.
- R6:** EIS -2, em que são reduzidas as últimas duas letras e adiciona-se -eis à palavra digitada.

Figura 8: Representação dos neurônios da camada de saída
Fonte: Conceição, Dahmer & Silva.

Detalhando melhor, a camada de entrada recebe três sinais que são as duas últimas letras da última sílaba da palavra e a sua classificação é obtida, respectivamente, através do algoritmo de Divisão Silábica de Texto Escrito e Falado (GOUVEIA, 2000) e do algoritmo para o Sintetizador de Voz de Qualidade LianeTTS (BORGES, 2007).

Como visto no referencial teórico, as terminações das palavras determinam se a palavra está no singular ou no plural, excetuando-se casos em que ela fica invariável. Dessa maneira, o primeiro passo é codificar as informações em valores numéricos, considerando algumas das regras de flexão.

- a) Em substantivos simples, acrescenta-se a desinência 's' ao substantivo terminado em vogal, ditongo oral ou ditongo nasal 'ãe';
- b) Os substantivos terminados em 'al', 'el', 'ol' e 'ul' fazem o plural trocando o 'l' por 'is'.

Sendo assim, para efeitos ilustrativos, podem-se representar os sinais de entrada conforme quadro 7.

Terminação	=	Valor Numérico de Entrada
a	=	10
e	=	10
o	=	10
al	=	20
el	=	20
ol	=	20

Quadro 7: Terminações codificadas em valores numéricos
Fonte: Conceição, Dahmer & Silva.

Esse quadro contém os valores numéricos para dois neurônios de entrada: quando se levar em conta somente a última letra, codifica-se como 10; se for a penúltima, 20. O terceiro neurônio não se constitui como variável porque a classe da palavra está definida como parâmetro único, tanto para substantivo como para adjetivo.

Quanto à camada intermediária ou escondida, trata-se do local em que é realizado o processamento através de conexões ponderadas. O número de elementos numa camada intermediária define a treinabilidade e a precisão da aproximação funcional. Para o trabalho, foram utilizados 4 neurônios nessa camada, que é a média ponderada entre a camada de entrada e a camada de saída. A quantidade de neurônios é proporcional ao aumento da capacidade de aprendizado da rede: quanto mais, mais aprende (KOVACS, 2002).

No que se refere à camada de saída, apresenta-se o resultado final obtido após os cálculos efetuados nas camadas anteriores, ou seja, um neurônio é ativado com a desinência depreendida, conforme quadro 8 abaixo, que apresenta uma amostragem de algumas das regras do plural e as saídas possíveis..

Terminação	Saída
a, o, u; n; ão	Adicionar o "S"
s – oxítonos	Adicionar o "ES"
s – não oxítonos; x	Palavra fica igual
al, el, ol, ul – oxítonas	Reduzir uma letra e adicionar o "IS"
il – oxítonas	Reduzir uma letra e adicionar o "S"
il – paroxítonas	Reduzir duas letras e adicionar o "EIS"

Quadro 8: Resumo das regras do plural
Fonte: Conceição, Dahmer & Silva.

A seguir, a implementação do modelo computacional lança as bases para a criação do protótipo do software PluralRNA.

4.4 IMPLEMENTAÇÃO DO MODELO

Nesta seção, são abordadas questões referentes à implementação e ao projeto físico do protótipo do *software* que foi denominado PluralRNA, que já é utilizado para avaliação do presente modelo. Para atingir o objetivo do algoritmo computacional,

foram efetuados muitos estudos de forma heurística até chegar no desenvolvimento desse protótipo de *software*, conforme Apêndice C. A partir daí foi possível avaliar o comportamento e a utilização de Redes Neurais Artificiais na Morfologia do PB. O usuário digita na entrada uma palavra no singular e o computador, “automaticamente”, gera a palavra no plural, o que é cotejado com o conhecimento do léxico e suas regras de flexão do plural do PB para garantir ou não a sua exatidão.

Em relação à Modelagem das funcionalidades, para documentar o modelo proposta para o trabalho, foram utilizados diagramas especificados pela notação UML (OMG, 2006), confeccionados com o auxílio da ferramenta JUDE na versão *Community 5.2.1* (JUDE, 2007), e demonstrados na subseção 3.3.3 (O Algoritmo em linguagem natural e ilustrativo).

A dinâmica do sistema foi modelada através de diagramas de caso de uso, relacionando as funcionalidades e os elementos externos que interagem com o sistema, através do elemento ator. Os atores são as pessoas ou entidades computacionais envolvidas na execução das funcionalidades do sistema.

A partir de uma análise funcional do principal requisito desse protótipo, verificou-se a necessidade de dois níveis de usuário do software: administradores e professores.

O administrador representa o usuário com a responsabilidade de configurar parâmetros como número de épocas, neurônios da camada de entrada, neurônios da camada oculta, neurônios da camada de saída, taxa de aprendizagem e taxa de inércia. Todos estes parâmetros estão diretamente ligados à arquitetura que foi indicada por este trabalho e o porquê de existir este nível de usuário no sistema.

E o professor é a representação do usuário com a tarefa principal de utilizar a aplicação e avaliar o funcionamento e as respostas obtidas da RNA que foi implementada. No caso de uma resposta inválida, este usuário tem a opção de efetuar um treinamento e mostrar para a RNA qual a resposta certa.

Os casos de uso demonstram como ocorre a interação entre os atores e o sistema, e eles serão descritos a seguir.

No gerenciamento da RNA, as operações são realizadas pelo administrador, conforme os tipos de configuração a seguir:

a) **Configuração da arquitetura da RNA:** O administrador pode configurar

parâmetros como número de épocas, neurônios da camada de entrada, neurônios da camada oculta, neurônios da camada de saída, taxa de aprendizagem e taxa de inércia.

- b) **Configuração dos pesos da RNA:** Por padrão, esses pesos foram deixados fixos no programa para evitar maiores dificuldades, em que foi definida a conversão da letra para seu valor na tabela ASCII. Os valores adotados para as taxas de aprendizagem e inércia irão interferir diretamente no processo de atualização dos pesos.

Uma taxa alta de aprendizagem pode fazer o processo oscilar em torno da solução procurada. Por isso se usa um determinado valor para a taxa de inércia com o objetivo de evitar tais oscilações e atingir a solução em menor tempo.

O treinamento da RNA está disponível aos dois atores, compreendendo o processo de treinamento/aprendizado. Há um subconjunto definido de valores de entrada e de saída correlacionados, que é apresentado à rede MLP em várias épocas ou iterações, ou seja, é apresentado o par lexical para que a rede neural identifique e depreenda as desinências do plural.

Nesta apresentação, o valor obtido na saída é comparado com o valor desejado ou "alvo". Através de *backpropagation*, as conexões sinápticas são atualizadas uma a uma, de forma a minimizar o erro e encontrar um mínimo global para a rede MLP. Desta forma, a rede é capaz de generalizar soluções para o problema em foco.

Os casos de uso compreendem o processo de teste da RNA. Após a fase de treinamento, a rede MLP está apta (caso tenha convergido adequadamente) para executar a tarefa esperada. Apresenta-se então um subconjunto de valores de entrada diferente do subconjunto de treinamento, e, a partir dos estados atuais das sinapses, a rede MLP processa esses valores e produz uma saída compatível e conforme o "conhecimento adquirido".

Como já foi definida a implementação do modelo, já é possível descrever o protótipo do PluralRNA, o que está apresentado na seção seguinte.

4.5 DESCRIÇÃO DO PROTÓTIPO

Depois das análises e dos resultados obtidos na implementação do modelo adotado, optou-se pela utilização de seus componentes de rede MLP, com aprendizado de *retropropagação* de erro.

A escolha desse modelo de RNA deu-se principalmente pela grande capacidade de generalização e na sua operacionalização com baixa complexidade.

Dentre as ferramentas de programação utilizadas, a que melhor satisfizes a necessidade para o desenvolvimento do protótipo foi o Delphi¹¹, porque sua característica RAD (*Rapid Application Development* ou Desenvolvimento Rápido de Aplicações), torna-se um diferencial em relação a outras linguagens de programação para esse tipo de trabalho, em que foi avaliado o comportamento da RNA e não questões de desenvolvimento de *software*.

A primeira etapa da implementação foi a estruturação da base de dados. Para esse protótipo optou-se pela criação de apenas um banco de dados, em que todas as estruturas necessárias para a aplicação foram criadas e atualizadas. A base de conhecimento é apresentada no Apêndice A. Essa estrutura tem o objetivo de armazenar informações das áreas de estudos suportadas pelo PluralRNA e contempla o cadastramento inferencial de Regras do Plural e a Arquitetura da RNA.

Poucas tabelas foram utilizadas nesse protótipo. A tabela CONFIGURAÇÃO é utilizada para armazenar a configuração atual da RNA e possui os atributos descritos anteriormente: neurônios de entrada (configurado para três), neurônios ocultos (configurado para quatro), neurônios de saída (configurado para seis), número de iterações, taxa de aprendizagem e taxa de inércia. Esses últimos três parâmetros são melhor detalhados na implementação das telas.

A tabela LETRA_PESO é utilizada para representar o valor de cada letra do alfabeto e o seu valor da tabela ASCII, respectivamente. Por exemplo, letra A tem valor na tabela ASCII igual a 65, letra B, valor 66, e assim por diante.

A tabela PALAVRA_CLASSIFICACAO contém as três classificações quanto à sílaba tônica, são elas: oxítone, paroxítone e proparoxítone. Os monossílabos, para efeito de entrada no software, são entendidos como oxítonas. Com isso evitou-se

¹¹ Ferramenta de programação

ampliar a complexidade do programa, o que não inviabiliza fazê-lo em futuros estudos.

A tabela SINGULAR_PLURAL é utilizada para o cadastramento dos treinamentos a serem executados na RNA. Nessa tabela o usuário cadastra a palavra no singular e a palavra no plural. A partir desse cadastro o sistema identifica a desinência de plural que está sendo usada nesta combinação e a envia para a RNA. Essa desinência é armazenada na tabela NEURONIO_SAIDA, em que as letras são adicionadas e o quanto é necessário alterar a palavra de resposta, além de informar qual o neurônio que deve ser “ativado” na saída da RNA. O processo inverso ocorre quando do teste da RNA. A partir do neurônio de saída, as letras devem ser adicionadas na nova palavra.

As tabelas CONHECIMENTO e UTILIZACAO são utilizadas para gerar os logs necessários para avaliação desse modelo.

E, por fim, as tabelas PALAVRA e PALAVRA_CLASSE representam o dicionário léxico com base no LMCPD - Léxico Multifuncional Computadorizado do Português Contemporâneo (NASCIMENTO, 2000) que foi inserido nesse protótipo.

Na segunda etapa da implementação proposta, construíram-se as interfaces do aplicativo: algoritmos de separação de sílaba tônica, apresentados no Apêndice B.

Para construir um modelo computacional que validasse a eficiência da rede neural, a topologia MLP utilizada foi a mais adequada. O programa desenvolvido para o fim específico de pluralizar a palavra possui agora uma interface que permite a manipulação dos dados referentes à configuração dos parâmetros dessa rede MLP, uma interface que permite o treinamento e o teste da rede com a entrada de palavra no singular.



Figura 9: Tela de configuração da rede neural
Fonte: Conceição, Dahmer & Silva.

A configuração da rede neural, conforme figura 9, mostra a interface destinada à informação dos dados de configuração da arquitetura da rede MLP, na qual são apresentados os campos para as alterações de variáveis que definem o tipo de treinamento da rede MLP.

O campo **Nº de Épocas** refere-se ao número de iterações de treinamento da rede MLP; o campo **Neurônios Camada Entrada** recebe as três informações importantes para o processamento: a penúltima letra da palavra, a última letra e sua classificação quanto à sílaba tônica; o campo **Neurônios Camada Oculta** refere-se à quantidade de neurônios que irão constituir a camada oculta da rede MLP, ou seja, a camada ligando os neurônios de entrada aos de saída; o campo **Neurônios Camada Saída** possibilita as seis saídas que formam o plural das palavras, de acordo com cada desinência depreendida: S, ES, igual, IS (-1), S (-1) e EIS(-2), ressaltando que as desinências que não estão acompanhadas por parênteses somente acrescentam à palavra a desinência do plural. Ao contrário das que têm, o (-1) refere-se à eliminação de uma letra antes da inserção desinencial, e a (-2), a eliminação de duas. O campo **Taxa de Aprendizagem** refere-se ao índice que define a atualização das sinapses (variando de 0% a 100%), ou seja, o quanto deverá ser alterado a partir do valor atual da sinapse para a próxima etapa; e o **Taxa de Inércia** refere-se ao índice que define a inércia ou *Momentum* da aprendizagem (variando de 0% a 100%), ou seja, quanto do valor atual da sinapse irá persistir na próxima época.

Haykin (1994) afirma que

taxa de aprendizagem tem um papel importante no treinamento por retropropagação do erro. Se a taxa é muito baixa, o algoritmo demorará para convergir. Se, por outro lado, a taxa é muito alta, o algoritmo pode se tornar instável. Um método simples de aumentar a velocidade de convergência e evitar a instabilidade é modificar a lei de ajuste adicionando um termo de momentum que é proporcional ao ajuste de pesos anterior.

Essa afirmação corrobora a decisão de adequar os valores para que a rede seja otimizada. Esses valores foram adotados para as taxas de aprendizagem e inércia, as quais interferem diretamente no processo de atualização dos pesos. Uma taxa alta de aprendizagem pode fazer o processo oscilar em torno da solução procurada. Por isso é utilizado um determinado valor para a taxa de inércia com o objetivo de evitar oscilações e atingir a solução em menor tempo. Após exaustivos

testes, o melhor valor encontrado para a taxa de aprendizagem e a taxa de inércia, respectivamente, foram 90% e 50%.

A figura 10 mostra a interface destinada ao conjunto de entrada da rede MLP, ou seja, os campos destinados à digitação dos *pares lexicais singular e plural*.

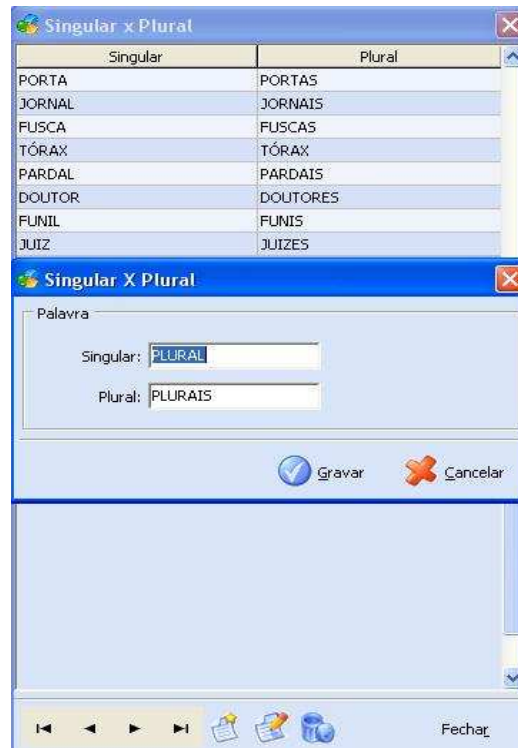


Figura 10: Tela do conjunto de entrada (singular x plural)
Fonte: Conceição, Dahmer & Silva.

Essa opção permite ao professor interferir indicando a palavra no singular e a resposta no plural. Analisando somente essa tela, entende-se equivocadamente que o sistema usa uma tabela com a palavra no singular e seu plural, processo que não permitiria ao sistema pluralizar palavras diferentes das que não estão no conjunto de entrada, além do fato de parecer que há uma base de dados de referência. Não é o caso porque esse tipo de programação não é procedimental, isto é, os dados resultantes do programa não são previamente armazenados, mas sim resultantes de um processamento em que o algoritmo depreende a desinência do par lexical e a armazena em um dos seis neurônios de saída. Quando uma palavra singular é digitada para ser pluralizada, essa desinência, pelo processamento estatístico, é nela afixada, constituindo assim a palavra flexionada em sua forma plural.

