

8 dicas para buscar resultados da análise morfossintáticas

As análises morfossintáticas obtidas com o módulo de pré-processamento baseado no CoGrOO não sincroniza com a utilização de um banco de dados relacional, como o que o *dadosSemiotica* usa, pois não existe um número pré-determinado de etiquetas resultantes para cada palavra. O *dadosSemiotica* registra o resultado completo para cada sentença num único campo do banco de dados, independente do número de etiquetas, utilizando uma notação compatível com o Tregex e é por meio desse tipo de expressão regular baseada em árvore que a consulta a esses dados é feita.

A semântica das expressões segue a tabela da figura 28. São essas as etiquetas¹ que resultam da análise morfossintática utilizada no CoGrOO e são salvas no banco de dados.

2.3. Categoria gramatical (nós terminais)

símbolo	categoria	
n	nome, substantivo	
prop	nome próprio	
adj	adjectivo	
n-adj	flutuação entre substantivo e adjectivo	
v	v-fin	verbo finito
	v-inf	infinitivo
	v-pcp	particípio
	v-ger	gerúndio
art	artigo	
pron	pron-pers	pronome pessoal
	pron-det	pronome determinativo
	pron-indp	pronome independente (com comportamento semelhante ao nome)
adv	advérbio	
num	numeral	
prp	preposição	
intj	interjeição	
conj	conj-s	conjunção subordinativa
	conj-c	conjunção coordenativa

Ilustração 28: Tabela de categorias gramaticais.

Como os resultados são hierárquicos (em forma de árvore), a utilização do Tregex foi a solução para as buscas. A sintaxe é baseada no Tregex Pattern (figura ss), uma linguagem criada para buscas em dados não previamente indexados, realizando uma busca linear que, embora lenta, pode ser aplicada a um conjunto arbitrário de árvores, sendo um dos recursos disponíveis para processamento de linguagem natural².

1 In: Glossário de etiquetas florestais da Linguateca. URL: <http://www.linguateca.pt/floresta/BibliaFlorestal/anexo1.html>

2 In: Tregex Pattern. URL: <http://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/trees/tregex/TregexPattern.html>

Symbol	Meaning
A << B	A dominates B
A >> B	A is dominated by B
A < B	A immediately dominates B
A > B	A is immediately dominated by B
A \$ B	A is a sister of B (and not equal to B)
A .. B	A precedes B
A . B	A immediately precedes B
A ,, B	A follows B
A , B	A immediately follows B

Ilustração 29: Início da tabela de padrões TRegex.

Uma das implementações previstas para o *dadosSemiotica* v2.0 é a busca Tregex simplificada, via formulários. **NA VERSÃO 1.0, no entanto, é necessário montar a expressão Tregex no campo apropriado do módulo de pós-processamento, no momento de obtenção de estatísticas, sendo possível realizar apenas uma busca por requisição de tabela (figura 30).**

Incluir o conteúdo do corpus e da sentença no CSV

Utilize o campo abaixo para obter o resultado de uma consulta TRegex no arquivo CSV.

Expressão TRegex:

Montar expressão simples

✓ Gerar Estatísticas ✓ Agendar Processamento

Ilustração 30: Exemplo de busca usando expressão TRegex. O preenchimento deste campo resulta em duas novas colunas na tabela de saída e um arquivo explicando a expressão utilizada e as colunas correspondentes.

A tabela de saída, quando solicitada a busca por uma expressão Tregex, conterá duas colunas além das categorias selecionadas: `casou_tregex` (indica presença – número 1 – ou ausência – zero – de presença do resultado da busca na sentença) e `resultado_tregex` mostrando o resultado da busca, permitindo tanto a análise pela presença do resultado quanto pelo tipo de resultado encontrado.

O conjunto de arquivos contidos no `resultado.zip` contém um arquivo `a_mais`, o `analise_morfossintatica.txt`, que contém a expressão buscada e explica as colunas `a_mais`.

O *dadosSemiotica* indica a página do Tregex Pattern para auxiliar na montagem das expressões, no entanto, mesmo assim, esta não é uma tarefa para iniciantes ou leigos em programação.

Para possibilitar o uso dessa ferramenta de buscas por não iniciados, fizemos uma pequena lista de exemplos de buscas que, combinados, permitem a construção de algumas buscas mais frequentes. Cada exemplo está organizado da seguinte forma:

- i) proposta (apresenta o objetivo da busca e a expressão usando variáveis delimitadas por \$);
- ii) fórmula;
- iii) exemplo genérico (apresenta um exemplo genérico e a expressão correspondente);

- iv) exemplo no Vira, Vira (aplica a expressão de forma adequada ao corpus de sentenças do Vira, Vira, poema anexo). As expressões propriamente ditas estão isoladas em linhas com fundo acinzentado e as variáveis, nas expressões, estão marcadas em negrito.

a) Busca por: palavra-chave

i) proposta: Buscar sentenças que contenham uma dada palavra-chave.

ii) Fórmula: a variável está escrita como §palavra§ (substitua §palavra§ pela palavra-chave que deseja buscar, sem os símbolos §):

