

**LINGÜISTA “PURO” VS. LINGÜISTA “COMPUTACIONAL”:
REVISITANDO A DISTINÇÃO ENTRE “LINGÜISTA DE POLTRONA” E
“LINGÜISTA APLICADO”**

Gabriel de Ávila Othero*

RESUMO: revisitamos a antiga questão do “lingüista de poltrona” vs. “lingüista de campo”, trazendo a discussão para a Lingüística Computacional, mais especificamente, para a área de processamento sintático, ou *parsing*. Discutiremos algumas idéias centrais da área de processamento mostrando alguns conflitos entre os objetivos dos trabalhos que visam ao entendimento da competência gramatical dos falantes e os objetivos dos trabalhos que visam ao estudo do desempenho lingüístico. Palavras-chave: processamento computacional; Lingüística Computacional; epistemologia da Lingüística.

ABSTRACT: we overview the old and well known conflict between the “armchair linguist” vs. the “field linguist”. Our approach uses examples from Computational Linguistics, specially focused on automatic parsing. We bring some key ideas involved in syntactic parsing and show some of the conflicts between studies that focus on competence and studies that are focused on performance.

Keywords: grammatical parsing; Computational Linguistics; Linguistics Epistemology.

1. Introdução

Neste artigo, iremos visitar a velha questão do “lingüista de poltrona”¹ vs. “lingüista de campo”, trazendo a discussão para a Lingüística Computacional. Sendo uma área de interface por natureza, a Lingüística Computacional envolve, pelo menos, pesquisadores destas duas áreas: Informática e Lingüística. Traremos a discussão do lingüista “puro” vs. lingüista “aplicado” para um campo que nos é conhecido e temos alguma experiência, o do processamento sintático computacional, ou *parsing*.

* Doutorando em Lingüística Aplicada pela Pontifícia Universidade Católica do Rio Grande do Sul – PUCRS. Texto escrito em março de 2008.

¹ Termo popularizado por Fillmore, 1991.

2. Lingüística Computacional e *parsing*

A Lingüística Computacional é a área da ciência Lingüística preocupada com o tratamento computacional da linguagem e das línguas naturais. Ela envolve diversos campos de estudo da Lingüística Teórica e Aplicada, e seus primeiros desenvolvimentos se deram nos anos 1950. A área teve grande impulso principalmente graças a esforços para o desenvolvimento de programas de tradução automática nas décadas de 1950 e 1960, e ela está intrinsecamente ligada a desenvolvimentos na área da Inteligência Artificial².

De acordo com Grisham, 1992: 1,

o potencial para o processamento de linguagem natural foi reconhecido bem cedo no desenvolvimento de computadores, e trabalhos em Lingüística Computacional – basicamente para tradução automática – começaram na década de 1950 em diversos centros de pesquisa. O rápido crescimento na área, no entanto, aconteceu principalmente a partir do final dos anos 1970.³

Hoje, a busca por agentes inteligentes, a procura por interfaces mais amigáveis e por *software* voltados para o trabalho com a linguagem natural têm motivado muitas pesquisas na área, e diversos são os frutos que envolvem áreas como a Inteligência Artificial e a Lingüística Computacional. Para McDonald & Yazdani, 1990: 176, “a pesquisa em PLN [Processamento de Linguagem Natural] pode proporcionar *insights* bastante úteis sobre processos e representações da linguagem na mente humana, apontando, assim, para a verdadeira IA”⁴.

A Sintaxe tem se mostrado uma área da Lingüística de extremo valor para os desenvolvimentos em Lingüística Computacional. O nível de formalização sintática a que chegamos tem dado bons resultados não somente

² Cf. Grisham, 1992, Sparck Jones, 1994, Russel & Norvig, 1995, e Hutchins, 2000, por exemplo.

³ Trecho original: “The potential for natural language processing was recognized quite early in the development of computers, and work in computational linguistics – primarily for machine translation – began in the 1950s at a number of research centers. The rapid growth in the field, however, has taken place mostly since the late 1970s”.