A figura 11 mostra a interface destinada ao treinamento da rede MLP. O número de iterações, em um primeiro momento, é definido visando ao menor custo

com o maior benefício, ou seja, define-se um número mínimo para que o processamento generalize as desinências de forma mais rápida.



Figura 11: Tela de treinamento da rede MLP
Fonte: Conceição, Dahmer & Silva.

Nesta fase, o treinamento da rede MLP acontece em um *loop* que conta o número de épocas especificado na configuração da rede e um *loop* interno que varre a tabela de treinamento do primeiro registro até o final da tabela. Então, para cada época, são apresentadas à rede todas as amostras do conjunto de treinamento, uma a uma. Para cada conjunto de treinamento, o algoritmo captura a penúltima e última letra da última sílaba; depois, é executado o que separa as sílabas para definir a que é tônica (Apêndice B), a fim de alimentar a entrada da rede (os três neurônios da entrada). Além disso, compara a palavra no singular com a palavra no plural e identifica a diferença entre elas. Essa diferença é a indicação de qual neurônio deve ser ativado na saída da rede, isto é, qual dos seis neurônios da saída deve ser identificado.

Após o treinamento, é gerado um *log*, conforme mostra a figura 12, na qual aparece o erro, ou o custo total, que a rede apresenta para o conjunto de treinamento processado. Esses valores serão explicados na subseção *Simulação de uso do PluralRNA* (4.6.2).

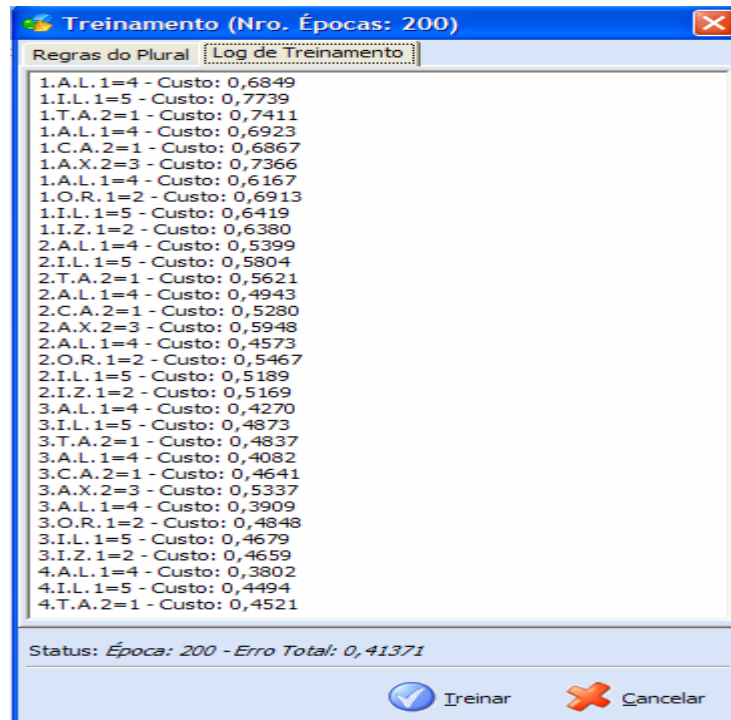


Figura 12: Tela de log do treinamento da rede MLP
Fonte: Conceição, Dahmer & Silva.

A interface destinada ao teste da rede MLP para pluralizar a palavra, conforme mostra a figura 13, contém dois campos com a sugestão de tonicidade, mas que é interativa caso seja necessária a intervenção do professor, no caso, qualquer usuário desse sistema.

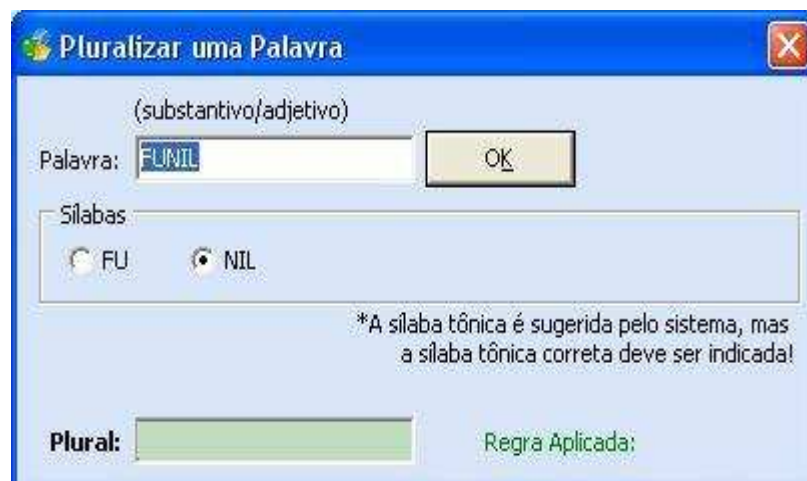


Figura 13: Tela de teste da rede MLP
Fonte: Conceição, Dahmer & Silva.

Ainda sobre a figura 13, para cada palavra digitada, o algoritmo existente no botão OK captura a penúltima e última letra da última sílaba; depois, outro algoritmo é executado com o propósito de separar as sílabas para definir qual é a tônica (Apêndice B). Disso resulta no que é mostrado na caixa Sílabas, na qual é possível a

intervenção do professor para indicar a sílaba tônica caso a que esteja marcada não seja a correta. Essas três informações são enviadas para a rede através dos três neurônios de entrada. Depois disso, a sílaba tônica na rede neural é testada. A desinência depreendida, ou seja, o neurônio ativado na saída da rede é mostrado no campo “Regra Aplicada”, que é fazer o plural da palavra alterando a sua terminação. Essa modificação da terminação acontece quando aplicada a desinência do plural, conforme a tabela NEURONIO_SAIDA.

Exemplo para a palavra “DOUTOR”. A regra morfológica dos substantivos terminados em “r” fazem o plural de acréscimo de “es”. O algoritmo fará a entrada na rede com três informações:

- 1) penúltima letra: o (valor ASCII = 79)
- 2) última letra: r (valor ASCII = 82)
- 3) classificação: oxítone (valor conforme tabela CLASSIFICACAO = 1)

Levando em consideração que a rede neural já tenha sido previamente treinada com outras palavras da mesma regra, esse teste mostrará o resultado no campo “Plural”: DOUTORES, e no campo “Regra Aplicada”: (R2) ES.

Finaliza-se esta seção com o visual da tela inicial do *software* PluralRNA, conforme figura 14, com os cinco ícones que têm funções distintas: na seqüência, **Configuração da rede neural**, **Inclusão do par lexical**, **Treinamento**, **Pluralizar palavra** e **Pluralizar utilizando Texto**. Esse último é para implementação em outro momento fora desta tese, pois se destina à área sintático-semântica.

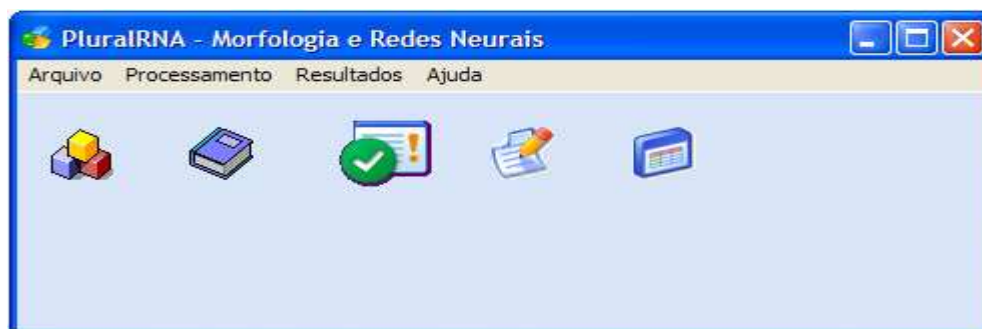


Figura 14: Tela principal do PluralRNA
Fonte: Conceição, Dahmer & Silva.

Na próxima seção, são apresentadas as avaliações feitas sobre o modelo computacional adotado: rede neural artificial de multicamadas aplicada na Morfologia do PB. Os resultados a seguir somente foram possíveis a partir do estudo feito e da utilização do protótipo desenvolvido.

4.6 APLICAÇÃO DO *CORPUS*/DADOS

São apresentadas, nesta seção, as simulações propostas para testar a rede neural. Os cenários são detalhados, visto que a variação principal é a depreensão desinencial do plural, conforme a regra morfológica correspondente. É importante também ressaltar que, para cada cenário definido nas simulações, são utilizados alguns pares lexicais para o seu processamento e posterior depreensão da desinência do plural e a sua relação com uma das seis regras morfológicas. Dessa forma, há uma resposta no *log* de treinamento (arquivo MLP.txt) para cada palavra e cada época de treinamento.

4.6.1 Treinamento *versus* aprendizado

Segundo Medeiros (2003), quando se estrutura uma rede MLP para um objetivo específico, geralmente os valores assumidos pelos pesos são aleatórios ou zerados. Quando mencionado “aprendizado”, não se está sendo tão abrangente quanto ao aprendizado referente ao ser humano. Em uma rede MLP, o processo de aprendizado restringe-se a uma espécie de treinamento por inibição ou reforço.

O aprendizado especifica valores para os pesos de forma a dar à rede MLP um sentido de processamento. Ao entrar com dados em uma rede não treinada, a saída será aleatória e não previsível. Na medida em que é mostrado à rede um padrão de entrada (exemplos de palavras no singular e no plural) também é mostrado como o resultado se apresenta na saída da rede. Para cada amostra apresentada, tem-se de indicar qual o resultado pretendido à saída, chamado par amostra-resultado ou par lexical.

Os pares lexicais que compõem a amostragem a seguir estão definidos na seção 4.5, especificamente em **Dados1**.

Foi preciso, muitas vezes, apresentar o conjunto de amostras-resultado, ou conjunto de treinamento, para a rede começar a gerar o plural das palavras.

Conforme gráfico 1, essas repetições são denominadas de épocas de treinamento.

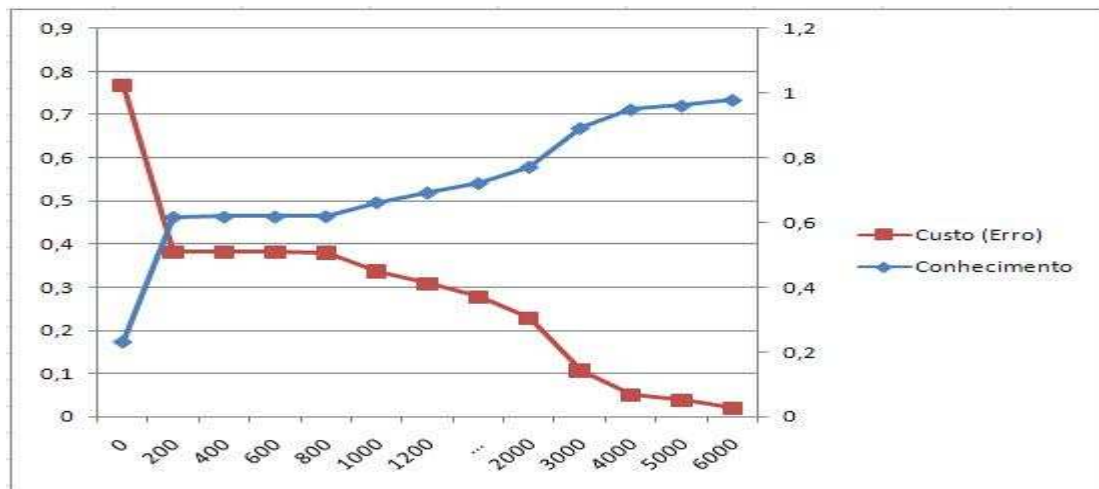


Gráfico 1: - Relação entre Treinamento *versus* Aprendizado.
Fonte: Conceição, Dahmer & Silva.

A mostra desse gráfico é somente para apresentação visual de como ocorre a aprendizagem pela rede. Por exemplo, a partir da época de treinamento 200, a rede passa a apresentar conhecimento de uma determinada regra e o custo obtido pelo retorno do *backpropagation* começa a ser minimizado. Se o erro é diminuído, possivelmente os próximos valores probabilísticos sejam trocados e a rede pode ficar apta, então, a executar o processo para o qual foi projetada. Por exemplo, a figura 15 abaixo ilustra bem o caso: foram seis regras, e a regra 4 (0,25134), correspondente ao neurônio de saída, é que foi depreendida pelos primeiros treinamentos, mas a regra 5 (0,24941) está com um valor probabilístico bem próximo.

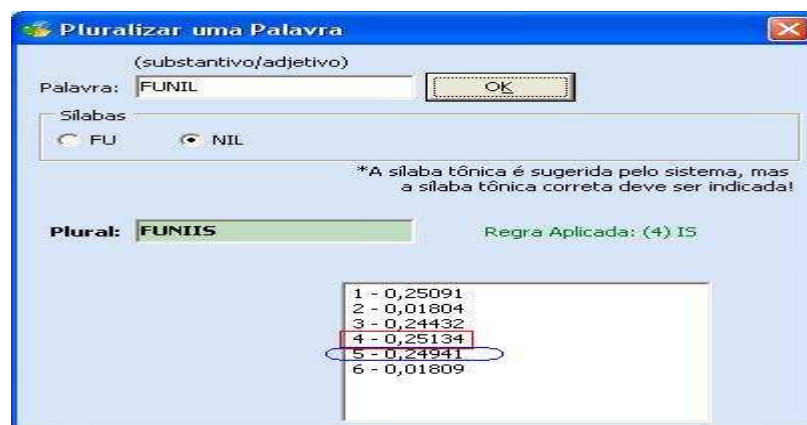


Figura 15: Tela de simulação: apresentação de erro
Fonte: Conceição, Dahmer & Silva.

Com novo treinamento da mesma rede, mais de 1300 ciclos, foi possível constatar, conforme figura 16, que o neurônio 5 (0,25582) foi o depreendido, pluralizando a palavra funil de forma adequada.

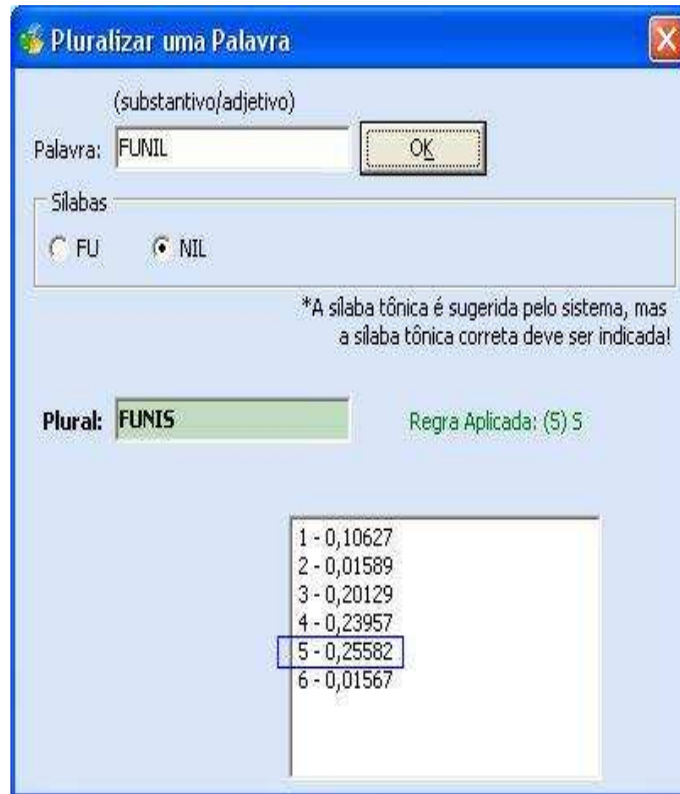


Figura 16:Tela de simulação: apresentação de acerto
Fonte: Conceição, Dahmer & Silva

Ficou claro que o maior valor (5 - 0,25582) corresponde à regra morfológica R5, em que ocorre a supressão da última letra “l” da palavra *funil* e é afixada a desinência de plural “s” em *funi*, resultando a palavra *funis*.

