Universidade de São Paulo - USP
Universidade Federal de São Carlos - UFSCar
Universidade Estadual Paulista - UNESP

SENTER: Um Segmentador Sentencial Automático para o Português do Brasil

Thiago Alexandre Salgueiro Pardo

NILC-TR-06-01

Janeiro, 2006

Série de Relatórios do Núcleo Interinstitucional de Lingüística Computacional
NILC - ICMC-USP, Caixa Postal 668, 13560-970 São Carlos, SP, Brasil

Resumo

Apresenta-se, neste relatório, a descrição do SENTER (*SENtence splitTER*), um segmentador sentencial automático para o português do Brasil. Em geral, segmentadores são ferramentas essenciais para a maioria das aplicações de Processamento de Línguas Naturais.

ÍNDICE

1. INTRODUÇÃO **2**
2. SENTER **2**
3. CONSIDERAÇÕES FINAIS **4**
REFERÊNCIAS **4**

1. Introdução

A segmentação textual é o processo de segmentar um texto em unidades menores, que podem ser orações, sentenças, parágrafos e até mesmo tópicos. A granularidade da segmentação depende exclusivamente da aplicação a que se destina a segmentação. Neste relatório, apresenta-se o SENTER[1] (*SENtence splitTER*), um segmentador sentencial automático para textos em português do Brasil.

O processo de segmentação é normalmente executado na fase de pré-processamento de ferramentas de Processamento de Línguas Naturais (PLN). É uma tarefa essencial para que essas ferramentas possam realizar suas funções específicas, por exemplo, etiquetação morfossintática, análise sintática, alinhamento textual, análise retórica, sumarização, tradução automática, etc.

Uma revisão bibliográfica sobre diversas técnicas de segmentação textual e sobre a problemática desta tarefa pode ser encontrada em Pardo e Nunes (2002).

Na próxima seção, descreve-se o SENTER. A seguir, na Seção 3, algumas considerações finais são apresentadas.

2. SENTER

O SENTER é um segmentador sentencial automático para textos em português do Brasil. O sistema foi implementado em C, utilizando-se o ambiente de programação (de uso livre) lcc-32[2] para MS Windows. Por ser implementado em C, o SENTER pode ser portado para qualquer plataforma computacional, bastando que seja devidamente compilado.

O sistema deve ser executado em linha de comando, observando a seguinte sintaxe:

senter.exe meutexto.txt

em que meutexto.txt contém o texto a ser segmentado. Note que o texto deve estar obrigatoriamente em formato *plain text* para que o sistema possa reconhecê-lo. Caso o arquivo com o texto não esteja no mesmo diretório em que o arquivo senter.exe esteja, deve-se especificar o caminho completo para o arquivo com o texto.

Como resultado, será produzido um arquivo de mesmo nome do arquivo com o texto original acrescido da extensão seg. Para o caso anterior, seria produzido o arquivo meutexto.txt.seg. O arquivo produzido contém as sentenças delimitadas, com uma por linha.

Como exemplo, mostram-se, nas Figuras 1-3, um texto a ser segmentado (cujo arquivo chama-se Texto1.txt), a linha de comando para execução do SENTER para esse texto e o texto segmentado (armazenado no arquivo Texto1.txt.seg), respectivamente. Note na Figura 2 que os arquivos do texto e de execução do SENTER encontram-se no mesmo diretório (C:\Temp\).

[1] Disponível para download em http://www.nilc.icmc.usp.br/
[2] Disponível para download em http://www.cs.virginia.edu/~lcc-win32/ (acessado em janeiro de 2006)