⁴ Trecho original: “Research into NLP may provide useful insights into processes and representation of language in the human mind, thus heading towards true AI”.

para pesquisas de cunho teórico envolvendo o estudo de línguas naturais, mas também para o desenvolvimento de aplicativos computacionais, como *parsers*, *taggers*, corretores ortográficos e gramaticais, classificadores automáticos de documentos digitais, tradutores automáticos, etc.⁵

O processamento sintático, ou *parsing*, diz respeito à interpretação automática (ou semi-automática) de sentenças de linguagem natural por meio de programas de computador conhecidos como *parsers*. Esses programas são capazes de classificar morfossintaticamente as palavras e expressões de sentenças em uma dada língua e atribuir às sentenças a sua estrutura de constituintes, baseando-se em um modelo formal de gramática.

De acordo com Covington, 1994: 42, fazer o *parsing* de uma sentença é “determinar, por um processamento algorítmico, se a sentença é gerada por uma determinada gramática, e se ela for, qual estrutura que a gramática atribui a ela”⁶. Para Bateman, Forrest & Willis, 1997: 166,

um dos principais objetivos da área de PLN nos últimos dez anos tem sido produzir um “analisador gramatical”, ou *parser*, de **abrangência ampla**. Para muitos aplicativos de PLN, o desafio é produzir um *parser* que poderá ser capaz de analisar automática e estruturalmente de maneira correta, de acordo com um esquema de *parsing* definido, qualquer sentença do inglês que possa ocorrer naturalmente, **sem restrições**, de uma gama de gêneros textuais tão vasta quanto possível.⁷ (grifos dos autores).

Outros dez anos já se passaram desde que as linhas acima foram escritas, e diversos *parsers* já foram desenvolvidos ao longo desse tempo. Porém, nenhum deles foi ainda capaz de alcançar o objetivo proposto por Bateman, Forrest & Willis. Por isso, esse ainda continua sendo um dos principais objetivos de PLN, de maneira geral, e do processamento sintático das línguas naturais, de maneira específica.

⁵ Cf. Bick, 1996, e Garside, Leech and McEnery, 1997, por exemplo.

⁶ Trecho original: “(...) to determine, by an algorithmic process, whether the sentence is generated by a particular grammar, and if so, what structure the grammar assigns it”.

⁷ Trecho original: “one of the major aims of NLP over the past ten years has been to produce a **wide-range** ‘grammatical analyser’ or **parser**. For many NLP applications, the challenge is to produce a parser which will automatically be able to structurally analyse correctly, according to a defined parsing scheme, any sentence of naturally occurring **unrestricted** English, from as wide a range of genres as possible”.

Qual é a grande dificuldade em formalizar os conhecimentos lingüísticos em um programa de computador? Sabemos que os estudos formais da linguagem fizeram grandes avanços no século XX. Contudo, não há ainda regras suficientemente formais, explícitas e *gerativas* para possibilitar a formulação de um *parser* que, de certa forma, “compreenda” alguma língua natural. De acordo com Newmeyer, 1980: 2,

houve um sentimento generalizado entre os lingüistas nos anos 1950 de que os problemas fundamentais da análise lingüística haviam sido solucionados e tudo o que restava era preencher alguns detalhes. As afirmações teórico-metodológicas básicas do trabalho de Bloch, “A Set of Postulates for Phonemic Analysis” (1948), e do trabalho de Harris, *Methods in Structural Linguistics* (1951), pareciam tornar qualquer trabalho teórico mais básico algo desnecessário. Na verdade, **muitos lingüistas achavam que os procedimentos tinham sido trabalhados de maneira tão detalhada, que computadores poderiam assumir o trabalho de análise lingüística. Tudo o que se precisava fazer (em teoria) seria colocar os dados no computador e uma gramática iria sair prontinha!**

Também se acreditava que os computadores poderiam resolver outro problema lingüístico tradicional – a tradução. A idéia de uma máquina tradutora foi primeiramente sugerida apenas em 1949 (em um memorando de Warren Weaver). Em 1955, tal trabalho de tradução estava acontecendo em três países, em meia dúzia de instituições. Esses seis anos foram o suficiente para converter os céticos (...)⁸. (Grifos meus)

Uma das decepções dos lingüistas e dos cientistas da computação das décadas de 1950 e 1960 foi justamente esta: ao tentar fazer com que um computador “compreendesse” a linguagem humana, percebemos que ela é de difícil formalização. Um trabalho que parecia simples, como a tradução de

⁸ Trecho original: “There was a widespread feeling among linguists in the 1950s that the fundamental problems of linguistic analysis had been solved and that all that was left was to fill in the details. The basic-theoretical-methodological statements of Bloch’s “A Set of Postulates for Phonemic Analysis” (1948) and Harris’s *Methods in Structural Linguistics* (1951) seemed to render any more basic theoretical work unnecessary. In fact, many linguists felt that the procedures had been so well worked out that computers could take over the drudgery of linguistic analysis. All one would have to do (in principle) would be to punch the data into the computer and out would come the grammar.