Na seção seguinte, algumas simulações com *corpus* mais complexo.

4.6.2 Simulação de uso do PluralRNA

Antes de uma simulação com base definida das seis regras, uma demonstração ilustrativa sobre a aprendizagem está no gráfico 2. Na horizontal é

representado o número de épocas de treinamento num intervalo de 0 a 6000. Na vertical está o valor de conhecimento adquirido, num intervalo de 0 a 1, onde zero significa nada de conhecimento e *um* significa 100% de conhecimento.

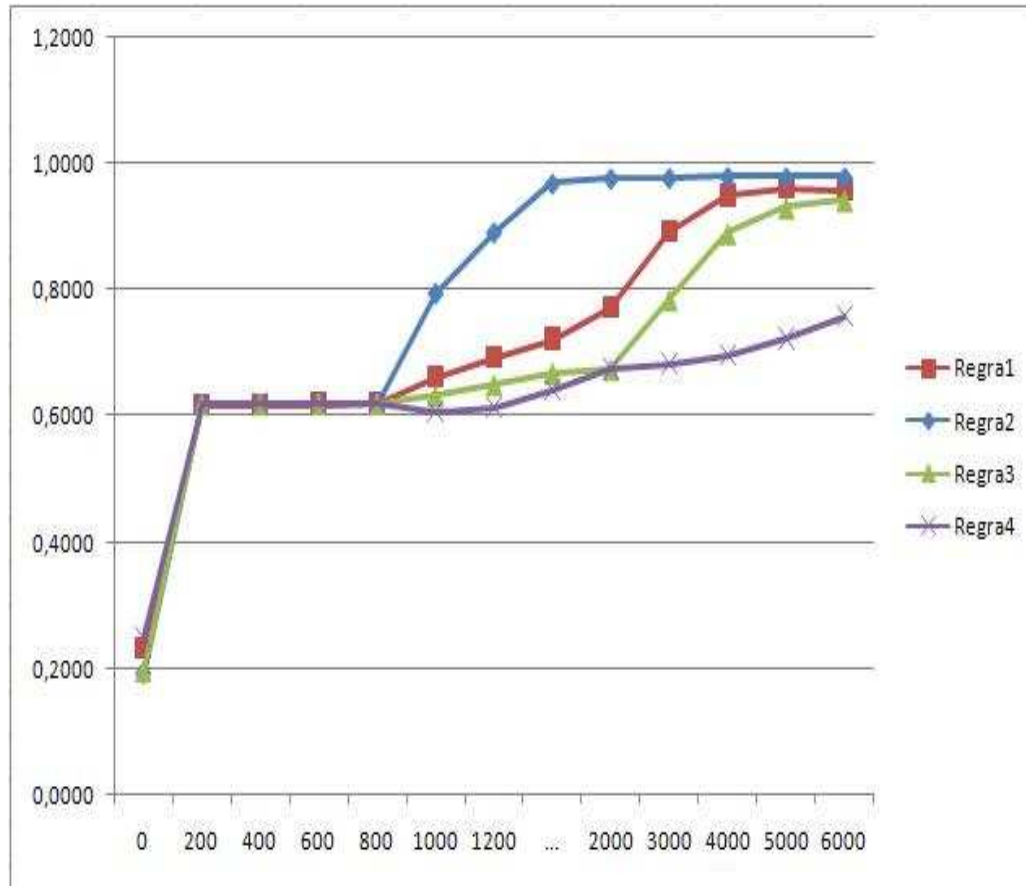


Gráfico 2: Log de aprendizado para quatro regras
Fonte: Conceição, Dahmer & Silva

Como é possível observar, esse gráfico foi gerado na fase inicial do processo de elaboração do PluralRNA, em que foram utilizadas apenas quatro regras. Convém ressaltar que a palavra *regras* está sendo usada como uma referência à morfologia, diferentemente da expressão *depreensão desinencial*, que corresponde ao processamento computacional.

Nas primeiras mil épocas de treinamento, a rede MLP não apresentava correta saída desinencial para nenhuma das quatro regras. A partir de 1200 épocas, a regra 2 começou a ser acertada, mas ainda a rede errava as outras regras. O perfeito acerto das quatro regras somente foi possível após 6000 épocas de treinamento.

Antes de simular uma rede com nível de complexidade maior, primeiro buscase a base dos pares lexicais que compõem a figura 11 (*porta, jornal, fusca, tórax,*

pardal, doutor, funil, juiz, plural e bombri!), *corpus* definido em **Dados2**. O propósito é esclarecer que essa base não é um banco de dados. As palavras são inseridas aos pares, mas não constituem banco de armazenamento lexical.

Para facilitar a visualização, são colocados na primeira linha os resultados de processamento da rede, dispostos em duas colunas: a primeira com 200 épocas e a segunda com 1200. Um abaixo do outro, correspondendo aos seus treinamentos, o léxico é confrontado coluna a coluna.

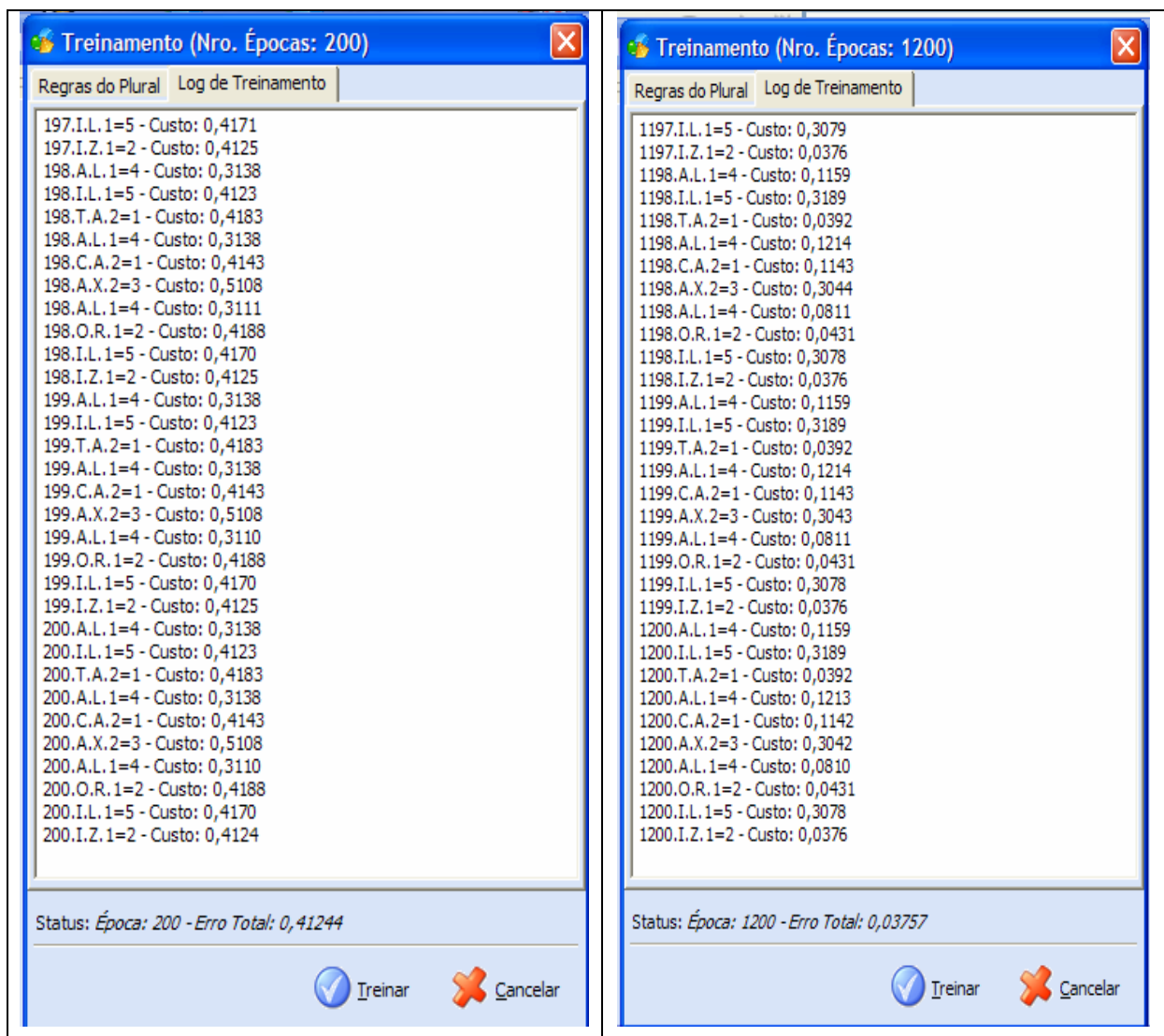


Figura 17: Tela de Treinamento com 200 e 1200 ciclos

Fonte: Conceição, Dahmer & Silva

Para o melhor entendimento do resultado, conforme figura 17, tem-se na última linha do *Treinamento das 200 épocas* a seguinte representação computacional: **200.I.Z.1=2 – Custo: 0,4124**, referente à palavra “juiz”, desdobrada abaixo:

200: quantidade de épocas/ciclos de treinamento

I: penúltima letra depreendida do par lexical

Z: última letra depreendida do par lexical

1: representa a palavra oxítone

=2: representa a regra 2 (depreensão da desinência /ES/)

Custo: 0,4124: valor atribuído que corresponde ao índice de erro na rede, também chamada taxa de aprendizagem.

Em relação às palavras paroxítonas, dois exemplos retirados da figura acima são visualizados abaixo:

200.C.A.2=1: corresponde à palavra “fusca”, em que **2** é paroxítona e “=1” é a regra 1, que depreende a desinência /S/.

200.A.X.2=3: corresponde à palavra “tórax”, em que **2** é paroxítona e “=3” é a regra 3, que mantém a palavra igual, ou seja, não há depreensão desinencial.

E para demonstrar a capacidade ou incapacidade de aprendizagem da rede, já é possível inferir que a coluna de 1200 ciclos apresenta o melhor e menor índice de erro do que a de 200 ciclos. É necessário fazer os testes de digitação para testar o desempenho do PluralRNA e também para comprovar a afirmação de que esse software não se constitui de um banco de dados, mas de valores desinenciais depreendidos pelo treinamento dos pares lexicais. Isso é feito com a digitação das mesmas palavras que entraram juntas com os seus pares no plural, depois de cada um dos dois treinamentos (200 e 1200 ciclos), conforme demonstrado na figura 18, constituída de duas colunas com 10 palavras cada uma. Ressalta-se que a finalização dos dois treinamentos pode ser visualizada pelos 10 últimos valores pertinentes a cada quantidade de ciclos, 200 e 1200, respectivamente, e estão com os seus pesos definidos para cada neurônio de entrada: a composição da penúltima e última letras depreendidas pelo PluralRNA quando da inserção dos pares lexicais.

<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="FUNIL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> FU <input checked="" type="radio"/> NIL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="FUNIIS"/> Regra Aplicada: (4) IS</p> <pre> 1 - 0,19514 2 - 0,20165 3 - 0,10092 4 - 0,29215 5 - 0,19689 6 - 0,01900 </pre>	<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="FUNIL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> FU <input checked="" type="radio"/> NIL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="FUNIS"/> Regra Aplicada: (5) S</p> <pre> 1 - 0,07293 2 - 0,01294 3 - 0,00084 4 - 0,04017 5 - 0,34246 6 - 0,00879 </pre>
<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="PORTA"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> POR <input type="radio"/> TA</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="PORTIS"/> Regra Aplicada: (4) IS</p> <pre> 1 - 0,19736 2 - 0,20329 3 - 0,10247 4 - 0,29309 5 - 0,19892 6 - 0,01951 </pre>	<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="PORTA"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> POR <input type="radio"/> TA</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="PORTAS"/> Regra Aplicada: (1) S</p> <pre> 1 - 0,99023 2 - 0,00111 3 - 0,00000 4 - 0,00035 5 - 0,21269 6 - 0,00501 </pre>
<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="FUSCA"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> FUS <input type="radio"/> CA</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="FUSCIS"/> Regra Aplicada: (4) IS</p> <pre> 1 - 0,19578 2 - 0,20140 3 - 0,10135 4 - 0,29310 5 - 0,19732 6 - 0,01911 </pre>	<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="FUSCA"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> FUS <input type="radio"/> CA</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="FUSCAS"/> Regra Aplicada: (1) S</p> <pre> 1 - 0,97755 2 - 0,00003 3 - 0,00004 4 - 0,12964 5 - 0,20802 6 - 0,00900 </pre>

<p>Pluralizar uma Palavra ✕</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="TÓRAX"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> TÓ <input type="radio"/> RAX</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="TÓRAIS"/> Regra Aplicada: (4) IS</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>1 - 0,19379 2 - 0,20076 3 - 0,09998 4 - 0,29147 5 - 0,19570 6 - 0,01870</p> </div>	<p>Pluralizar uma Palavra ✕</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="TÓRAX "/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> TÓ <input type="radio"/> RAX</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="TÓRAX"/> Regra Aplicada: (3)</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>1 - 0,00001 2 - 0,13395 3 - 0,76297 4 - 0,26743 5 - 0,08940 6 - 0,00575</p> </div>
<p>Pluralizar uma Palavra ✕</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="PARDAL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> PAR <input checked="" type="radio"/> DAL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="PARDAIS"/> Regra Aplicada: (4) IS</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>1 - 0,19442 2 - 0,20077 3 - 0,10041 4 - 0,29216 5 - 0,19615 6 - 0,01881</p> </div>	<p>Pluralizar uma Palavra ✕</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="PARDAL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> PAR <input checked="" type="radio"/> DAL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="PARDAIS"/> Regra Aplicada: (4) IS</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>1 - 0,00803 2 - 0,00061 3 - 0,11437 4 - 0,76352 5 - 0,17629 6 - 0,00804</p> </div>
<p>Pluralizar uma Palavra ✕</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="DOUTOR "/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> DOU <input checked="" type="radio"/> TOR</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="DOUTOIS"/> Regra Aplicada: (4) IS</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>1 - 0,19522 2 - 0,20219 3 - 0,10098 4 - 0,29171 5 - 0,19707 6 - 0,01904</p> </div>	<p>Pluralizar uma Palavra ✕</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="DOUTOR"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> DOU <input checked="" type="radio"/> TOR</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="DOUTORES"/> Regra Aplicada: (2) ES</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>1 - 0,00147 2 - 0,91393 3 - 0,00053 4 - 0,00061 5 - 0,37179 6 - 0,00703</p> </div>