§palavra§

iii) Exemplo genérico: buscar sentenças que contenham "livre". A fórmula é simplesmente:

livre

iv) Vira, Vira: Busca pela palavra-chave "vira":

vira

```
=> o resultado_tregex foi este, abaixo, em cada linha na qual a busca encontrou coincidência. Foi sempre o mesmo porque a palavra "nuvem" não aparece, por exemplo, no plural.  
# Tree 0  
## match 0 index = 5:  
vira  
=> Note que o título, que contém a palavra-chave iniciada por maiúsculas, não acusou sua presença, ou seja, a análise é case sensitive.  
=> A sentença que continha duas vezes a palavra-chave apresentou um resultado diverso:  
# Tree 0  
## match 0 index = 11:  
vira  
## match 1 index = 15:  
vira
```

b) Busca por: sequência de palavras-chave

i) proposta: Buscar sentenças que contenham uma dada palavra-chave seguida imediatamente por outra.

ii) Fórmula: a variável 1 (primeira palavra-chave) está escrita como §palavra1§ e a variável 2 (segunda palavra-chave) é §palavra2§:

§palavra1§.§palavra2§

iii) Exemplo genérico: buscar sentenças que contenham "software livre". A fórmula é simplesmente:

software.livre

iv) Vira, Vira: Busca pela sequência de palavras-chave "vira casaca":

vira.casaca

```
=> o resultado_tregex foi este, abaixo, na única linha na qual a busca encontrou coincidência.  
# Tree 0  
## match 0 index = 5:
```

vira

=> Observe que a sequência obrigatória é indicada pelo ponto, sem espaços.

c) Busca por: palavra tomada como lema

i) proposta: Buscar sentenças que apresentem uma dada palavra, tomada como lema, e seus derivados.

ii) Fórmula: a variável está escrita como §lema§ (substitua §lema§ pelo lema que deseja buscar; o lema entra sem os símbolos §):

```
/[^-]+-[^-]+-§lema§$/  
/
```

iii) Exemplo genérico: buscar sentenças que apresentem a palavra "livre" (e derivados como livres). É importante notar que palavras como "livremente" são consideradas, em si, como lemas pelo programa e não vão ser incluídas no resultado de uma busca por "livre":

```
/[^-]+-[^-]+-livre$/  
/
```

iv) Vira, Vira: Busca pelo lema "nuvem":

```
/[^-]+-[^-]+-nuvem$/  
/
```

=> o resultado_tregex foi este, abaixo, em cada linha na qual a busca encontrou coincidência. Foi sempre o mesmo porque a palavra "nuvem" não aparece, por exemplo, no plural.

```
# Tree 0
```

```
## match 0 index = 6:
```

```
(n-F=S-nuvem nuvem)
```

d) Busca por: palavra tomada como lema quando há mais de um lema

i) proposta: Buscar sentenças que apresentem uma dada palavra, tomada como lema, e seus derivados. Obtém todos os resultados para:

§lema§

seqal§lema§

seqal|seqal|§lema§

§lema§|seqal

onde "seqal" é uma sequência de letras qualquer.

ii) Fórmula: a variável está escrita como §lema§ (substitua §lema§ pelo lema que deseja buscar; o lema entra sem os símbolos §):

```
/[^-]+-[^-]+-(.*\|)*\b§lema§\b/  
/
```

iii) Exemplo genérico: buscar sentenças que apresentem a palavra "livre" (e derivados como livres). Mesmo essa busca não inclui palavras como "livremente":

```
/[^-]+-[^-]+-(.*\|)*\blivre\b/  
/
```

iv) Vira, Vira: Busca pelo lema "vira":

```
/[^-]+-[^-]+-(.*\|)*\bvira\b/
```

=> o resultado_tregex foi este, abaixo, em cada linha na qual a busca encontrou coincidência. Foi sempre o mesmo porque a palavra "nuvem" não aparece, por exemplo, no plural.
 # Tree 0
 ## match 0 index = 6:
 (n-F=S-nuvem nuvem)
 => note que o \b indica limites de palavras.

e) Busca por: verbo

i) proposta: buscar sentenças que apresentem um verbo determinado (e conjugações)

ii) Fórmula: a variável está escrita como §verbo§ (substitua §verbo§ pelo verbo que deseja buscar; o verbo entra sem os símbolos §):

```
/^v[^-]+-[^-]+-(.*\|)*\b§verbo§\b/
```

iii) Exemplo genérico: buscar sentenças que apresentem o verbo "usar" (e conjugações).

```
/^v[^-]+-[^-]+-(.*\|)*\busar\b/
```

iv) Vira, Vira: Busca pelo verbo "virar":

```
/^v[^-]+-[^-]+-(.*\|)*\bvirar\b/
```

=> o resultado_tregex foi este, abaixo, em cada linha na qual a busca encontrou coincidência.
 # Tree 0
 ## match 0 index = 4:
 (vfin-PR=3S=IND-virar|ver vira)

f) Busca por: verbo determinado seguido por verbo qualquer

i) proposta: buscar sentenças que apresentem um verbo determinado (e conjugações) seguido de outro verbo qualquer.