“There was also a feeling that computers could solve another traditional linguistic problem – translation. The idea of machine translation had been first suggested (in a memorandum by Warren Weaver) only in 1949. By 1955m such translation work was going on in three countries at half a dozen institutions. These six years were enough to convert the skeptics (...)”.

sentenças de uma língua para outra, mostrou-se um empreendimento complexo, dada a riqueza da linguagem. Este trecho de Baker, 2001: 3, é bem interessante:

Nós geralmente achamos que o xadrez é uma atividade intelectual requintada que pode ser dominada apenas pelos melhores e mais brilhantes. Qualquer pessoa comum, ao contrário, pode falar um horror em um vernáculo entendível sem necessariamente ser considerada inteligente. Ainda assim, mesmo que existam programas de computador que agora podem derrotar os melhores jogadores de xadrez do mundo, não existe nenhum sistema artificial que possa se igualar a um falante médio de cinco anos de idade no que diz respeito a falar e compreender sua língua materna⁹.

Já temos campeões de xadrez que são programas de computador, mas ainda não temos nenhum programa que consiga “compreender” ou “produzir” linguagem tão bem como um ser humano¹⁰. Mas os estudos formais da linguagem têm contribuído para o avanço da Linguística Computacional. E os estudos em sintaxe formal, especialmente de cunho gerativo, já avançaram muito no conhecimento do funcionamento da linguagem humana. Para Newmeyer, 1980: 250, “aprendemos mais sobre a natureza da linguagem nos últimos 25 anos do que nos 2.500 anos anteriores”¹¹. Há, porém, muito ainda para ser feito, e vários são os problemas que lingüistas, informatas e lingüistas computacionais enfrentam para entender o funcionamento da linguagem.

Acreditamos que o ser humano tenha um conhecimento inato do

⁹ Tradução adaptada. Cf. trecho original: “We usually think of chess as a quintessentially intellectual activity that can be mastered only by the best and brightest. Any ordinary person, in contrast, can talk your ear off in understandable English without necessarily being regarded as intelligent for doing so. Yet although computer programs can now beat the best chess players in the world, no artificial system exists that can match an average five-year-old at speaking and understanding English”. Baker está provavelmente se referindo a Deep Blue, uma máquina de jogar xadrez produzida pela IBM, que conseguiu vencer o então campeão mundial de xadrez Gary Kasparov, em 1997.

¹⁰ Há uma espécie de competição em que vários programas de conversação, conhecidos como *chatbots*, tentam se passar por seres humanos. É o Prêmio Loebner, que foi baseado a partir de idéias propostas pelo teste de Turing. Sobre essa competição, cf. Menuzzi & Othero, 2005, além do website www.loebner.net/Prize/loebner-prize.html. Sobre o teste de Turing, cf. Turing, 1950, Searle, 1980, e Hodges, 1983.

¹¹ Trecho original: “(...) more has been learned about the nature of language in the last 25 years than in the previous 2500”. Essa parece ser uma idéia mais ou menos generalizada entre os lingüistas formais. Para Strozer (1994: ix), os estudos lingüísticos “avançaram mais nos últimos dez ou quinze anos do que nos 30 anos anteriores”. Jairo Nunes (em comunicação pessoal) nos disse que frases como essas são comuns em seções introdutórias de textos recentes dos estudos gerativos. Cf. também a esse respeito Harris, 1993, Chomsky, 1988, 1995, 2002, e Thomas, 2004, por exemplo.

funcionamento sintático de sua língua. Há regras que os falantes obedecem ao produzir os enunciados, regras que a imensa maioria dos falantes não saberia explicitar. O trabalho em sintaxe computacional pode ser entendido desta maneira: deve-se explicitar a um computador as regras de funcionamento sintático da língua para que o computador possa então “compreender” a linguagem. Um programa como um *parser* analisa uma sentença em português, por exemplo, dizendo se a sentença é gramatical ou agramatical na língua, atribuindo às sentenças gramaticais a sua correta estrutura sintática (de acordo com a gramática que foi implementada no banco de dados do programa). Para isso, então, é necessário o trabalho do lingüista: é ele quem deverá formalizar as regra sintáticas de uma língua de tal forma que elas possam ser implementadas computacionalmente¹². O trabalho do lingüista parece, então, anteceder o trabalho do informata.