<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="JUIZ"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> JUIZ</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="JUIIS"/> Regra Aplicada: (4) IS</p> <pre>1 - 0,19404 2 - 0,20139 3 - 0,10016 4 - 0,29115 5 - 0,19601 6 - 0,01878</pre>	<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="JUIZ"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input checked="" type="radio"/> JUIZ</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="JUIZES"/> Regra Aplicada: (2) ES</p> <pre>1 - 0,00000 2 - 0,99312 3 - 0,12985 4 - 0,00109 5 - 0,15001 6 - 0,00500</pre>
<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="BOMBRIL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> BOM <input checked="" type="radio"/> BRIL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="BOMBRIIS"/> Regra Aplicada: (4) IS</p> <pre>1 - 0,19317 2 - 0,20115 3 - 0,10047 4 - 0,29009 5 - 0,19563 6 - 0,02101</pre>	<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="BOMBRIL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> BOM <input checked="" type="radio"/> BRIL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="BOMBRIS"/> Regra Aplicada: (5) S</p> <pre>1 - 0,07293 2 - 0,01294 3 - 0,00084 4 - 0,04017 5 - 0,34246 6 - 0,00879</pre>
<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="PLURAL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> PLU <input checked="" type="radio"/> RAL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="PLURAIIS"/> Regra Aplicada: (4) IS</p> <pre>1 - 0,00755 2 - 0,00110 3 - 0,08327 4 - 0,68592 5 - 0,18466 6 - 0,00805</pre>	<p>Pluralizar uma Palavra</p> <p>(substantivo/adjetivo)</p> <p>Palavra: <input type="text" value="PLURAL"/> <input type="button" value="OK"/></p> <p>Sílabas</p> <p><input type="radio"/> PLU <input checked="" type="radio"/> RAL</p> <p>*A sílaba tônica é sugerida pelo sistema, mas a sílaba tônica correta deve ser indicada!</p> <p>Plural: <input type="text" value="PLURAIIS"/> Regra Aplicada: (4) IS</p> <pre>1 - 0,19235 2 - 0,20028 3 - 0,09964 4 - 0,28941 5 - 0,19476 6 - 0,02070</pre>

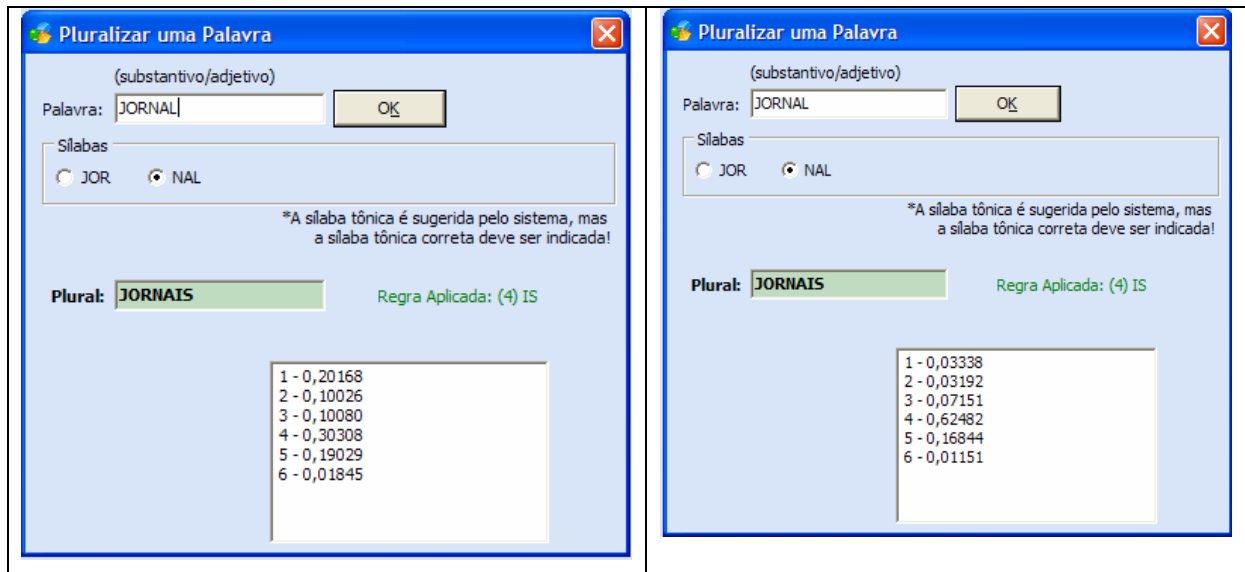


Figura 18: Tela de digitação das mesmas palavras que entraram com seus pares no plural
Fonte: Conceição, Dahmer & Silva

Percebe-se que a coluna com treinamento de 200 épocas depreendeu somente a regra R4, ao contrário da outra que, com 1200 épocas, depreendeu todas as regras que foram inseridas com os pares lexicais.

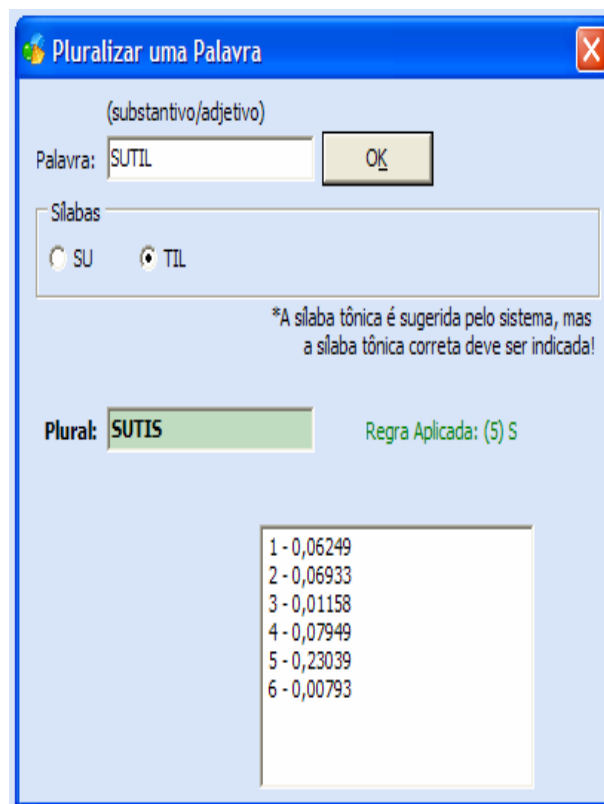


Figura 19: Tela de digitação de palavras diferentes da base de pares lexicais

Fonte: Conceição, Dahmer & Silva

Na figura 19, acima, a palavra *sutil* foi digitada para testar a rede que teve o treinamento em 1200 épocas: o PluralRNA pluralizou de forma adequada, com a regra 5 correspondente.

Outros testes foram efetuados nessa mesma rede, tendo como resultado a confirmação em todos os casos em que as palavras obedeciam a regras já definidas no programa, mas em função de não sobrecarregar a tese com figuras, optou-se por somente relatá-los.

Com a necessidade de se fazer mais simulações, buscou-se uma base de dados mais complexa, com os dados já definidos na seção 4.2 (Dados), **Dados 3**. Foram 20 pares lexicais, inclusive um deles (*limão-limões*) não estava entre as seis regras. Os resultados estão relatados abaixo a partir da quantidade de épocas definidas para cada treinamento:

- a) 5000: Somente as palavras *hífen*, *carnaval*, *fóssil*, *funil* e *azul* não tiveram as desinências depreendidas; obs.: a palavra *limão* não foi pluralizada porque não há depreensão de três letras na camada inicial do *software*, mas foi introduzida com seu par *limões* para demonstrar que um neurônio a mais nessa camada inicial possibilitaria essa depreensão.
- b) 10.000: *carnaval* e *azul* foram as únicas palavras que não conseguiram ser pluralizadas de forma adequada.
- c) 20.000: *fóssil* e *rouxinol* perderam a regra na rede, mas as outras mantiveram suas flexões.

Quanto mais complexa a base de dados da rede, mais iterações devem ser feitas até que a rede aprenda. Nesse caso, teríamos de continuar aumentando os ciclos e analisá-los um a um, o que será possível na continuidade desses estudos.

Outros testes lexicais a partir de regras já depreendidas não surpreenderam em função de que já se sabe que nessa regra, em dado treinamento, não há mudança no resultado. Não acontece quando há variações de palavras que envolvem a mesma regra, mas que, na composição de letras, é diferente e não há garantia de uma inserção desinencial positiva.

4.7 ANÁLISE DOS RESULTADOS: capacidades, limitações e perspectivas

Foram muitas simulações não descritas nesta tese, mas que certamente estariam se a quantidade de material resultante de cada processamento não fosse deixá-la muito volumosa. Isso porque a demonstração de cada fluxo de treinamento necessita de todas as figuras que compõem esse processo.

A capacidade do PluralRNA é ser um *software* de fácil manuseio e com poucas variáveis, mas que combinadas geram uma pluralidade de possibilidades que não se esgotam. Pelo contrário, suscitam muitos estudos, principalmente pela incógnita do processamento da camada de neurônios intermediária. Para os próprios cientistas da computação, essa camada pode ser comparada a uma “caixa preta”, tamanha a complexidade de seu processamento.

As evidências dessa capacidade estão demonstradas na seção 4.6, que trata do treinamento e da aprendizagem pela máquina, especificamente quanto ao resultado das simulações (subseção 4.6.2).

Podem-se descrever alguns casos pontuais, por exemplo:

- a) Como regra geral, o plural é feito pelo acréscimo do -s à forma do singular: na regra geral não foi encontrada nenhuma dificuldade, pois a rede MLP foi treinada para ativar a saída um, correspondente à regra morfológica identificada como R1, doravante somente R n , em que n se refere a qualquer das seis regras já definidas. Após esse treinamento a rede respondeu corretamente.
- b) Os substantivos terminados em *al*, *el*, *ol* e *ul* fazem o plural trocando o *l* por *-is*: Para algumas palavras, a rede MLP foi capaz de aplicar a desinência adequada. O neurônio quatro (R4) foi ativado e o resultado foi correto, por exemplo: jornal = jornais, azul = azuis.
- c) Os substantivos terminados em *l* fazem o plural de duas formas:
 - i. Quando são oxítonos, trocam o *l* por *-s*: Não foi encontrado problema. O neurônio de saída cinco (R5) foi ativado e o resultado foi obtido sem problemas.
 - ii. Quando são paroxítonos, trocam o *l* por *eis*: Não foi encontrado

problema. O neurônio de saída seis (R6) foi ativado e o resultado foi obtido corretamente.

- d) Quando não são oxítonos, os substantivos terminados em -s não mudam de forma no plural: Nenhuma dificuldade encontrada, pois a rede MLP foi treinada para ativar a saída três (R3). Após esse treinamento a rede respondeu corretamente (lápiz = lápiz, tênis = tênis).

Outro tipo de capacidade está na relação custo/benefício no que se refere à seleção do léxico que incorporou o *corpus* do software, pois foi levada em conta a exigência da participação humana no processo de verificação da convenção de determinados léxicos. Com isso foi possível fazer relação direta com o ideal de treinamento da rede para comprovar a boa capacidade de generalização e encontrar o ponto ótimo de parada dos ciclos de processamento e assim concluí-lo com êxito.

De um modo geral, a rede neural MLP apresentou-se como ótima solução para generalizar a flexão do plural das palavras e apresentou o comportamento esperado.

Quanto às suas limitações, importante destacar o aspecto arbitrário na flexão de algumas palavras, que, sem lógica alguma, apresenta um plural fora das regras, ou seja, é um plural constituído de irregularidades, por exemplo:

- a) Os substantivos terminados em *r* e *z* fazem o plural pelo acréscimo de -es: o neurônio de saída é o número dois e, para a grande maioria das palavras relacionadas com essa regra 2 (R2), a rede MLP mostrou-se como ótima solução (doutor = doutores). Porém, para palavras em que ocorrem mudanças na quantidade de sílabas e a nova palavra formada recebe o acento, a rede MLP não foi capaz de retirá-lo (juiz = juízes).
- b) Os substantivos terminados em *n* fazem o plural pelo acréscimo de -es ou -s: Para essa regra, foi assumido como neurônio de saída o acréscimo do -s (R1). Existem algumas discordâncias entre gramáticos e lingüistas do PB sobre as regras, o que foi o caso aqui: alguns indicam acréscimo de -es e outros o acréscimo de -s. Como não foi objetivo do trabalho estudar o porquê dessa diferença, assumiu-se pelo mais simples: acréscimo de -s pelo neurônio de saída número um (R1). Após a rede convergir, o plural das palavras foi gerado sem problemas. Exceção feita para casos onde

acontece a mudança de acentuação e tonicidade da sílaba, por exemplo: hífen = hífens, pois o processamento manteve o acento da palavra no singular.

- c) Os substantivos monossílabos e os oxítonos terminados em -s fazem o plural pelo acréscimo de -es: a mesma situação enfrentada anteriormente, ocorrendo mudança de acentuação e tonicidade na gramática do PB, por exemplo: mês = meses, acento mantido no resultado, mas que não corresponde às regras morfológicas do PB.
- d) Os substantivos terminados em x não mudam de forma no plural: Nenhuma dificuldade encontrada, pois a rede MLP foi treinada para ativar a saída três (R3) e a rede respondeu corretamente (tórax = tórax). O problema é a discordância entre os gramáticos e lingüistas, não necessariamente nessa ordem. Uns definem a regra dessa forma e outros definem a plural como *tóraces*, o que já foi apresentado anteriormente.
- e) Os substantivos terminados em ão podem formar o plural de três maneiras:
 - i. Pelo acréscimo de -s;
 - ii. Pela transformação do ão em ães;
 - iii. Pela transformação do ão em ões.

A maneira escolhida para gerar o plural das palavras terminadas em ão foi a primeira (a) (*irmão = irmãos*), mas alguns problemas foram identificados, por exemplo: *alemão = alemãos*. Como esse problema constitui irregularidade lingüística no tratamento do plural, optou-se por não tratá-la nesta tese, mas abrir a possibilidade de estudos futuros sobre essa complexa flexão.

Algumas questões foram identificadas, como o caso das regras de acentuação e a mudança na tonicidade da palavra, o que de certo modo dificultou o resultado 100%. Mas algumas implementações no trabalho deixaram-no mais confiável em termos de resultados esperados, por exemplo, o algoritmo para identificar a sílaba tônica da palavra e o algoritmo para separação de sílabas.

Outra limitação é quanto à quantidade de informação na rede. Conforme percebido nos comentários das simulações, quanto mais pares lexicais foram apresentados para a rede MLP, mais longo foi o período para alcançar o ideal de aprendizado. Em uma delas, a rede MLP alcançou bom resultado após 15.000

épocas de treinamento para um conjunto de par amostra de 6 diferentes regras do plural, o que demandou certo esforço computacional, mas que depois teve 100% de acerto.

Mas como fazer para a máquina depreender e aplicar adequadamente a desinência para cada léxico?

Para responder essa pergunta, é necessário retomar as questões iniciais desta tese:

a) Se as regras morfológicas do PB podem ser processadas computacionalmente, ou seja, é possível tratar o plural de maneira sistemática e mecânica, fundamentado em regularidades, mesmo sabendo da existência de casos específicos de exceção a regras? b) É possível a geração de um programa computacional que permita fazer esse tipo de tratamento lingüístico?

A hipótese que responde parcialmente a primeira pergunta é de que realmente é possível fazer esse estudo fundamentado em regularidades. O problema foi com a existência de irregularidades, havendo nesta tese a necessidade de uma delimitação: a exclusão dos casos irregulares, o que se constitui como perspectiva para futuros estudos. Optou-se somente pelas flexões regulares, porque as divergências de convenções gramaticais dependem tanto de conhecimento extralingüístico e aceitabilidade de vários tipos de flexão, como também de estudos etimológicos.

A perspectiva referida anteriormente é em relação ao próprio *software*: teríamos de gerar mais neurônios na camada de entrada e aumentar a da intermediária, interligando a questão lingüística, especificamente os estudos etimológicos latinos para cada léxico de entrada no sistema. Esse estudo já apontaria para a resolução do problema da pluralização de palavras com a terminação em *ão*.

Mas a perspectiva mais promissora é quanto à possibilidade de um programa específico que atenda a essas exigências flexionais.

5 CONCLUSÃO

Uma visão panorâmica lingüístico-computacional foi apresentada na introdução deste estudo com o propósito de situar o leitor para o assunto-objeto da tese: um algoritmo ilustrativo do plural em uma simulação computacional do comportamento humano para as questões lingüísticas, notadamente a flexão de número do PB.

