Processamento de Linguagem Natural

Aula 9 – Análise Sintática

*Objetivo: ao fim desta aula, o aluno será capaz de desenhar gramáticas para a construção de árvores sintáticas em linguagem natural para acessar texto semanticamente*

**Nesta aula, estou assumindo que todos os alunos já usaram gramáticas livres de contexto para fazer *parsing* e análise sintática em um contexto de disciplina de compiladores. Se isso não aconteceu, por favor, pare a aula e pergunte.**

|  |
| --- |
| **Exercício 0** |
| *Objetivo: lembrar-se do que é uma gramática livre de contexto e uma árvore sintática* |
| Numa gramática livre de contexto arbitrária, temos as seguintes regras de produção:  S -> A B  A -> A A  A -> a  B -> A B C  B -> b  C -> c  (a) Mostre a árvore sintática que gera a string “a a a b c”  (b) A string “a b c” pode ser gerada por essa gramática? |

|  |
| --- |
| **Exercício 1** |
| *Objetivo: lembrar-se do que é sujeito, objeto e predicado* |
| Na frase: “o lobo soprou a casa”, encontre o sujeito, o verbo, o predicado e o objeto. Como essas estruturas se organizam entre si? |

|  |
| --- |
| **Exercício 2** |
| *Objetivo: aplicar o conceito de gramática de substituições para compor uma frase em português* |
| Uma gramática de substituições funciona da seguinte forma:  Partimos de uma *string de símbolos* que contém somente uma raiz (geralmente é chamada de S, mas aqui chamaremos de FRASE.  Temos uma série de regras que nos permitem substituir um dos símbolos por outro ou outros. Por exemplo, podemos ter:  FRASE -> SUJEITO PREDICADO  FRASE -> PREDICADO SUJEITO  Quando temos mais de uma regra, podemos gerar frases com qualquer uma delas.  Por exemplo, nosso símbolo FRASE poderia ter sido substituído por dois símbolos:  SUJEITO PREDICADO  Daí, temos mais e mais regras:  SUJEITO -> ARTIGO SUBSTANTIVO  SUJEITO -> SUBSTANTIVO  SUBSTANTIVO -> ‘lobo’  ARTIGO -> ‘o’  (a) Quais regras seriam necessárias para montar a frase “o lobo soprou a casa”, de forma que obrigatoriamente passamos por construções como sujeito, predicado e objeto?  (b) Aplique as regras que você construiu (ou seja, a gramática) para montar a árvore sintática que permite analisar a frase “o lobo soprou a casa”. |

|  |
| --- |
| **Exercício 3** |
| *Objetivo: fazer parsing usando as ferramentas do NLTK* |
| Parsing é o processo que nos dá a árvore sintática que gera uma frase, à partir de uma gramátca.  Para fazer parsing, precisamos de (i) definir uma gramática e (ii) definir um algoritmo para fazer parsing.  (a) Em seu notebook, encontre como definimos nossa gramática. Como ela se compara com aquela que você definiu no exercício anterior?  (b) Encontre agora o trecho em que passamos nossa gramática escrita como string para um formato acessível pelas ferramentas do NLTK. Que tipo de objeto é usado para isso?  (c) Encontre o trecho em que definimos qual será o algoritmo a ser usado no processo de parsing.  (d) Por fim, encontre o trecho em que executamos o algoritmo propriamente dito e imprimimos a árvore sintática. Como essa árvore se compara com a que você construiu no exercício 2?  (e) Modifique o código para que ele imprima diretamente a árvore sintática, ao invés de usar o método pretty\_print(). Como a árvore se organiza? |

|  |
| --- |
| **Exercício 4** |
| *Objetivo: relacionar a análise sintática com perguntas sobre o conteúdo do texto* |
| Em uma frase que tem um sujeito, um verbo na voz ativa e um objeto (use a frase “o lobo soprou a casa” como exemplo):  (a) como podemos saber quem é o sujeito?  (b) quem realizou a ação descrita no verbo?  (c) como podemos saber o que o sujeito fez?  (d) sobre quem o sujeito aplicou a ação descrita no verbo? |

|  |
| --- |
| **Exercício 5** |
| *Objetivo: navegar pela árvore sintática gerada pelo parser para responder perguntas quanto ao conteúdo do texto* |
| Veja no notebook o código exemplo para responder à pergunta: “quem soprou a casa”.  (a) Qual é a ideia subjacente que usamos para responder à pergunta?  (b) Como podemos encontrar o sujeito da frase?  (c) Seguindo essa mesma ideia, escreva o trecho de código que permita responder automaticamente às perguntas: “o que o lobo fez?” e “o que o lobo soprou?”. |

|  |
| --- |
| **Exercício 6** |
| *Objetivo: entender como gramáticas livres de contexto probabilísticas podem ajudar a resolver ambiguidades* |
| Em português, podemos ter várias inversões sintáticas. Por exemplo: “ouviram do Ipiranga as margens plácidas” é gramaticalmente válido, assim como “as margens plácidas do Ipiranga ouviram”. Isso não significa, porém, que todas as variações de colocações sejam igualmente prováveis. Por exemplo: na linguagem cotidiana, é muito mais comum termos:  FRASE -> SUJEITO PREDICADO PREDICADO -> VERBO OBJETO  que  FRASE -> PREDICADO SUJEITO  PREDICADO -> OBJETO VERBO  Ou seja:  “Carlos viajou para o Caribe”  É bem mais comum que  “Para o Caribe viajou Carlos”  Podemos resolver esse tipo de ambiguidade de forma computacional, atribuindo probabilidades às produções de uma gramática.  (a) Qual é a ambiguidade sintática que temos na frase “ele entrou na loja de calças”?  (b) Quais regras de produção da gramática modelam essa ambiguidade?  (c) Quais foram as árvores sintáticas geradas? Qual é a probabilidade de cada uma delas ter sido gerada pela gramática?  (d) De acordo com nossa gramática, qual é a interpretação mais provável para esta frase?  (e) Modifique a gramática de forma a inverter qual é a interpretação mais provável da frase. |

|  |
| --- |
| **Exercício 7** |
| *Objetivo: relacionar os nós-folhas da árvore sintática a PoS-tagging* |
| O conteúdo dos nós-folha das árvores sintáticas que fizemos até o momento, na verdade, aplica uma funcionalidade que já conhecemos bem.  Qual é essa funcionalidade? |

|  |
| --- |
| **Exercício 8** |
| *Objetivo: usar uma estratégia de parsing baseada em expressões regulares* |
| Uma outra estratégia de Parsing é definir expressões regulares baseadas em símbolos. Veja o exemplo em seu notebook.  (a) Qual das expressões regulares é equivalente à produção “ENTIDADE -> ART N”?  (b) Como a árvore sintática gerada se compara com as árvores geradas anteriormente?  (c) Em que ordem as regras das expressões regulares são aplicadas?  (d) O que acontece se encontrarmos alguma situação que não é contemplada por uma regra?  (e) Modifique o código para que ele deixe de montar uma árvore sintática completa e somente encontre as entidades (artigo ou preposição + substantivo) presentes na frase de entrada.  (f) Modifique a sua definição de entidades para que ela funcione na frase “O Rio de Janeiro continua lindo”.  (g) A estratégia de parsing com expressões regulares é altamente dependente de uma etapa anterior. Que etapa é essa, e quão frágil é o parsing em relação a erros nessa etapa? |