Acontece que há aí um problema metodológico. Na prática, encontramos um problema metodológico entre o trabalho do lingüista e o do lingüista computacional. Enquanto é praxe na Lingüística que os sintaticistas se debrucem sobre o estudo da **competência** dos falantes, os lingüistas computacionais que desejam desenvolver programas robustos que analisem uma língua natural devem estar preparados para lidar com sentenças efetivamente produzidas na língua. Ou seja, eles devem se preocupar também com o **desempenho** dos falantes.

Na verdade, esse embate entre o estudo da competência e o estudo do desempenho não é recente e nem surgiu com a Lingüística Computacional. Labov, 1987: 3, já observava o problema:

a posição idealista foi reforçada recentemente por uma distinção entre ‘desempenho’ e ‘competência’. O que é efetivamente dito e usado na comunicação entre as pessoas é entendido como um produto do ‘desempenho lingüístico’, que é regido por muitos outros fatores além da faculdade lingüística, e é profundamente distorcido por diversos tipos de erros dos falantes. O objetivo da Lingüística é chegar à ‘competência’ subjacente do falante, e o estudo do desempenho deve ficar de fora da Lingüística propriamente dita. A

¹² Existem *parsers* que constroem sua gramática a partir de análises em corpora previamente etiquetados. Não é o caso que mencionamos aqui obviamente.

visão materialista é que a ‘competência’ só pode ser entendida através do estudo do ‘desempenho’, e que essa dicotomia envolve um regresso infinito: se existem regras separadas de desempenho para serem analisadas, então elas devem também compreender uma ‘competência’, e então novas regras de ‘desempenho’ para usá-las e assim sucessivamente.¹³

A preocupação dos lingüistas com a descrição formal da sintaxe da competência lingüística dos falantes acaba acarretando dois problemas para o informata e para o lingüista computacional: (i) os princípios e regras formais da sintaxe de uma língua, tais como descritas por sintaticistas teóricos, dificilmente podem ser aplicados fidedignamente em implementações computacionais; e, do outro lado da moeda, (ii) as gramáticas e os programas de *parsing* desenvolvidos para efetuar o *parsing* robusto das línguas nem sempre são vistos com bons olhos por lingüistas “puros”, dado seu caráter marcadamente *ad hoc*¹⁴.

Esses dois pontos são, evidentemente, uma generalização de casos. Existem alguns formalismos sintáticos que se esforçam para satisfazer os dois lados desse cabo de guerra. Há teorias e formalismos sintáticos que pretendem ser fiéis à teoria lingüística (mantendo sua elegância teórica) e fiéis à sua implementação computacional (pretendendo ser facilmente implementáveis e prometendo apresentar bons resultados em sistemas robustos de *parsing*)¹⁵.

¹³ Trecho original: “The idealist position has more recently been reinforced by a distinction between ‘performance’ and ‘competence.’ What is actually said and communicated between people is said to be the product of ‘language performance’, which is governed by many other factors besides the linguistic faculty, and is profoundly distorted by speaker errors of various kinds. The goal of linguistics is to get at the underlying ‘competence’ of the speaker, and the study of performance is said to lie outside of linguistics proper. The materialist view is that ‘competence’ can only be understood through the study of ‘performance’, and that this dichotomy involves an infinite regress: if there are separate rules of performance to be analyzed, then they must also comprise a ‘competence’, and then new rules of ‘performance’ to use them, and so on”. Ver também Searle, 1972, Lakoff, 1973, Newmeyer, 1983, e Fillmore, 1991.

¹⁴ Um exemplo disso pode ser visto na gramática do *parser* do português Grammar Play, que apresenta algumas regras sintáticas de funcionamento reconhecidamente *ad hoc* (cf. Othero, 2006, e Alencar, 2008), mesmo não sendo o Grammar Play um *parser* robusto. Outro exemplo pode ser visto na gramática do *parser* Palavras, um *parser* robusto do português, que apresenta uma série de limitações e regras gramaticais por vezes “inusitadas” sob o ponto de vista lingüístico (cf. Bick, 2000). Para uma boa discussão sobre o assunto, cf. Martins & Othero (no prelo).