No segundo capítulo (*Sobre o plural na Morfologia Lingüística e o número em português*), foram apresentados os conceitos de *Morfologia Lingüística*, o estudo do plural, com as suas regularidades, irregularidades e o próprio processo de pluralização do PB. As abordagens descritivo-normativas e a definição do tópico lingüístico, o plural, foram também destacadas nesta tese. Sua contribuição para esse estudo está na medida em que separa metodologicamente as duas visões lingüísticas: uma é a descrição científica dos morfemas que compõem uma dada língua, e a outra para definir as regras convencionadas pela gramática e que são, momentaneamente, imutáveis, mas aparentemente de fácil tratamento. Outra contribuição foi a apresentação dos problemas gerados pelas regras de flexão de número do PB.

No capítulo 3 (*Na interface Lingüística/Computação*), os conhecimentos computacionais sobre a disciplina Lingüística nesse meio tecnológico, como também a Morfologia, foram apresentados em três seções distintas: Lingüística, Morfologia e Modelagem Computacionais E para fazer referência ao título desta tese, a seção Modelagem Computacional subdividiu-se em Ciências Cognitivas, Inteligência Artificial e especificamente o Algoritmo, tanto em linguagem natural como o ilustrativo, representado em sua forma de fluxograma.

Como o foco é a relação entre o comportamento do número em PB e o algoritmo criado para verificar se ele corresponde às hipóteses geradas no início desta tese, o capítulo 4 mostra como os pares lexicais de singular e plural são armazenados no PluralRNA, como também mostra os estudos para refutar ou corroborar a hipótese principal sobre a capacidade do software em simular o comportamento humano em relação às regras gramaticais previamente estabelecidas. A contribuição deste capítulo está na visibilidade estrutural do caminho pelo qual o algoritmo percorre dentro deste estudo científico, representando de forma matemático-computacional as sinapses cerebrais humanas. Nas seções

4.1 e 4.2, o *Corpus* e os Dados forneceram a base lingüístico-computacional para a estrutura desta tese. Na seção 4.3, a Aplicação Metodológica do Modelo Computacional apresentou os passos para a operacionalização do software e também a sua configuração de arquitetura da RNA, o que constitui uma grande contribuição para a Lingüística, porque interage com o mundo computacional de forma efetiva através de um objeto gramatical de muita relevância no PB. A Implementação do Modelo, na seção 4.4, definiu a opção por um modelo de RNA e também mostrou a programação do PluralRNA na Descrição do Protótipo, seção 4.5, o que demonstrou a capacidade desse software para seguir os estudos de flexão de número, podendo se estender a flexões de outro tipo.

Ainda com relação ao capítulo 4, a finalização de todo o processo: a Aplicação do Corpus/Dados na seção 4.7, em que primeiramente são destacados o treinamento e o aprendizado para depois partir às simulações do PluralRNA. De maneira bem didático-metodológica, o material é apresentado, pormenorizando situações complexas para o entendimento geral.

Entende-se que esta tese está com o objetivo alcançado. Primeiro porque foi criado um programa para que fosse possível a máquina depreender a desinência de plural e aplicá-la a cada nova palavra inserida em seu banco de dados.

Segundo, com relação à configuração de rede, as escolhas pelo melhor modelo computacional ainda são feitas de forma empírica, embora existam algumas heurísticas que conduzam a opções mais acertadas. Em geral, o procedimento requer grande experiência dos projetistas, o que só é obtido com o esforço ao longo do tempo em várias aplicações. A solução empírica apresentada neste trabalho com certeza não é a única possível, mas algum esforço já foi gasto para encontrar a melhor solução.

Não se pode deixar de destacar a natureza da pesquisa, pois ela gerou conhecimentos novos e úteis para o avanço da ciência e para aplicação prática, dirigidos à solução do problema de reconhecimento de linguagem natural.

Como as hipóteses foram feitas com o propósito de simular o comportamento cerebral dos humanos nos computadores, entende-se que o avanço para o objetivo ser alcançado foi considerável, uma vez que até o modelo neuronal foi definido para que as simulações comesçassem a se concretizar.

O ideal é que o próprio algoritmo se dividisse em uma regra geral e numa listagem dos casos que não se enquadrem nessa mesma regra, a geral. Mas o ideal

depende de outras circunstâncias.

Ainda que não tivesse sido objeto deste trabalho, a aplicabilidade do PluralRNA para a pesquisa parece evidente, tanto pelo aspecto gramatical como lingüístico e computacional, pois ele tem potencial de aplicação e representa uma perspectiva de continuidade de pesquisa. Poderia, inclusive, resolver o problema das irregularidades.

REFERÊNCIAS

ALMEIDA, Napoleão Mendes. **Gramática Metódica da Língua Portuguesa**. 34 ed. São Paulo: Saraiva, 1986.

BASÍLIO, Margarida. **Teoria Lexical**. 7 ed. São Paulo: Editora Ática, 2002.

BECHARA, Evanildo. **Moderna Gramática Portuguesa**. 37. ed. Rio de Janeiro: Lucerda, 2006.

BILOBROVEC, Marcelo; MARÇAL, Rui F. Martins; PILATTI, Luiz Alberto. **Aplicações de Redes Neurais em Simulações**. In: XI SIMPEP – SIMPÓSIO DE ENGENHARIA DE PRODUÇÃO, XI., 2004, Bauru. **Anais...** Bauru, 2004.

BLOOMFIELD, L. **Language**. Chicago, The University of Chicago Press, 1984.

BORGES, José Antônio, **Sistema de Síntese de Fala para Português. Sintetizador DOSVOX**. Rio de Janeiro: NCE/UFRJ, 2007.

CÂMARA JR, J. Mattoso, **Estrutura da língua portuguesa**. 15 ed, Vozes, Petrópolis, RJ, 1985.

_____. **História e estrutura da língua portuguesa**. 2 ed, Rio de Janeiro: Editora Padrão, 1976.

CAUDILL, M & BUTTER, C. **Naturally intelligent systems**. Cambridge: MIT, 1991

CONRADS, P. A. e ROEHL Jr. E. A. **Comparing Physics-Based and Neural Network Models for Simulating Salinity, Temperature, and Dissolved Oxygen in a Complex, Tidally Affected River Basin**. Estados Unidos, 1999. Disponível em http://smig.usgs.gov/SMIG/features_0302/beaufort.html. Acesso em 03 de março de 2008.

COSTA, Sônia Bastos Borba. **A Lingüística e os Estudos de Linguagem rumo ao século XXI**. 2003. Disponível em <http://www.prohpor.ufba.br/alinguis.html>. Acesso em 07 de janeiro de 2009.

CUNHA, Celso Cunha e CINTRA, Lindley. **Nova Gramática do Português Contemporâneo**. 3. ed. Rio de Janeiro: Nova Fronteira, 2001.

ELMAN, J e PLUNKETT, Kim. **Exercises in Rethinking Innateness**: a handbook for connectionist simulations. Cambridge, MA: MIT Press, 1997

ELSON, B e PICKETT, V. **Introdução à morfologia e à sintaxe**. Petrópolis, RJ: Vozes, 1973.

GLEASON, H. A. Jr. **Introdução à lingüística descritiva**. Lisboa: Fundação Calouste Gulbenkian, 1978.

GOUVEIA, Paulo D. F. **Divisão Silábica Automática do Texto Escrito e Falado**. In: PROPOR 2000 – 5º ENCONTRO PARA O PROCESSAMENTO COMPUTACIONAL DA LÍNGUA PORTUGUESA ESCRITA E FALADA, 5., 2000, São Paulo. **Anais...** São Paulo, 2000.

HAYKIN, S. **Neural networks: a comprehensive foundation**. New York: Macmillan College Publishing Company, 1994.

HOCKETT, Charles F. **Curso de Lingüística Moderna**. Buenos Aires: Eudeba, 1973.

HOUAISS, Antônio. **Dicionário Houaiss da língua portuguesa**. Rio de Janeiro: Objetiva, 2001.

JUDE (**UML Modeling Tool**, 2006), 2007. Disponível em: < <http://jude.change-vision.com/>>. Acesso em: 13 abr. 2008.

KOVACS, Zsolt László. **Redes neurais artificiais: fundamentos e aplicações: um texto básico**. 3. ed. São Paulo: Livraria da Física, 2002.

LOPES, Edward. **Fundamentos da lingüística contemporânea**. São Paulo: Editora Cultrix, 1993.

LUDWIG JR, Oswaldo; MONTGOMERY, Eduardo. **Redes neurais: fundamentos e aplicações com programas em C**. Rio de Janeiro: Ciência Moderna, 2007.

LUFT, Celso Pedro. **Moderna Gramática Brasileira**. 2. ed. São Paulo: Globo, 2002.

LUGER, George F., **Inteligência artificial: estruturas e estratégias para a solução de problemas complexos**. Porto Alegre: [s.n.], 2004.

MARCONI, M. A.; LAKATOS, E. M. **Fundamentos de metodologia científica**. 4. ed. São Paulo: Atlas, 2001.

MARTINET, André. **Elementos de lingüística geral**. 3 ed. Lisboa. Livraria Sá da Costa Editora, 1971.

MATTOS E SILVA, Rosa Virgínia. **O português arcaico** : fonologia, morfologia e sintaxe. São Paulo: Contexto, 2006. p. 105.

MCCULLOCH, W & PITTS, W. **A Logical Calculus of the ideas Immanent in Nervous**. Activity. Bull. Math. Biophysics. [s.l.], n. 5, p. 115-133, 1943.

MEDEIROS, Luciano Frontino de. **Redes Neurais em Delphi**. Florianópolis: Visual Books, 2003.

MEDINA, Marco. & FERLIG, Cristina. **Algoritmos e Programação**: teoria e prática. São Paulo: Novatec, 2005.

MORDEBE. **Portal de Língua Portuguesa**. Disponível em <http://www.portaldalinguaportuguesa.org>. Acesso em 24 de dezembro de 2009.

NASCIMENTO, Maria Fernanda Bacelar do. *et al.* **Léxico Multifuncional Computadorizado do Português Contemporâneo**. Lisboa: Universidade de

Lisboa, 2000. Disponível em http://www.clul.ul.pt/sectores/projecto_lmcp.html. Acesso em: março 2008.

NEVES, Maria Helena de Moura. **A gramática : história, teoria e análise, ensino**. São Paulo: UNESP, 2002.

_____. **Guia de uso do Português : confrontando regras e usos**. São Paulo: UNESP, 2003. p. 137 e 138.

O'REILLY, R. C., MUNAKATA, Y. **Computational explorations in cognitive neuroscience**. Cambridge: MIT Press, 2000.

OMG (Object Management Group) - UML (**Unified Modeling Language**), 2006. Disponível em <http://www.uml.org>. Acesso em: mar. 2008.

OSÓRIO, Fernando; BITTENCOURT, João Ricardo. **Sistemas Inteligentes baseados em Redes Neurais Artificiais aplicados ao Processo de Imagens**. In: WORKSHOP DE INTELIGÊNCIA ARTIFICIAL, I, 2000, Santa Cruz do Sul. **Anais...** Santa Cruz do Sul, 2000.

PINKER, Steven. **The Language Instinct**. London: Penguin, 1994. p. 130-131.

_____. **Words and rules** : The ingredients of language. New York: Basic Books, 1999.

PLAUT, D. C., MCCLELLAND, J. L., SEIDENBERG, M. S., and PATTERSON, K. (1996). **Understanding normal and impaired word reading** : Computational principles in quasi-regular domains. *Psychological Review*, 103, 56-115.

ROHDE, Douglas. **L E N S : The light, efficient network simulator**. Pittsburgh: Carnegie Mellon University, 1999.

ROSENBLAT, F. The Perceptron: **A probabilistic model for information storage and organization in the brain**, *Psychological Review*, vol 65, 1957. p. 386-408.

RUMELHART, D. e MCCLELLAND, J.; PDP Research Group. **Parallel Distributed Processing**. Cambridge: Mass MIT Press, 1986, v. I.

SANTOS, Diana. **Processamento computacional da Língua Portuguesa** <http://www.linguateca.pt>. 1997. Acesso em 07 de janeiro de 2009.

SAUSSURE, Ferdinand. **Curso de lingüística Geral**. São Paulo: Cultrix, 7. ed. 1975.

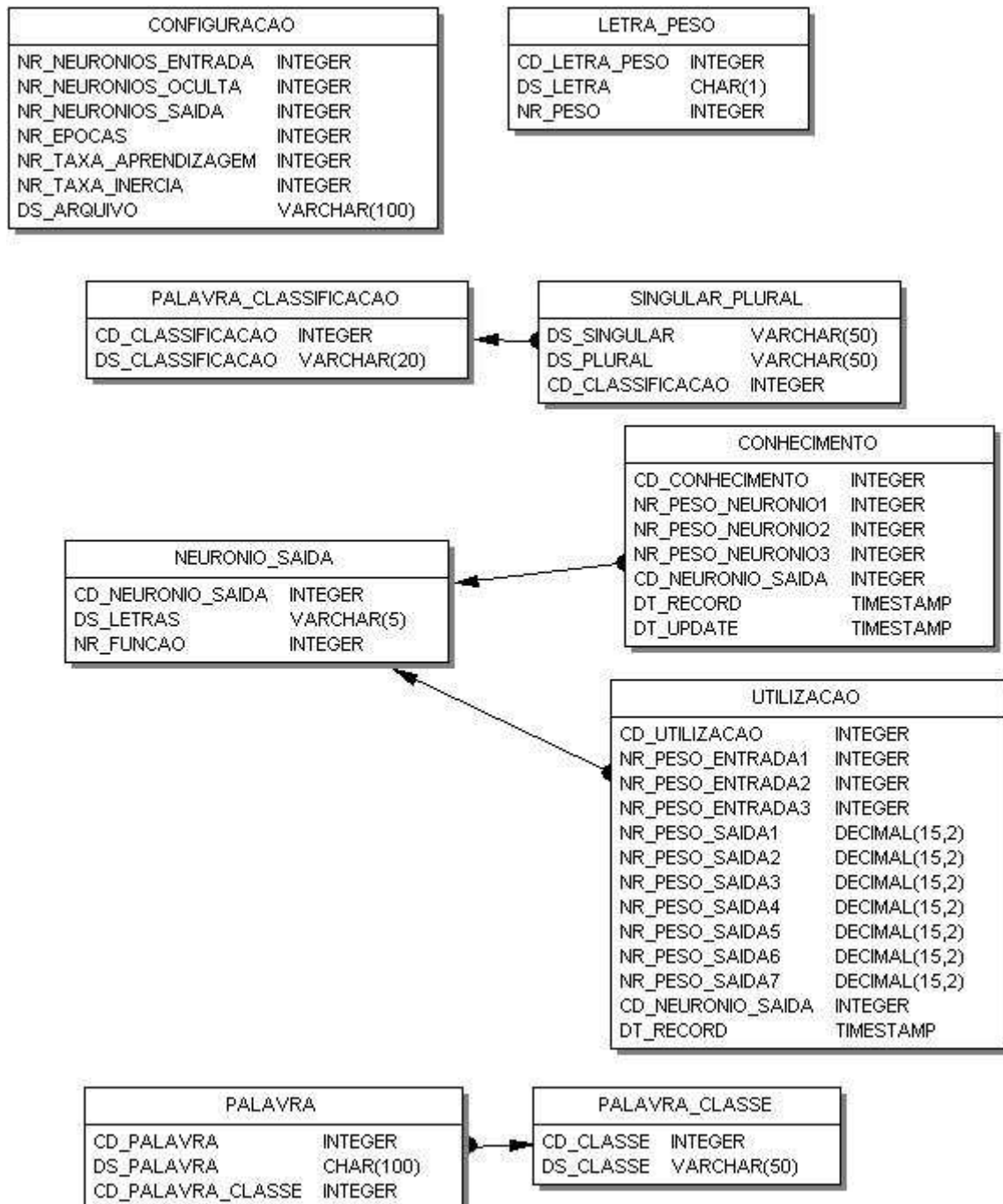
SILVA, Bento Carlos Dias da. **Estudo lingüístico-computacional da linguagem**. Letras de Hoje. Porto Alegre: v. 41, nº 2, p. 103-138. junho 2006. Disponível em <http://revistaseletronicas.pucrs.br/ojs/index.php/fale/article/viewFile/597/428>. Acesso em: 25 nov 2008.