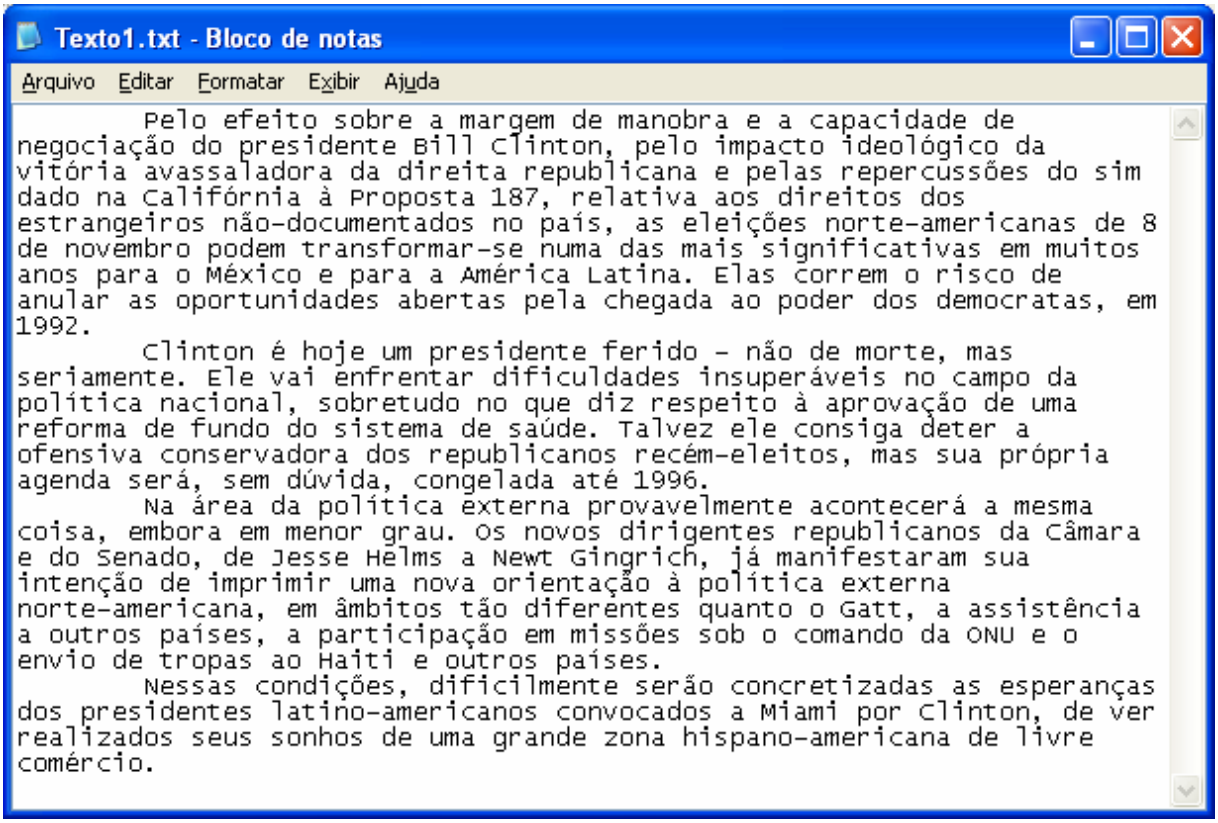

Figura 1 – Texto a ser segmentado

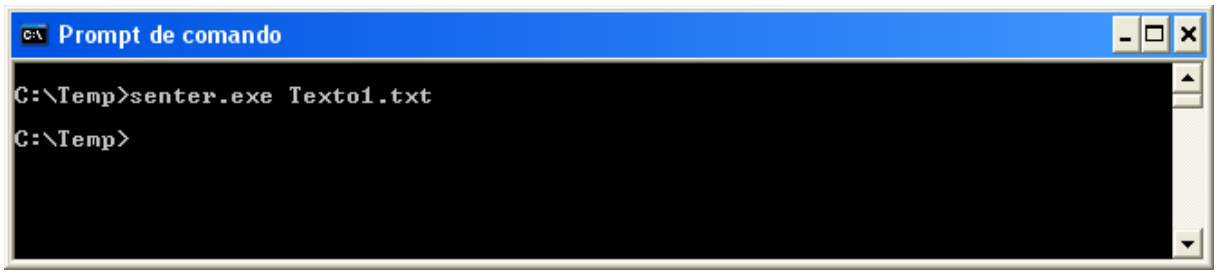

Figura 2 – Execução do SENTER em linha de comando

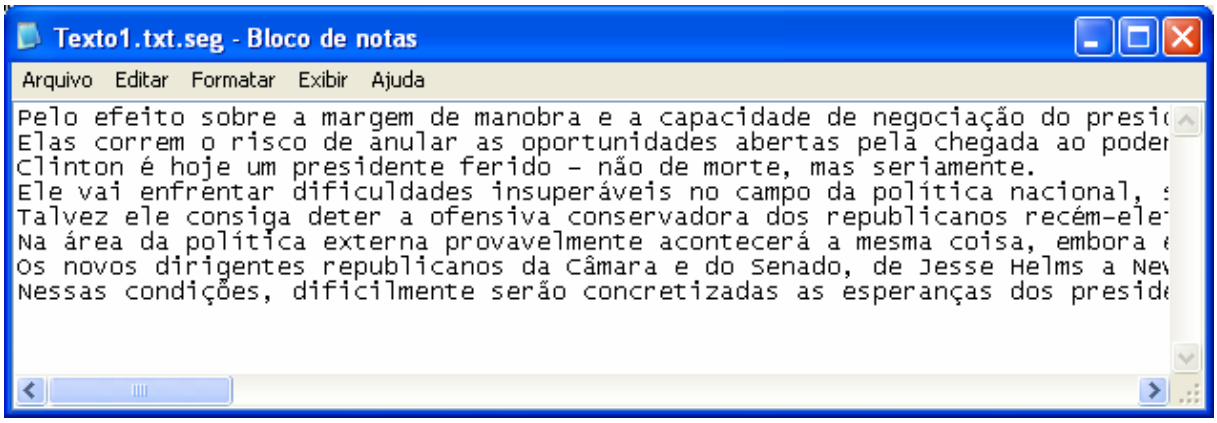

Figura 3 – Texto da Figura 2 segmentado, com uma sentença por linha

O SENTER faz uso de regras para realizar a segmentação sentencial, as quais são:

1. Delimita-se uma sentença sempre que uma marca de nova linha (*carriage return* e *line feed*) é encontrada, independentemente de um sinal de fim de sentença ter sido encontrado anteriormente;
2. Não se delimitam sentenças dentro de aspas, parênteses, chaves e colchetes;
3. Delimita-se uma sentença quando os símbolos de interrogação (?) e exclamação (!) são encontrados;
4. Delimita-se uma sentença quando o símbolo de ponto (.) é encontrado e este não é um ponto de número decimal, não pertence a um símbolo de reticências (...), não faz parte de endereços de e-mail e páginas da Internet e não é o ponto que segue uma abreviatura;
5. Delimita-se uma sentença quando uma letra maiúscula é encontrada após o sinal de reticências ou de fecha-aspas.

Por meio da regra 1, garante-se que a segmentação textual seja realizada mesmo que o escritor esqueça, por exemplo, de fechar aspas ou parênteses ou de delimitar a sentença com um ponto. Em relação à regra 2, o sistema mantém controle sobre o número de aspas, parênteses, chaves e colchetes abertos que devem ser fechados. Na regra 4, para identificar abreviaturas, o