¹⁵ Como a *Head-Driven Phrase Structure Grammar* (HPSG – Pollard & Sag, 1994; Sag, Wasow & Bender, 2003) e as *Tree-Adjoining Grammars* (TAGs – Joshi & Schabes, 1996;

Acontece que nem sempre o objetivo do lingüista formal ou do sintaticista é o mesmo que o do lingüista computacional ou do informata. Os primeiros costumam se preocupar com o conhecimento dos falantes, sua faculdade da linguagem, sua competência lingüística, sua língua-I. Suas explicações têm um ônus com o processamento mental das regras sintáticas de uma língua (ou das línguas humanas), com a aquisição da linguagem e, muitas vezes, com a cognição humana. A pergunta que costuma dirigir seus estudos é “o que sabemos quando sabemos uma língua?”¹⁶.

Para os informatas e lingüistas computacionais, isso nem sempre é importante. O estudo sintático computacional de uma língua não precisa se preocupar com o sistema cognitivo de seus falantes, nem tampouco com a aquisição da linguagem pelas crianças. Deve, antes, se preocupar com a **produção** dos falantes de uma língua. Um *parser* robusto deve poder analisar textos reais e espontâneos, deve saber analisar as sentenças tais como se apresentam nos *corpora* de uma determinada língua, deve poder responder ao desafio proposto por Bateman, Forrest & Willis, 1997, que vimos anteriormente.

Dessa dicotomia, nascem às vezes alguns desentendimentos teóricos ou metateóricos entre lingüistas puros e lingüistas computacionais. Afinal, nossa fala é repleta de frases truncadas, repetições e construções ambíguas ou inusitadas. Parece simples construir uma gramática que analise uma sentença como (1) e efetuar seu *parsing*:

(1) O João adora chocolates.

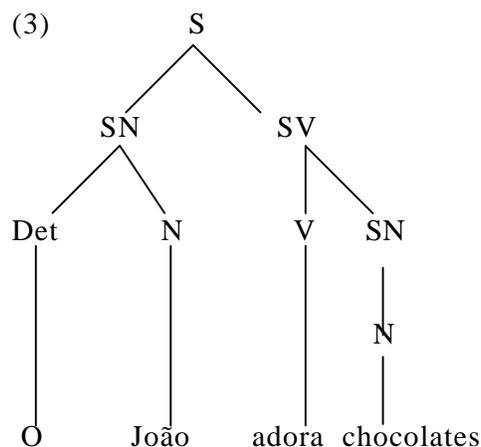
Para isso, poderíamos pensar em regras como as de (2), abaixo:

(2) $S \rightarrow SN SV$
 $SN \rightarrow (Det) N$
 $SV \rightarrow V SN$

Prolo, 2008).

¹⁶ Cf. Chomsky, 1957, 2002, Raposo, 1992, Culicover, 1997, Jackendoff, 2002, entre outros.

Com base nessas regras, podemos analisar a sentença (1) e representá-la em uma estrutura arbórea:



Onde temos os símbolos:

S = sentença

SN = sintagma nominal

Det = determinante

N = nome (ou substantivo)

SV = sintagma verbal

V = verbo

Mas as análises das sentenças de uma língua em uso nem sempre são tão simples, como podemos ver com a sentença (4), que parece ter uma estrutura sintagmática um tanto mais complexa:

(4) De tudo ao meu amor serei atento \ antes, e com tal zelo, e tanto \ que mesmo em face do maior encanto \ dele se encante mais meu pensamento¹⁷.

¹⁷ Moraes, 1991[1951].

Ou como estas amostras de língua falada, que podem representar um desafio ainda maior para um *parser*:

(5) A meu ver... o principal entrave entre o estudo da língua portuguesa nas escolas de primeiro e segundo grau... e os alunos diz basicamente referência ao método como se se trabalha... e também à concepção de língua que se é trabalhada.. a língua portuguesa não é esse fenômeno é:: homogêneo... estático... que é vinculado pela gramática normativa... e pela/infelizmente... pela maioria dos grandes professores de língua portuguesa observamos que a língua evolui... a língua muda... e a escola precisa mudar e evoluir pra trazer o aluno que já é um falante e um usuário da língua portuguesa... a se envolver com o estudo da língua portuguesa¹⁸.