WINOGRAD, T. **SHRDLU** : natural language understanding computer program. Disponível em http://en.wikipedia.org/wiki/Terry_Winograd. Acesso em 07 de Janeiro de 2009.

YONG, C. K. e LIM, C. M. **An Integrated Water Quality Monitoring System using Artificial Neural Networks**. Singapura, 2001. Disponível em http://www.np.edu.sg/~yck/nn_waterquality.pdf. Acesso em: mar. 2008.

YOUNG, Stephen.; CONCAR, David. **These calls were made for learning**. New Scientist, november 21, p. 2-8, 1992.

APÊNDICE A – Diagrama ER da base de conhecimento



APÊNDICE B – Algoritmos: Separar Sílabas e Identificar a Sílabas Tônicas

```
unit uFuncoes;
```

```
interface
```

```
uses SysUtils;
```

```
function Ilf( Expressao : Variant; ParteTRUE, ParteFALSE : Variant ) : Variant;
```

```
function Contem( letra : String; texto : String ) : Boolean;
```

```
function Right( Source : String; Lengths : Integer ) : String;
```

```
function DescobreTonica( palavra : String ) : Integer;
```

```
function Separar_Silabas( _word : String ) : String;
```

```
//var
```

```
implementation
```

```
function Ilf( Expressao : Variant; ParteTRUE, ParteFALSE : Variant ) : Variant;
```

```
begin
```

```
  if Expressao then
```

```
    Result := ParteTRUE
```

```
  else
```

```
    Result := ParteFALSE;
```

```
end;
```

```
function Contem( letra : String; texto : String ) : Boolean;
```

```
begin
```

```
  result := false;
```

```
  if (Pos( letra, texto ) > 0) then result := true;
```

```
end;
```

```
function Right( Source : String; Lengths : Integer ) : String;
```

```
begin
```

```
  Result := Copy( Source, (Length(Source) - Lengths) + 1, Lengths );
```

```

end;

function DescobreTonica( palavra : String ) : Integer;
var
  p      : Integer;
  c      : Char;
  posAcento : Integer;
  estado  : Integer; { Indica estado corrente do diagrama }
const
  vogais : set of char = ['a','e','i','o','u','y','w','ü'];
  acentos : set of char = ['á','â','ã','à','é','ê','í','ó','ô','õ','ú'];
begin
  descobreTonica := 0;
  palavra      := ansiLowerCase (palavra);
  posAcento    := 0;

  for p := Length(palavra) downto 1 do
  begin
    c := palavra[p];

    if c in acentos then
    begin
      descobreTonica := p;
      Exit;
    end;

    if (posAcento = 0) and (c in vogais) then
      posAcento := p;
    end;

    if posAcento = 0 then Exit; // palavra não tem vogais

  estado := 0;
  p      := length (palavra);

  while p > 0 do
  begin
    c := palavra[p];

```

```

case estado of
  0 : case c of
    'a', 'e', 'o':
      begin
        descubreTonica := p;
        estado := 1;
      end;
    'u':
      begin
        descubreTonica := p;
        estado := 10;
      end;
    'i':
      begin
        descubreTonica := p;
        estado := 20;
      end;
    's' : estado := 0;
    'm' : estado := 40;
  else
    estado := 30;
  end;

{--- palavras com última vogal 'a', 'e', 'o' ---}
  1 : if c = 'u' then
    estado := 4
  else
    if c in vogais then
      begin
        descubreTonica := p;
        p := 0;
      end
    else estado := 2;
  end;

  2 : case c of
    'i', 'u':
      begin

```

```
    descubreTonica := p;  
    estado := 3;  
end;  
'a', 'e', 'o':  
begin  
    descubreTonica := p;  
    p := 0;  
end;  
end;
```

```
3 : begin  
    if c in ['a', 'e', 'o'] then  
        begin  
            descubreTonica := p;  
            p := 0;  
        end  
    else  
        if c = 'u' then  
            estado := 5  
        else p := 0;  
    end;  
end;
```

```
4 : begin  
    if c in ['g', 'q'] then  
        begin  
            descubreTonica := p + 2;  
            estado := 2;  
        end  
    else  
        begin  
            descubreTonica := p + 1;  
            p := 0;  
        end;  
    end;  
end;
```

```
5 : begin  
    if c in ['g', 'q'] then  
        descubreTonica := p + 2
```

```

else
  descobreTonica := p + 1;
  p := 0;
end;

```

```
{--- palavras com última vogal 'u' ---}
```

```

10 : begin
  if c in ['a','e','i','o'] then
    descobreTonica := p;
    p := 0;
  end;

```

```
{--- palavras com última vogal 'i' ---}
```

```

20 : begin
  if c in ['a','e','o'] then
    descobreTonica := p
  else
    if c = 'u' then
      estado := 21
    else p := 0;
  end;

```

```

21 : begin
  if c in ['g','q'] then
    descobreTonica := p + 2
  else descobreTonica := p + 1;

  p := 0;
end;

```

```
{--- palavras terminadas por consoante, exceto 's', 'm' ---}
```

```

30 : if c in vogais then
  begin
    descobreTonica := p;
    p := 0;
  end;

```

```
{--- palavras terminadas por 'm' ---}
```



```

end;

_point := Pos('VCV',_word2);

while (_point <> 0) do
begin
_word := Copy(_word ,1,_point)+'-'+RIGHT(_word ,Length(_word )-_point);
_word2 := Copy(_word2,1,_point)+'-'+RIGHT(_word2,Length(_word2)-_point);
_point := Pos('VCV',_word2);
end;

_point := Pos('VCCV',_word2);

while _point <> 0 do
begin
C1 := UPPERCase(Copy(_word,_point+1,1));
C2 := UPPERCase(Copy(_word,_point+2,1));

if ( ( not Contem(C1,'JLHMNQRSXZ') ) and Contem(C2,'LR') ) or ( (Pos(C1,'CLNPST')>0) and (Pos(C2,'H')>0) ) then
begin
// V-CCV
_word := Copy(_word ,1,_point)+'-'+RIGHT(_word ,Length(_word )-_point);
_word2 := Copy(_word2,1,_point)+'-'+RIGHT(_word2,Length(_word2)-_point);
end
else
begin
// VC-CV
_word := Copy(_word ,1,_point+1)+'-'+RIGHT(_word ,Length(_word )-( _point+1));
_word2 := Copy(_word2,1,_point+1)+'-'+RIGHT(_word2,Length(_word2)-( _point+1));
end;
_point := Pos('VCCV',_word2);
end;

_point := Pos('VCCCV',_word2);
while _point <> 0 do
begin
C1 := UPPERCase(Copy(_word,_point+2,1));
C2 := UPPERCase(Copy(_word,_point+3,1));

```

```

if ( (NOT Contem(C1,'JLHMNQRSXZ') ) and Contem(C2,'LR') ) or ( Contem(C1,'CLNPST') and (C2='H')) then
begin
  _word := Copy(_word ,1, _point+1)+'-'+RIGHT(_word ,Length(_word )-( _point+1));
  _word2 := Copy(_word2,1, _point+1)+'-'+RIGHT(_word2,Length(_word2)-( _point+1));
end
else
begin
  _word := Copy(_word ,1, _point+2)+'-'+RIGHT(_word ,Length(_word )-( _point+2));
  _word2 := Copy(_word2,1, _point+2)+'-'+RIGHT(_word2,Length(_word2)-( _point+2));
end;

_point := Pos('VCCCV',_word2);
end;

_point := Pos('VCCCCV',_word2);
while _point <> 0 do
begin
  _word := Copy(_word ,1, _point+2)+'-'+RIGHT(_word ,Length(_word )-( _point+2));
  _word2 := Copy(_word2,1, _point+2)+'-'+RIGHT(_word2,Length(_word2)-( _point+2));
  _point := Pos('VCCCCV',_word2);
end;

_point := Pos('VV',_word2);
while _point <> 0 do
begin
  if
      not
      Pos(''+UPPERCase(Copy(_word,_point
,2))+','!'!AI!AO!ÃO!ãO!AU!E!EU!OE!ÕE!õE!OI!OO!OU!' +iif(_point<>1,iif(Pos(UPPERCase(Copy(_word,_point-
1,1)), 'GQ')>0,'UA!ÜA!üA!UE!ÛE!üE!UI!ÛI!üI!UO!ÜO!üO!UU!ÛU!üU!',","))>0 then
begin
  _word := Copy(_word ,1, _point)+'-'+RIGHT(_word ,Length(_word )-_point);
  _word2 := Copy(_word2,1, _point)+'-'+RIGHT(_word2,Length(_word2)-_point);
end;

if Pos('VV',RIGHT(_word2,Length(_word2)-_point)) <> 0 then
  _point := _point + Pos('VV',RIGHT(_word2,Length(_word2)-_point))
else _point := 0;
end;
end;

```



```
result := (_word);
```

```
end;
```

```
end.
```

APÊNDICE C – Algoritmo do funcionamento do PluralRNA

```
unit uFuncoes;
```

```
interface
```

```
uses SysUtils;
```

```
//var
```

```
implementation
```

```
function If( Expressao : Variant; ParteTRUE, ParteFALSE : Variant ) : Variant;
```

```
begin
```

```
  if Expressao then
```

```
    Result := ParteTRUE
```

```
  else
```

```
    Result := ParteFALSE;
```

```
end;
```

```
function Contem( letra : String; texto : String ) : Boolean;
```

```
begin
```

```
  result := false;
```

```
  if (Pos( letra, texto ) > 0) then result := true;
```

```
end;
```

```
function Right( Source : String; Lengths : Integer ) : String;
```

```
begin
```

```
Result := Copy( Source, (Length(Source) - Lengths) + 1, Lengths );
```

```
end;
```

```
function DescobreTonica( palavra : String ) : Integer;
```

```
var
```

```
  p    : Integer;
```

```
  c    : Char;
```

```
  posAcento : Integer;
```

```
  estado   : Integer; { Indica estado corrente do diagrama }
```

```
const
```

```
  vogais : set of char = ['a','e','i','o','u','y','w','ü'];
```

```
  acentos : set of char = ['á','â','ã','à','é','ê','í','ó','ô','õ','ú'];
```

```
begin
```

```
  descobreTonica := 0;
```

```
  palavra := ansiLowerCase (palavra);
```

```
  posAcento := 0;
```

```
  for p := Length(palavra) downto 1 do
```

```
    begin
```

```
      c := palavra[p];
```

```
      if c in acentos then
```

```
        begin
```

```
          descobreTonica := p;
```

```
          Exit;
```

```
        end;
```

```
      if (posAcento = 0) and (c in vogais) then
```

```
        posAcento := p;
```

```
end;
```

```
if posAcento = 0 then Exit; // palavra não tem vogais
```

```
estado := 0;
```

```
p := length (palavra);
```

```
while p > 0 do
```

```
begin
```

```
  c := palavra[p];
```

```
case estado of
```

```
  0 : case c of
```

```
    'a', 'e', 'o':
```

```
      begin
```

```
        descobreTonica := p;
```

```
        estado := 1;
```

```
      end;
```

```
    'u':
```

```
      begin
```

```
        descobreTonica := p;
```

```
        estado := 10;
```

```
      end;
```

```
    'i':
```

```
      begin
```

```
        descobreTonica := p;
```

```
        estado := 20;
```

```
      end;
```

```
    's' : estado := 0;
```

```
'm' : estado := 40;

else

    estado := 30;

end;

{--- palavras com última vogal 'a', 'e', 'o' ---}

1 : if c = 'u' then

    estado := 4

else

    if c in vogais then

        begin

            descubreTonica := p;

            p := 0;

        end

    else estado := 2;

end;

2 : case c of

    'i', 'u':

        begin

            descubreTonica := p;

            estado := 3;

        end;

    'a', 'e', 'o':

        begin

            descubreTonica := p;

            p := 0;

        end;

end;

end;
```

```
3 : begin
    if c in ['a', 'e', 'o'] then
        begin
            descobreTonica := p;
            p := 0;
        end
    else
        if c = 'u' then
            estado := 5
        else p := 0;
        end;
end;
```

```
4 : begin
    if c in ['g', 'q'] then
        begin
            descobreTonica := p + 2;
            estado := 2;
        end
    else
        begin
            descobreTonica := p + 1;
            p := 0;
        end;
end;
```

```
5 : begin
    if c in ['g', 'q'] then
        descobreTonica := p + 2
    else
```

```

    descobreTonica := p + 1;

    p := 0;

end;

{--- palavras com última vogal 'u' ---}

10 : begin

    if c in ['a','e','i','o'] then

        descobreTonica := p;

        p := 0;

    end;

{--- palavras com última vogal 'i' ---}

20 : begin

    if c in ['a','e','o'] then

        descobreTonica := p

    else

        if c = 'u' then

            estado := 21

        else p := 0;

    end;

21 : begin

    if c in ['g','q'] then

        descobreTonica := p + 2

    else descobreTonica := p + 1;

    p := 0;

end;

```

```

{--- palavras terminadas por consoante, exceto 's', 'm' ---}

30 : if c in vogais then

    begin

        descubreTonica := p;

        p := 0;

    end;

{--- palavras terminadas por 'm' ---}

40 : if c in ['i', 'o', 'u'] then

    begin

        descubreTonica := p;

        p := 0;

    end

else

    if c in ['a', 'e'] then

        begin

            descubreTonica := p;

            estado := 1;

        end

        else estado := 0;

    end;

    p := p - 1;

end;

end;

function Separar_Silabas( _word : String ) : String;

var

    _word2 : string;

```



```

while _point <> 0 do

begin

C1 := UPPERCase(Copy(_word,_point+1,1));

C2 := UPPERCase(Copy(_word,_point+2,1));

if ( ( not Contem(C1,'JLHMNQRSXZ') ) and Contem(C2,'LR') ) or ( (Pos(C1,'CLNPST')>0) and (Pos(C2,'H')>0) ) then

begin

// V-CCV

_word := Copy(_word ,1,_point)+'-'+RIGHT(_word ,Length(_word )-_point);

_word2 := Copy(_word2,1,_point)+'-'+RIGHT(_word2,Length(_word2)-_point);

end

else

begin

// VC-CV

_word := Copy(_word ,1,_point+1)+'-'+RIGHT(_word ,Length(_word )-( _point+1));

_word2 := Copy(_word2,1,_point+1)+'-'+RIGHT(_word2,Length(_word2)-( _point+1));

end;

_point := Pos('VCCV',_word2);

end;

_point := Pos('VCCCV',_word2);

while _point <> 0 do

begin

C1 := UPPERCase(Copy(_word,_point+2,1));

C2 := UPPERCase(Copy(_word,_point+3,1));

if ( (NOT Contem(C1,'JLHMNQRSXZ') ) and Contem(C2,'LR') ) or ( Contem(C1,'CLNPST') and (C2='H')) then

begin

```

```

_word := Copy(_word ,1,_point+1)+''+RIGHT(_word ,Length(_word )-( _point+1));

_word2 := Copy(_word2,1,_point+1)+''+RIGHT(_word2,Length(_word2)-( _point+1));

end

else

begin

_word := Copy(_word ,1,_point+2)+''+RIGHT(_word ,Length(_word )-( _point+2));

_word2 := Copy(_word2,1,_point+2)+''+RIGHT(_word2,Length(_word2)-( _point+2));

end;

_point := Pos('VCCCV',_word2);

end;

_point := Pos('VCCCCV',_word2);

while _point <> 0 do

begin

_word := Copy(_word ,1,_point+2)+''+RIGHT(_word ,Length(_word )-( _point+2));

_word2 := Copy(_word2,1,_point+2)+''+RIGHT(_word2,Length(_word2)-( _point+2));

_point := Pos('VCCCCV',_word2);

end;

_point := Pos('VV',_word2);

while _point <> 0 do

begin

if not Pos(''+UPPERCase(Copy(_word,_point
,2))+','!'AI!AO!ÃO!ãO!AU!E!EU!OE!ÕE!øE!OI!OO!OU!' +iif(_point<>1,iif(Pos(UPPERCase(Copy(_word,_point-
1,1)), 'GQ')>0,'UA!ÛA!üA!UE!ÛE!üE!U!Ü!ü!UO!ÛO!üO!UU!ÛU!üU!',"))>0 then

begin

_word := Copy(_word ,1,_point)+''+RIGHT(_word ,Length(_word )-_point);

_word2 := Copy(_word2,1,_point)+''+RIGHT(_word2,Length(_word2)-_point);

```

end;

if Pos('VV',RIGHT(_word2,Length(_word2)-_point)) <> 0 then

 _point := _point + Pos('VV',RIGHT(_word2,Length(_word2)-_point))

else _point := 0;

end;

result := (_word);

end;

end.