sistema faz uso de uma lista de abreviaturas. Essa lista é a lista utilizada pelo revisor gramatical ReGra (Martins et al., 1998) e contém aproximadamente 240 abreviaturas. Ainda na regra 4, a detecção de endereços de e-mail e de páginas da Internet é feita pela verificação da presença dos elementos http, ftp, www e o símbolo @ no contexto anterior ao ponto.

A seguir, algumas considerações finais são apresentadas.

3. Considerações Finais

Apresentou-se, neste relatório, a descrição do SENTER, um segmentador sentencial automático para textos em português do Brasil.

Várias extensões são possíveis de serem realizadas no sistema ou nos resultados produzidos por este. Por exemplo, em adição, pode-se delimitar os parágrafos de um texto, assim como formatar os dados de saída do sistema em XML[3] (*eXtensible Markup Language*), um padrão que tem sido amplamente utilizado.

Referências

Martins, R.T.; Hasegawa, R.; Nunes, M.G.V.; Montilha, G.; Oliveira Jr., O.N. (1998). Linguistic issues in the development of ReGra: a Grammar Checker for Brazilian Portuguese. *Natural Language Engineering*, Vol. 4, N. 4, p287-307. Cambridge University Press.

Pardo, T.A.S. e Nunes, M.G.V. (2002). *Segmentação Textual Automática: Uma Revisão Bibliográfica*. Série de Relatórios Técnicos do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, no. 185. São Carlos-SP, Fevereiro, 21p.

[3] Especificação completa disponível em http://www.w3.org/TR/REC-xml (acessado em janeiro de 2006)