(6) Então a minha [filha] de onze anos... ela supervisiona o trabalho dos cinco... então ela vê se as gavetas estão em orde/... em ordem se o:: material escolar já foi re/arrumado para o dia seguinte... se nenhum:: fez arte demais no banheiro... porque às vezes... estão tomando banho e ficam jogando água pela janela quer dizer essa é supervisora nata é assim... ah... toma conta... precocemente não? das atividades dos irmãos¹⁹.

Programas de *parsing* automático que conseguissem efetuar o *parsing* dos trechos (4), (5) e (6) certamente estariam muito próximos de alcançar o desafio proposto por Bateman, Forrest & Willis, que vimos anteriormente. Por isso, estudos sintáticos cuidadosos têm ainda muito a acrescentar ao lingüista computacional que pretende formalizar regras sintáticas das línguas naturais. E esses estudos podem se basear na competência dos falantes (como nos estudos teóricos ou “puros” de Sintaxe), ou no seu desempenho (como em estudos de *parsing* baseados em *corpora* ou voltados para desenvolvimentos de *parsers*

¹⁸ Amostra de Marcuschi, 2001: 81.

¹⁹ Amostra de Castilho, 2000: 79.

robustos).

3. Palavras Finais

Vimos que a Lingüística Computacional envolve pesquisadores de áreas teóricas – como a Sintaxe Formal – e de áreas aplicadas, práticas – como a Informática. Nos estudos em Lingüística Computacional, parece inegável que os avanços de uma área influenciam nos avanços da outra. Conforme Alencar, manuscrito: 3,

a lingüística computacional não constitui somente uma área de aplicação das pesquisas em gramática gerativa. O inverso também se verifica, com a primeira área constituindo ferramenta importantíssima na construção de modelos dessa última. De fato, a tarefa do lingüista gerativo é muitas vezes descrita como a de descoberta das representações mentais e dos algoritmos executados pelos falantes sobre essas representações na produção ou recepção de frases. É evidente que esse empreendimento será tão mais rigoroso, quanto seja possível testar empiricamente em computadores modelos computacionais da Linguagem construídos pelo lingüista. Talvez não seja exagero dizer que o estágio atual de sofisticação das teorias gerativas decorre, em grande parte, dos progressos na ciência da computação (e, evidentemente, nas disciplinas que informam essa ciência, como é o caso da matemática). Cabe ressaltar que a lingüística gerativa e a ciência da computação nasceram praticamente na mesma época (i.e., meados do século XX) e que Noam Chomsky não é só o fundador dessa teoria lingüística, mas está entre os cientistas que mais contribuíram para a teoria dos compiladores, por meio de estudos na área das línguas formais (a “Hierarquia de Chomsky” e “Forma Normal de Chomsky” são duas noções indispensáveis em qualquer manual de informática teórica).

Acreditamos que o trabalho conjunto do lingüista e do informata seja essencial para que consigamos desenvolver tecnologia que lide com a linguagem (como no caso dos *parsers*). A implementação de teorias lingüísticas consistentes é um desafio constante na Lingüística Computacional e talvez seja isso mesmo o que torne a área bastante especial.