Celso Augusto Nunes da Conceição

Graduação em Letras pela Faculdade Porto-Alegrense de Educação, Ciências e Letras (1988), Mestrado em Linguística pela Pontifícia Universidade Católica do Rio Grande do Sul (1997) e Doutorado (2009) na área de Interfaces linguísticas: léxico, morfologia, linguística computacional e inteligência artificial; professor e pesquisador no Centro Universitário La Salle – UNILASALLE, ministrando as disciplinas de Linguística Textual, Semântica, Pragmática, Psicolinguística e Português Instrumental, de base lógica, e com pesquisas nas áreas do conexionismo, da psicolinguística, da semântica, da pragmática, da linguística computacional e da inteligência artificial; professor na Fundação Escola Superior da Defensoria Pública do Rio Grande do Sul, ministrando a disciplina de Língua Portuguesa, com abordagens morfossintático-semântico-pragmáticas e lógica para a linguagem jurídica.

(Texto informado pelo autor)

Última atualização em 11/06/2009


Endereço para acessar este CV:

<http://lattes.cnpq.br/1124430988788748>

Dados Pessoais

Nome	Celso Augusto Nunes da Conceição
Filiação	Celso Siedler da Conceição e Noely Nunes da Conceição
Nascimento	07/01/1955 - Porto Alegre/RS - Brasil
Carteira de Identidade	801088242 SJS - RS - 17/12/1998
CPF	18365205068
Endereço residencial	Rua São Luiz, 146 Centro - Canoas 92310-120, RS - Brasil Telefone: 51 99664419
Endereço profissional	Centro Universitário La Salle, Departamento de Letras, Departamento de Letras Av. Victor Barreto, 2288 -Centro Centro - Canoas 92010-000, RS - Brasil Telefone: 051 34768500
Endereço eletrônico	e-mail para contato : celsus@terra.com.br e-mail alternativo : celsus@unilasalle.edu.br

Formação Acadêmica/Titulação

2004 - 2009	Doutorado em Linguística e Letras. Pontifícia Universidade Católica do Rio Grande do Sul, PUC RS, Porto Alegre, Brasil Título: Morfologia Computacional: um algoritmo ilustrativo sobre o plural no Português Brasileiro, Ano de obtenção: 2009 Orientador: Jorge Campos da Costa Bolsista do(a): Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
1993 - 1997	Mestrado em Linguística e Letras. Pontifícia Universidade Católica do Rio Grande do Sul, PUC RS, Porto Alegre, Brasil Título: Criação de um Programa Computacional de Banco de Dados para a Implementação do Heureka - O Dicionário Remissivo, Ano de obtenção: 1997 Orientador: Prof Dr. José Marcelino Poersch 
1985 - 1988	Graduação em Letras. Faculdade Porto-Alegrense de Educação, Ciências e Letras, FAPA, Brasil

Formação complementar

1999 - 1999	Curso de curta duração em Semântica Formal. Centro Universitário Franciscano de Santa Maria, UNIFRAN, Brasil
--------------------	---

- 2000 - 2000** Curso de curta duração em Semântica Discursiva.
Pontifícia Universidade Católica do Rio Grande do Sul, PUC RS, Porto Alegre, Brasil
- 2002 - 2002** Curso de curta duração em Linguagem, Cognição e Informática.
Universidade do Vale do Rio dos Sinos, UNISINOS, Sao Leopoldo, Brasil
- 2006 - 2006** Extensão universitária em LIBRAS.
Centro Universitário La Salle, UNILASALLE, Canoas, Brasil
- 2007 - 2007** Curso de curta duração em Metodologia de Educação Superior: Expressão Oral e.
Centro Universitário La Salle, UNILASALLE, Canoas, Brasil
- 2006 - 2007** Semântica Lógica.
Pontifícia Universidade Católica do Rio Grande do Sul, PUC RS, Porto Alegre, Brasil
- 2007 - 2007** Curso de curta duração em Metodologia de Educação Superior.
Centro Universitário La Salle, UNILASALLE, Canoas, Brasil
- 2008 - 2008** Semântica das interfaces lógico-lingüísticas.
Pontifícia Universidade Católica do Rio Grande do Sul, PUC RS, Porto Alegre, Brasil
- 2008** Ensino à Distância.
Centro Universitário La Salle, UNILASALLE, Canoas, Brasil

Atuação profissional

1. Centro Universitário La Salle - UNILASALLE

Vínculo institucional

- 2008 - Atual** Vínculo: Professor horista , Enquadramento funcional: Professor horista , Carga horária: 20, Regime: Parcial
- 2003 - 2008** Vínculo: Celetista , Enquadramento funcional: Professor Assistente , Carga horária: 40, Regime: Integral
- 1999 - 2003** Vínculo: Celetista , Enquadramento funcional: Professor horista , Carga horária: 16, Regime: Parcial

Atividades

- 03/2008 - Atual** Graduação, Letras
- Disciplinas Ministradas:*
Língua Portuguesa VI - Semântica, Pragmática e Teoria do Discurso , Lingüística I
- 03/2008 - Atual** Extensão Universitária, Departamento de Extensão
- Especificação:*
Curso de Competência Lingüística do Programa de Nivelamento
- 08/2006 - 03/2009** Projetos de pesquisa, Departamento de Pesquisa
- Participação em projetos:*
Uma simulação computacional conexcionista da aquisição do plural
- 08/2006 - 10/2006** Especialização
- Especificação:*
Estudos de Lingüística Textual
- 03/2006 - 07/2006** Extensão Universitária, Departamento de Extensão
- Especificação:*
Lingüística para o Direito - Módulo I: noções básicas
- 05/2004 - 09/2004** Especialização
- Especificação:*
Estudos Lingüísticos do Texto
- 03/2004 - 02/2008** Serviço Técnico Especializado, Consultoria Técnica
- Especificação:*
Revisor de texto
- 03/2004 - 12/2005** Ensino de Graduação, Letras, , Departamento de Letras, Departamento de Letras
- Disciplinas Ministradas:*
LP II (Lingüística Textual) , LP VII (Semântica) , LP VIII (Pragmática)
- 03/2004 - 02/2008** Extensão Universitária, Departamento de Extensão

*Especificação:
Revisão de fôlderes*

07/2003 - 12/2003 Especialização

*Especificação:
Semiótica*

03/2003 - 02/2005 Direção e Administração, Departamento de Letras, Departamento de Letras

*Cargos Ocupados:
Coordenador Adjunto*

08/2002 - 08/2002 Direção e Administração, Processo Seletivo de Alunos

*Cargos Ocupados:
Coordenador da Subcomissão de Provas de Vestibular*

01/2001 - 12/2009 Conselhos, Comissões e Consultoria, Conselho Universitário - CONSUN

*Especificação:
Conselheiro*

2. Fundação Escola Superior da Defensoria Pública Rio Grande do Sul - FESDEP

Vínculo institucional

2009 - Atual Vínculo: Colaborador , Enquadramento funcional: Professor horista , Carga horária: 8, Regime: Parcial

Atividades

03/2009 - Atual Aperfeiçoamento

*Especificação:
Língua Portuguesa para o Direito*

3. TGS - Eventos Educacionais - TGS

Vínculo institucional

2009 - Atual Vínculo: Parceria profissional , Enquadramento funcional: Professor , Carga horária: 8, Regime: Parcial

Atividades

07/2009 - Atual Aperfeiçoamento

*Especificação:
Atualização em Língua Portuguesa: Novo Acordo Ortográfico*

4. Faculdade São Judas Tadeu - SAO JUDAS

Vínculo institucional

1999 - 2005 Vínculo: Professor horista , Enquadramento funcional: Professor horista , Carga horária: 8, Regime: Parcial

Atividades

03/1998 - 03/2005 Ensino de Graduação, Administração de Comércio Exterior, , Departamento Ciências Humanas, Departamento Ciências Humanas

*Disciplinas Ministradas:
Português Instrumental para Comércio Exterior*

Projetos

2006 - Atual Uma simulação computacional conexionista da aquisição do plural

Situação: Em Andamento Natureza: Pesquisa
Alunos envolvidos: Doutorado (1);
Integrantes: Celso Augusto Nunes da Conceição (Responsável);
Financiador(es): Centro Universitário La Salle-UNILASALLE
Número de produções C,T & A: 1/

Revisor de periódico

1. Diálogo (Canoas) -

Vínculo

2008 - Atual Regime: Parcial

2. Informativo Unilasalle -

Vínculo

2006 - Atual Regime: Parcial

Membro do corpo editorial

1. Informativo Unilasalle -

Vínculo

2006 - Atual Regime: Parcial

Áreas de atuação

1. Linguística Computacional
2. Semântica
3. Português Instrumental
4. Semiótica
5. Pragmática
6. Lógica

Idiomas

Inglês Compreende Razoavelmente , Fala Pouco, Escreve Razoavelmente, Lê Razoavelmente

Espanhol Compreende Razoavelmente , Fala Razoavelmente, Escreve Razoavelmente, Lê Razoavelmente

Produção em C, T & A

Produção bibliográfica

Artigos completos publicados em periódicos

1. CONCEIÇÃO, C. A. N., MOREIRA, E. A. S.
Tecnologia, Linguística e Escola: um processo de integração. Diálogo (Canoas). , v.1, p.139 - 156, 2005.

Capítulos de livros publicados

1. ★ CONCEIÇÃO, C. A. N.
Criação de um programa computacional de banco de dados para a implementação do projeto Heureka, o dicionário remissivo In: Rumo à Psicolinguística Conexionista.1 ed.Porto Alegre : Edipucrs, 2004, v.1, p. 1-321.

Trabalhos publicados em anais de eventos (resumo)

1. CONCEIÇÃO, C. A. N.
Uma simulação computacional conexionista da aquisição do plural In: 8º Congresso Internacional da ISAPL, 2007, Porto Alegre.
8º Congresso Internacional da ISAPL. Porto Alegre - RS: Edipucrs, 2007.
2. CONCEIÇÃO, C. A. N.
Os conetivos logo e porque em uma relação proposicional inversa de causa e consequência para justificar o argumento In: Atividades Integradas de Extensão, 2006, Canoas.
Atividades Integradas de Extensão. , 2006.
3. CONCEIÇÃO, C. A. N.
Pôster: A Inteligência Artificial na morfologia: uma abordagem conexionista computacional para a aquisição do plural In: IX SEFIC e Mostra de Trabalhos, 2006, Canoas.
IX SEFIC e Mostra de Trabalhos. , 2006.
4. CONCEIÇÃO, C. A. N.
Os conetivos logo e porque em uma relação proposicional inversa de causa e consequência para justificar o argumento In: VII Sefic, 2005, Canoas-RS.
Anais do VII Sefic. , 2005.
5. CONCEIÇÃO, C. A. N.
Pôster: Tecnologia, Linguística e Escola: um processo de integração In: VII Sefic, 2005, Canoas-rs.
Anais do VII Sefic. , 2005.
6. ★ CONCEIÇÃO, C. A. N.
How to get an expression by starting from the content In:
. , 1999.
7. ★ CONCEIÇÃO, C. A. N.
How to get an expression by starting from the content In: 12th World Congress of Applied, 1999, Tóquio.
AILA'99 TOKYO. Tokyo: N&M, 1999. v.1. p.326 - 326
8. ★ CONCEIÇÃO, C. A. N.
Electronic Devices for the Eureka In: 16ème Congrès International des Linguistes, 1997, Paris.
CIL 16. Meudon, France: CNRS LLACAN, 1997. v.1. p.335 - 335
9. CONCEIÇÃO, C. A. N.
Quand un dictionnaire donne l'expression en partant du contenu In:
. , 1997.
10. CONCEIÇÃO, C. A. N.
When a dictionaty goes from content to expression In:
. , 1997.

Apresentação de Trabalho

1. CONCEIÇÃO, C. A. N.
Tratamento computacional de conetivos prepositivos para ressignificação proposicional, 2008. (Comunicação,Apresentação de Trabalho)
2. CONCEIÇÃO, C. A. N.
A validade do argumento na teoria dos blocos semânticos e na lógica clássica: uma contribuição para a relação sintática entre as orações causais e explicativas, 2007. (Comunicação,Apresentação de Trabalho)
3. CONCEIÇÃO, C. A. N.
Uma simulação computacional conexionista da aquisição do plural, 2007. (Comunicação,Apresentação de Trabalho)
4. CONCEIÇÃO, C. A. N.
A Adjetivação na construção do sentido no discurso: uma proposta metodológica, 2006. (Comunicação,Apresentação de Trabalho)
5. CONCEIÇÃO, C. A. N.
A aplicabilidade teórica do adjetivo segundo a Teoria do Discurso, 2006. (Comunicação,Apresentação de Trabalho)
6. CONCEIÇÃO, C. A. N.
O aprendizado do plural: uma simulação computacional na visão conexionista, 2006. (Comunicação,Apresentação de Trabalho)
7. CONCEIÇÃO, C. A. N.
Aprendizado da leitura e da escrita através do computador, 1999. (Comunicação,Apresentação de Trabalho)
8. CONCEIÇÃO, C. A. N.
Reflexão e Indagação Estudantil sobre Língua Portuguesa, 1991. (Comunicação,Apresentação de Trabalho)
9. CONCEIÇÃO, C. A. N.
Conteúdos de formação geral no ENADE: o que é isto?, 2008. (Conferência ou palestra,Apresentação de Trabalho)
10. CONCEIÇÃO, C. A. N.
Mídia, Internet e outros fatores de influência no vocabulário, 2007. (Conferência ou palestra,Apresentação de Trabalho)
11. CONCEIÇÃO, C. A. N.
A influência dos conetivos na relação proposicional: uma abordagem lógico-semântica da Gramática à Teoria do Discurso, 2006. (Conferência ou palestra,Apresentação de Trabalho)

12. CONCEIÇÃO, C. A. N.
Letras e as novas tecnologias, 2002. (Conferência ou palestra, Apresentação de Trabalho)
13. CONCEIÇÃO, C. A. N.
Filosofia aplicada ao ensino lingüístico, 2000. (Conferência ou palestra, Apresentação de Trabalho)
14. CONCEIÇÃO, C. A. N.
Uma simulação computacional conexionista da aquisição do plural, 2007. (Congresso, Apresentação de Trabalho)
15. CONCEIÇÃO, C. A. N.
A inteligência artificial na morfologia da Língua Portuguesa: uma simulação computacional conexionista da aquisição do plural, 2006. (Seminário, Apresentação de Trabalho)
16. CONCEIÇÃO, C. A. N.
O Sujeito e Predicado: uma abordagem da Gramática à Teoria do Discurso, 2006. (Seminário, Apresentação de Trabalho)
17. CONCEIÇÃO, C. A. N.
Semântica da Teoria dos Conjuntos na Lógica da Regra do Modus Ponens e do Modus Tollens: o hipônimo e o hiperônimo na base da validade da argumentação, 2007. (Outra, Apresentação de Trabalho)
18. CONCEIÇÃO, C. A. N.
Recursos Computacionais para a Área de Letras, 1999. (Outra, Apresentação de Trabalho)