ii) Fórmula: a variável está escrita como §verbo§ (substitua §verbo§ pelo verbo que deseja buscar; o verbo entra sem os símbolos §):

```
/^v[^-]+-[^-]+-(.*\|)*\b§verbo§\b/ . @vfin|vinf
```

iii) Exemplo genérico: buscar sentenças que apresentem o verbo "usar" (e conjugações).

```
/^v[^-]+-[^-]+-(.*\|)*\busar\b/ . @vfin|vinf
```

iv) Vira, Vira: Busca pelo verbo "virar":

```
/^v[^-]+-[^-]+-(.*\|)*\bvirar\b/ . @vfin|vinf
```

=> no Vira, Vira não temos nenhuma ocorrência deste tipo.
 => observe que, para conseguir uma sequência formada por um verbo qualquer seguido de um verbo determinado, basta inverter os fatores antes e depois do ponto:
 vfin|vinf . @/^v[^-]+-[^-]+-(.*\|)*\b§verbo§\b/

```
=> @vfin|vinf vai casar com vfin, vinf, vger etc.3
```

g) Busca por: frase em primeira do singular

i) proposta: Buscar sentenças em primeira pessoa do singular.

ii) Fórmula:

```
/=1S=/
```

iii) Alternativas: para obter segunda ou terceira pessoa, substitua o número por 2 ou 3 respectivamente; para obter pessoa do plural, substitua S por P.

iv) Vira, Vira: Busca por sentenças na terceira pessoa do singular:

```
/=3S/
```

```
=> os resultado_tregex foram:  
"# Tree 0  
## match 0 index = 4:  
(vfin-PR=3S=IND-virar|ver vira)"  
  
"# Tree 0  
## match 0 index = 10:  
(vfin-PR=3S=IND-mexer mexe)"  
  
"# Tree 0  
## match 0 index = 10:  
(vfin-PR=3S=IND-virar|ver vira)"  
  
=> para ter certeza de buscar somente verbos, prefira usar:  
/^\v.*=1S=  
  
=> para verificar se faz parte de um sintagma verbal:  
@VP << /^\v.*=3S=
```

h) busca por verbo no infinitivo

i) proposta: Buscar sentenças que contenham verbo no infinitivo.

ii) Fórmula:

```
@vinf
```

iii) Alternativas: para obter verbos em outras conjugações, substitua vinf pelas opções presentes na tabela da figura 28.

iv) Vira, Vira: Busca por sentenças com verbo finito:

```
@vfin
```

```
=> os resultado_tregex foram:  
"# Tree 0
```

3 Conforme figura 28.

```
## match 0 index = 4:  
(vfin-PR=3S=IND-virar|ver vira)"  
  
"# Tree 0  
## match 0 index = 10:  
(vfin-PR=3S=IND-mexer mexe)"  
  
"# Tree 0  
## match 0 index = 10:  
(vfin-PR=3S=IND-virar|ver vira)"  
  
=> para ter certeza de buscar somente verbos, prefira usar:  
/^\v.*=vfin=/  
  
=> para verificar se faz parte de um sintagma verbal:  
@VP << /^\v.*=vfin=/
```

i) Montando sua busca

Podemos dizer que uma busca Tregex é definida por uma semântica e uma sintaxe. A semântica diz respeito às etiquetas gramaticais e aos lemas e vocábulos que serão buscados. A sintaxe diz respeito às especificações de um termo de busca e às relações entre termos de busca.

Além dos exemplos acima, que podem ter seus termos recombinaados e as especificações utilizadas com outros lemas e vocábulos, você pode montar outras buscando especificações no Tregex Pattern (figura 29) e etiquetas na tabela de categorias gramaticais (figura 28).

Como, **NA VERSÃO 1.0, você só pode realizar uma busca por vez**, para poder cruzar os resultados de mais de uma busca recomendamos os seguintes passos:

- 1) gere a tabela com as categorias desejadas e a primeira busca. Salve.
- 2) solicite nova tabela, desta vez apenas com o texto e a próxima busca. Salve.
- 3) repita o passo 2 até realizar todas as buscas desejadas.
- 4) abra num editor de planilhas (como o Calc, do libreoffice) o arquivo csv da primeira tabela.
- 5) altere o nome dos campos de casou_tregex e resultado_tregex incluindo uma informação sobre a busca (por exemplo, casou_tregex_1s resultado_tregex_1s para primeira pessoa do singular).
- 6) salve a tabela com o nome tabelaCompleta.csv, escolhendo o tipo “texto csv” na hora de salvar o arquivo. Mantenha a tabelaCompleta aberta.
- 7) abra a segunda tabela e copie os resultados das colunas casou_tregex e resultado_tregex na tabelaCompleta, ao lado das últimas colunas; modifique o nome incluindo a informação sobre a busca e salve.
- 8) repita a etapa 7 para todas as buscas realizadas.

Pronto. A tabela da figura 31 contém os resultados de duas buscas tregex para o Vira,Vira.