Referências

- ALENCAR, Leonel F. Formalismos gramaticais baseados em estruturas de traços. (Manuscrito não publicado).
- ALENCAR, Leonel F. Resenha de “Teoria X-barras: descrição do português e aplicação computacional”. *ReVEL*, vol. 8, n. 10, março de 2008.
- BAKER, Mark C. *The atoms of language – the mind’s hidden rules of grammar*. New York: Basic Books, 2001.
- BATEMAN, J.; FORREST, J.; WILLIS, T. The use of syntactic annotation tools: partial and full parsing. In: GARSIDE, R.; LEECH, G.; McENERY, A. *Corpus annotation: linguistic information from computer text corpora*.
- BICK, Eckhard. Automatic parsing of Portuguese. In: GARCÍA, L. S. (Ed.). *Anais do II Encontro para o Processamento Computacional de Português Escrito e Falado*. Curitiba: CEFET-PR. 1996.
- BICK, Eckhard. *The parsing system Palavras - automatic grammatical analysis of Portuguese in a constraint grammar framework*. Tese (Doutorado). Aarhus: Aarhus University Press, 2000.
- CASTILHO, Ataliba Teixeira de. *A língua falada no ensino de português*. São Paulo: Contexto, 2000.
- CHOMSKY, Noam. *Language and problems of language – the Managua lectures*. Cambridge: MIT Press, 1988.
- CHOMSKY, Noam. *On nature and language*. Cambridge: Cambridge University Press, 2002.
- CHOMSKY, Noam. *Syntactic structures*. The Hague: Mouton 1957.
- CHOMSKY, Noam. *The minimalist program*. Cambridge: MIT Press, 1995.
- CULICOVER, Peter W. *Principles and parameters – an introduction to*
- FILLMORE, C. J. “Corpus linguistics” or “Computer-aided armchair linguistics”. In: SVARTVIK, Jan (ed.) *Directions in Corpus Linguistics, Proceedings of Nobel Symposium 82*, Stockholm, 4-8 August 1991, Berlin/NY: Mouton de Gruyter, 1992.
- GARSIDE, R.; LEECH, G.; McENERY, A. *Corpus annotation: linguistic information from computer text corpora*. London / New York: Longman, 1997.
- GRISHAM, Ralph. *Computational linguistics: an introduction*. Cambridge: Cambridge University Press, 1992.
- HARRIS, Randy Allen. *The linguistics wars*. New York: Oxford University Press, 1993.
- HODGES, Andrew. *Alan Turing: the enigma*. New York: Simon and Schuster, 1983.
- HUTCHINS, W. J. (ed.). *Early years in machine translation* Amsterdam: John Benjamins, 2000.
- JACKENDOFF, Ray. *Foundations of language: brain, meaning, grammar, evolution*. Oxford: Oxford University Press, 2002.
- JOSHI, Aravind K.; SCHABES, Yves. Tree-Adjoining Grammars. In: A. Salomaa and G. Rozenberg (eds.). *Handbook of Formal Languages*. Springer-Verlag, Berlin, 1996.
- MARCUSCHI, Luiz Antônio. *Da fala para a escrita – atividades de retextualização*. São Paulo: Cortez, 2001.
- MARTINS, Ronaldo Teixeira; OTHERO, Gabriel de Ávila. Parsing do

- português. In: ALENCAR, L. F.; KELLING, C.; OTHERO, G. A. (orgs). *Teoria da gramática: abordagens computacionais*. (no prelo).
- McDONALD, Carlton; YAZDANI, Masoud. *Prolog programming: a tutorial introduction*. Oxford: Blackwell Scientific Publications, 1990.
- MENUZZI, S. M.; OTHERO, G. A. *Linguística computacional: teoria & prática*. São Paulo: Parábola, 2005.
- MORAES, Vinicius de. *Livro de Letras*. São Paulo: Companhia das Letras, 1991.
- NEWMeyer, F. J. *Linguistic theory in America: the first quarter-century of transformational generative grammar*. New York: Academic Press, 1980.
- OTHERO, Gabriel de Ávila. *Teoria X-barras: descrição do português e aplicação computacional*. São Paulo: Contexto, 2006.
- POLLARD, C.; SAG, I. *Head-driven Phrase Structure Grammar*. Chicago: University of Chicago Press, 1994.
- PROLO, Carlos A. Tree-Adjoining Grammars: uma entrevista com Carlos Prolo. *ReVEL*, vol. 6, n. 10, março de 2008.
- RAPOSO, E. P. *Teoria da gramática. A faculdade da linguagem*. Lisboa: Caminho, 1992.
- RUSSEL, S.; NORVIG, P. *Artificial intelligence*. New Jersey: Prentice-Hall, 1995.
- SAG, I.; WASOW, T.; BENDER, E. *Syntactic theory: a formal introduction*. Stanford: CSLI Publications, 2003.
- SEARLE, John. Minds, brains, and programs. *Behavioral and Brain Sciences* 3, 1980.
- SPARCK JONES, Karen. Natural language processing: a historical review. In ZAMPOLI, A.; CALZOLARI, N.; PALMER, M. (eds.). *Current issues in Computational Linguistics: in honour of Don Walker*. Amsterdam: Kluwer, 1994.
- STROZER, J. R. *Language acquisition after puberty*. Washington, DC: Georgetown University Press, 1994.
- syntactic theory*. Oxford: Oxford University Press, 1997.
- THOMAS, Margaret. *Universal Grammar in second language acquisition: a history*. New York: Routledge, 2004.
- TURING, Alan. Computing machinery and intelligence. *Mind*, Vol. 59, No. 236, 1950.