Produção Técnica

Softwares sem registro ou patente

1. CONCEIÇÃO, C. A. N., SILVA, Everton Johnny da, DAHMER, Alessandra
PluralRNA, 2008

Trabalhos técnicos

1. CONCEIÇÃO, C. A. N.
Núcleo de Apoio Pedagógico, 2008
2. CONCEIÇÃO, C. A. N., Clarice Teresinha Menegat
Tecnólogo em Secretariado Executivo, 2008
3. CONCEIÇÃO, C. A. N.
Núcleo de Apoio Pedagógico, 2007
4. CONCEIÇÃO, C. A. N.
Revisão Textual, 2007
5. CONCEIÇÃO, C. A. N.
Núcleo de Apoio Pedagógico, 2006
6. CONCEIÇÃO, C. A. N.
Revisão Textual, 2006
7. CONCEIÇÃO, C. A. N.
Núcleo de Apoio Pedagógico, 2005
8. CONCEIÇÃO, C. A. N.
Revisão Textual, 2005
9. CONCEIÇÃO, C. A. N.
Núcleo de Apoio Pedagógico, 2004
10. CONCEIÇÃO, C. A. N.
Revisão Textual, 2004

Demais produções técnicas

1. CONCEIÇÃO, C. A. N.
Competência Textual do Programa de Nivelamento Institucional, 2008. (Extensão, Curso de curta duração ministrado)
2. CONCEIÇÃO, C. A. N.
Competência Textual do Programa de Nivelamento Institucional, 2007. (Extensão, Curso de curta duração ministrado)
3. CONCEIÇÃO, C. A. N.
Semantica da Teoria dos Conjuntos na Lógica da Regra do Modus Ponens e do Modus Tollens: o hipônimo e o hiperônimo na base da validade da argumentação, 2007. (Extensão, Curso de curta duração ministrado)
4. CONCEIÇÃO, C. A. N.
Trabalhos na educação, 2007. (Outra produção técnica)
5. CONCEIÇÃO, C. A. N.
Competência Textual do Programa de Nivelamento Institucional, 2006. (Extensão, Curso de curta duração ministrado)

6. CONCEIÇÃO, C. A. N.
O Sujeito e Predicado: abordagens lógico-lingüística da Gramática à Teoria do Discurso, 2006. (Extensão, Curso de curta duração ministrado)
7. CONCEIÇÃO, C. A. N.
Os conetivos "logo" e "porque" segundo uma abordagem lógico-lingüística, 2006. (Extensão, Curso de curta duração ministrado)
8. CONCEIÇÃO, C. A. N.
Competência Textual do Programa de Nivelamento Institucional, 2005. (Extensão, Curso de curta duração ministrado)
9. CONCEIÇÃO, C. A. N., MOREIRA, Edelvira Aida da Silva
Estratégia de Leitura e Predição (ELP), 2002. (Outro, Curso de curta duração ministrado)
10. CONCEIÇÃO, C. A. N.
O uso de software educacional para melhorar a capacidade de compreensão textual dos alunos das terceiras e quartas séries, 2002. (Outro, Curso de curta duração ministrado)
11. CONCEIÇÃO, C. A. N.
A compreensão textual: uma abordagem semântico-funcional para provocar e entender ambigüidades lexicais e estruturais, 2001. (Extensão, Curso de curta duração ministrado)
12. ★ CONCEIÇÃO, C. A. N.
Não Tropece nas Letras, 2001. (Outra, Programa de Rádio ou TV)
13. CONCEIÇÃO, C. A. N.
Implementação computacional para Heureka, o Dicionário Remissivo, 1998. (Relatório de pesquisa)
14. CONCEIÇÃO, C. A. N.
II Encontro do Círculo de Estudos Lingüísticos do Sul - CELSUL, 1997. (Anais, Editoração)
15. CONCEIÇÃO, C. A. N.
Ambigüidade: A causa de interpretações diversas no uso discriminado do gerúndio, 1990. (Relatório de pesquisa)

Produção artística/cultural

1. CONCEIÇÃO, C. A. N.
Não Tropece nas Letras, 2005.
2. CONCEIÇÃO, C. A. N.
Não Tropece nas Letras, 2004.
3. CONCEIÇÃO, C. A. N.
Não Tropece nas Letras, 2003.
4. CONCEIÇÃO, C. A. N.
Não Tropece nas Letras, 2002.
5. CONCEIÇÃO, C. A. N.
Não Tropece nas Letras, 2001.
6. CONCEIÇÃO, C. A. N.
Não Tropece nas Letras, 2000.

Orientações e Supervisões

Orientações e Supervisões concluídas

Monografias de conclusão de curso de aperfeiçoamento/especialização

1. Lilian Patrícia da Silva. **Ambigüidade nos textos jornalísticos**. 2008. Monografia (Leitura e Produção Textual) - Centro Universitário La Salle
2. Joseane Rosa Rockenbach. **Análise e aplicação dos articuladores na produção textual de alunos de 8ª série do ensino fundamental**. 2008. Monografia (Leitura e Produção Textual) - Centro Universitário La Salle
3. Inaciane Teixeira da Silva. **Análise de elementos que contribuem paa a constituição do humor nas tiras em quadrinhos de Hagar o horrível**. 2006. Monografia (Leitura e Produção Textual) - Centro Universitário La Salle
4. Milena Machado Ferreira. **Coesão seqüencial: função do conetivo "e" nos textos elaborados por ciranças na 5a. série do ensino fundamental**. 2006. Monografia (Leitura e Produção Textual) - Centro Universitário La Salle
5. Lorena Brandli Zambon. **O texto no nível da microestrutura (coesão)**. 2006. Monografia (Leitura e Produção Textual) - Centro Universitário La Salle
6. Maria Cristina Pfeifer Fonseca. **A ESTRUTURA DO PORTUGUÊS FALADO NOS TEXTOS COMUNICATIVOS**. 2005. Monografia (Leitura e Produção Textual) - Centro Universitário La Salle

Trabalhos de conclusão de curso de graduação

1. Everton Johnny da Silva. **UM ESTUDO DE ALGORITMOS DE APRENDIZAGEM APLICADOS À MORFOLOGIA DA LÍNGUA PORTUGUESA**. 2008. Curso (Ciência da Computação) - Centro Universitário La Salle
2. Cibeli dos Santos Fernandes. **A influência da elaboração das questões no desempenho lingüístico do estudante**. 2007. Curso (Letras) - Centro Universitário La Salle
3. Carla Rosana Gonçalves. **As relações imperfeitas da sinonímia**. 2003. Curso (Letras) - Centro Universitário La Salle
4. Isabel Cristina Ribeiro. **Por que os alunos têm dificuldades de escrever?**. 2003. Curso (Letras) - Centro Universitário La Salle
5. Everton de Fraga Falcão. **A ambigüidade estrutural: o uso inadvertido ou intencional do pronome relativo**. 2002. Curso (Letras) - Centro Universitário La Salle

Demais Trabalhos

1. CONCEIÇÃO, C. A. N. **Formatação e geração de CD (em formato Acrobat) do II Encontro do Círculo de Estudos Lingüísticos do Sul - CELSUL**, 1999.

Eventos

Participação em eventos

1. Apresentação Oral no(a) **Centro de Estudos Lingüísticos do Sul - Celsul**, 2008. (Encontro)
Tratamento computacional dos conectivos prepositivos para ressignificação proposicional.
2. Apresentação Oral no(a) **III Encontro Estadual de Educação Básica - EnEB**, 2007. (Encontro)
A validade do argumento na teoria dos blocos semânticos e na lógica clássica: uma contribuição para a relação sintática entre as orações causais e explicativas.
3. Apresentação Oral no(a) **III Encontro Estadual de Educação Básica - EnEB**, 2007. (Encontro)
Múltiplas linguagens na 12a sessão de comunicações.
4. Homenageado no(a) **Formatura do Curso de Letras**, 2007. (Outra)
Paraninfo.
5. Apresentação Oral no(a) **IV Atividades Integradas de Extensão**, 2007. (Encontro)
Semântica da Teoria dos Conjuntos na Lógica da Regra do Modus Ponens e do Modus Tollens: o hipônimo e o hiperônimo na base da validade da argumentação.
6. Apresentação Oral no(a) **8th International Congress of ISAPL**, 2007. (Congresso)
Uma simulação computacional conexionista da aquisição do plural.
7. Apresentação Oral no(a) **VII Celsul - Centro de Estudos Lingüísticos do Texto**, 2006. (Encontro)
A aplicabilidade teórica do adjetivo segundo a Teoria do Discurso.
8. Conferencista no(a) **Semana Acadêmica de Letras - URI**, 2006. (Outra)
A influência dos conectivos na relação proposicional: uma abordagem lógico-semântica da Gramática à Teoria do Discurso.
9. Apresentação de Poster / Painel no(a) **IX SEFIC e Mostra de Trabalhos**, 2006. (Outra)
A inteligência artificial na morfologia da Língua Portuguesa: uma simulação computacional conexionista da aquisição do plural.
10. Apresentação de Poster / Painel no(a) **II Semana Científica do Unilasalle**, 2006. (Encontro)
A Inteligência Artificial na Morfologia da Língua Portuguesa: uma simulação computacional conexionista da aquisição do plural.
11. Apresentação Oral no(a) **Seminário de Letras**, 2006. (Seminário)
A inteligência artificial na morfologia da Língua Portuguesa: uma simulação computacional conexionista da aquisição do plural.
12. Homenageado no(a) **Formatura do Curso de Letras**, 2006. (Outra)
Paraninfo.
13. **LIBRAS**, 2006. (Oficina)
14. Homenageado no(a) **Formatura do Curso de Letras**, 2005. (Outra)
Paraninfo.
15. Apresentação Oral no(a) **XI EGEL - Encontro Gaúcho dos Estudantes de Letras**, 2004. (Encontro)
Os conectores lógicos logo e porque na relação proposicional inversa de causa e consequência em uma abordagem lógico-lingüística para fundamentar o argumento.
16. Homenageado no(a) **Formatura do Curso de Letras**, 2004. (Outra)
Paraninfo.
17. Homenageado no(a) **Formatura do Curso de Letras**, 2003. (Outra)
Paraninfo.
18. Homenageado no(a) **Formatura do Curso de Letras**, 2002. (Outra)
Paraninfo.

19. Apresentação Oral no(a) **IV SEFIC-Salão e Feira de Iniciação Científica**, 2001. (Outra)
A compreensão textual: uma abordagem semântico-funcional para provocar e entender as ambigüidades lexicais e estruturais.
20. Homenageado no(a) **Formatura do Curso de Letras**, 2001. (Outra)
Paraninfo.
21. Conferencista no(a) **PENSARE, Salão de Vivências Pedagógicas no Ensino de Filosofia**, 2000. (Encontro)
Filosofia aplicada ao ensino lingüístico.
22. Homenageado no(a) **Formatura do Curso de Letras**, 2000. (Outra)
Homenageado pela Habilitação de Língua Portuguesa.
23. Apresentação Oral no(a) **III Encontro Bianual do CELSUL**, 1999. (Encontro)
Ensino infantil com recursos computacionais.
24. Apresentação Oral no(a) **12th World Congress of Applied Linguistics da AILA'99**, 1999. (Congresso)
How to get an expression by starting from the content.
25. Apresentação Oral no(a) **16ème Congrès International des Linguistes**, 1997. (Congresso)
When a Dictionary goes from content to expression.

Organização de evento

1. CONCEIÇÃO, C. A. N., Rosa, Lúcia R L, Clarice Teresinha Menegat
III Seminário de Letras, 2008. (Outro, Organização de evento)
2. CONCEIÇÃO, C. A. N.
Avaliação de redações no Vestibular Unilasalle 2005-1, 2005. (Concurso, Organização de evento)
3. CONCEIÇÃO, C. A. N.
Avaliação de redações no Vestibular Unilasalle 2004-1, 2004. (Concurso, Organização de evento)
4. CONCEIÇÃO, C. A. N.
Avaliação de redações no Vestibular Unilasalle 2004-2, 2004. (Concurso, Organização de evento)
5. CONCEIÇÃO, C. A. N.
Avaliação de redações no Vestibular Unilasalle 2003-1, 2003. (Concurso, Organização de evento)
6. CONCEIÇÃO, C. A. N.
Avaliação de redações no Vestibular Unilasalle 2003-2, 2003. (Concurso, Organização de evento)

Bancas

Participação em banca de comissões julgadoras

Professor titular

1. **Seleção de professores do Unilasalle**, 2007
Centro Universitário La Salle
2. **Trabalhos de Iniciação Científica**, 2007
Centro Universitário La Salle
3. **Seleção de professores do Unilasalle**, 2006
Centro Universitário La Salle
4. **Seleção de professores do Unilasalle**, 2005
5. **Seleção de professores do Unilasalle**, 2004
Centro Universitário La Salle

Outra

1. **Avaliador de Redações no Vestibular do Unilasalle**, 2007
Centro Universitário La Salle
2. **II Seminário de Letras - Linguagens e Práticas Educativas**, 2007
Centro Universitário La Salle
3. **Avaliação de Redações no Vestibular do Unilasalle**, 2006
4. **Trabalhos de Iniciação Científica**, 2005
5. **Trabalhos de Iniciação Científica**, 2003
Centro Universitário La Salle
6. **Trabalhos de Iniciação Científica**, 2001
Centro Universitário La Salle

7. Trabalhos de Iniciação Científica, 2000
Centro Universitário La Salle

Totais de produção

Produção bibliográfica

Artigos completos publicado em periódico	1
Capítulos de livros publicados	1
Trabalhos publicados em anais de eventos	10
Apresentações de Trabalhos (Comunicação)	8
Apresentações de Trabalhos (Conferência ou palestra)	5
Apresentações de Trabalhos (Congresso)	1
Apresentações de Trabalhos (Seminário)	2
Apresentações de Trabalhos (Outra)	2

Produção Técnica

Softwares (computacional)	1
Trabalhos técnicos (assessoria)	4
Trabalhos técnicos (consultoria)	5
Trabalhos técnicos (elaboração de projeto)	1
Curso de curta duração ministrado (extensão)	8
Curso de curta duração ministrado (outro)	2
Editoração (anais)	1
Programa de Rádio ou TV (outra)	1
Relatório de pesquisa	2
Outra produção técnica	1

Orientações

Orientação concluída (monografia de conclusão de curso de aperfeiçoamento/especialização)	6
Orientação concluída (trabalho de conclusão de curso de graduação)	5

Eventos

Participações em eventos (congresso)	3
Participações em eventos (seminário)	1
Participações em eventos (oficina)	1
Participações em eventos (encontro)	9
Participações em eventos (outra)	11
Organização de evento (concurso)	5
Organização de evento (outro)	1
Participação em banca de comissões julgadoras (professor titular)	5
Participação em banca de comissões julgadoras (outra)	7

Produção cultural

Programa de rádio ou TV (outro)	6
---------------------------------	---

Demais trabalhos relevantes

Demais trabalhos relevantes	1
-----------------------------	---

Citações em bases bibliográficas

Web of Science Número total de citações : 10;Número de trabalhos : Data : 24/11/2006
Nome(s) do autor utilizado(s) na consulta para obter o total de citações:
100

Página gerada pelo Sistema Currículo Lattes em 29/09/2009 às 13:39:17.