

Processamento Natural de Linguagem

Aula 2

Expressões regulares e Pré-processamento



Semantix[®]

All about data

Quem sou eu?



Filipe Theodoro

Cientista de dados da Semantix
Instrutor do Semantix Academy

Contatos

filipe.theodoro@semantix.com.br
linkedin.com/in/filipe-theodoro



Ementa

- Expressões regulares
- Tokenização
- N-gram
- Normalização
- Remoção de stopwords
- Correção ortográfica
- Part Of Speech

Expressão Regular



Expressão Regular

- Como fazer para encontrar uma data no formato DD/MM/AAAA no texto ao lado?

Esse é um texto de exemplo que simula um contrato firmado pelo aluno José da Silva nascido em 24/12/1995 com CPF 012.345.678-90 natural da cidade de João Monlevade – MG. O aluno está se comprometendo a assistir as aulas do curso da SEMANTIX ACADEMY com duração de 49 horas.

Esse contrato pode ser validado no site:

<https://www.validarlink.com.br>

São Paulo, SP
04/05/2005

Expressão Regular

- Também conhecido como REGEX
- Identificar padrões de caracteres em um pedaço de texto
- Feito a partir da análise sintática
- Utilizado para:
 - Procura
 - Substituição
 - Validação de formatos
 - Realce de sintaxe

```
1  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML
2  <html>
3      <head>
4          <title>Example</title>
5          <link href="screen.css" rel="sty
6      </head>
7      <body>
8          <h1>
9              <a href="/">Header</a>
10         </h1>
11         <ul id="nav">
12             <li>
13                 <a href="one/">One</a>
14             </li>
15             <li>
16                 <a href="two/">Two</a>
17             </li>
```

Expressão Regular

- Caracteres Especiais
 - [] – Encontra 1 e apenas 1 caractere em um grupo de caracteres
 - [Aa] - encontra a letra 'A' ou 'a'
 - [a-z] – encontra qualquer letra minúscula no intervalo de a até z
 - ^ - Negação ou início de uma frase
 - [^Aa] – encontra tudo menos 'A' e 'a'
 - ^[Aa] – encontra 'A' ou 'a' apenas se for o início do texto
 - \$ - Encontra o final de uma frase
 - . – Encontra qualquer coisa menos nova linha
 - | - Ou
 - tigre|urso – encontra tigre ou urso
 - \ - Serve para encontrar caracteres especiais
 - \\$ - Encontra '\$' no texto

Expressão Regular

- Quantificadores
 - $\{n\}$ – encontra exatamente n vezes
 - $[a]\{5\}$ – encontra 'aaaaa' no texto
 - $\{n,\}$ – encontra n ou mais vezes
 - $[a|b]\{2,\}$ – encontra 'aa', 'bb', 'aba', 'aaa', etc
 - $\{n,m\}$ – encontra um intervalo entre n e m
 - $[a]\{3,4\}$ – encontra 'aaa' e 'aaaa'
 - $?$ – o mesmo que $\{0,1\}$
 - $*$ – o mesmo que $\{0,\}$
 - $+$ – o mesmo que $\{1,\}$

Expressão Regular

- Caracteres não imprimíveis
 - \n – nova linha
 - No Windows é escrito como \r\n
 - \t – espaço tab
 - \v – espaço tab vertical
 - \b – espaço em branco entra uma letra e uma não letra

Expressão Regular

- Classes de caracteres
 - `\d` – Qualquer dígito ou `[0-9]`
 - `\D` – Qualquer não dígito ou `[^0-9]`
 - `\w` – Qualquer parte de uma palavra ou `[A-Za-z0-9_]`
 - `\W` – Qualquer parte de uma palavra sem letra ou `[^a-zA-Z0-9_]`
 - `\s` – Espaço em branco ou `[\t\r\n]`
 - `\S` – Qualquer coisa menos espaço em branco ou `[^\t\r\n]`

Expressão Regular

- Olhar para frente e olhar para trás
 - Olhar para frente positivo
 - `Iron(?=man)` – Encontra *Iron* apenas se for seguido de *man*
 - Olhar para frente negativo
 - `Iron(?!man)` – Encontra *Iron* apenas se NÃO for seguido de *man*
 - Olhar para trás positivo
 - `(?<=Iron)man` – Encontra *man* apenas se vier depois de *Iron*
 - Olhar para trás negativo
 - `(?<!=Iron)man` – Encontra *man* apenas se não vier depois de *Iron*

Tokenização



Tokenização

- Tokenizar ou transformar em tokens.
- *Quebrar um texto em pedaços menores*
- *Podemos quebrar o texto em palavras, frases, parágrafos, e até sílabas.*

“Esse é um exemplo de um texto. A partir desse texto vamos aplicar tokenização e separar os tokens.”



[‘Esse’, ‘é’, ‘um’, ‘exemplo’, ‘de’, ‘um’, ‘texto’, ‘.’, ‘A’, ‘partir’, ‘desse’, ‘texto’, ‘vamos’, ‘aplicar’, ‘tokenização’, ‘e’, ‘separar’, ‘os’, ‘tokens’, ‘.’]

[“Esse é um exemplo de um texto”,
“A partir desse texto vamos aplicar tokenização e separar os tokens”]

N-grama



N-gram

- Sequencia de N itens de uma amostra de texto.
- Pode ser fonemas, sílabas, letras ou palavras.
- Ajuda a entender um pouco mais do contexto dos itens.

“Esse é um exemplo de um texto.”



Uni grama = ['Esse', 'é', 'um', 'exemplo', 'de', 'um', 'texto']

Di grama = ['Esse_é', 'é_um', 'um_exemplo', 'exemplo_de', 'de_um', 'um_texto']

Tri grama = ['Esse_é_um', 'é_um_exemplo', 'um_exemplo_de', 'exemplo_de_um', 'de_um_texto']

Normalização



Normalização

- Deixar o texto todo em um padrão bem definido.
- Exemplos:
 - Letras minúscula
 - Sem caracteres especiais [ç, ã, é, ...]

“Esse é um exemplo de um texto escrito por João Rebouças.”



“esse é um exemplo de um texto escrito por joão rebouças.”

“esse e um exemplo de um texto escrito por joao reboucas.”

Remoção de stopwords



Remoção de stopwords

- Stopwords são palavras que por serem muito comuns não agregam valor algum ao texto.
- Exemplos:
 - de, do, da
 - e, a, o, é
 - como, por, com, tem
 - um, uma, uns
 - esse, essa, este, estes

“Esse é um exemplo de um texto escrito por João Rebouças.”



“exemplo texto escrito João Rebouças.”

Correção ortográfica

Correção ortográfica

- Palavras escritas erradas podem levar a um entendimento equivocado.
- Identificar palavra escrita de forma errada.
- Substituir pela forma correta.
- Tipos de edições:
 - Remoção – remove uma letra
 - Abaccate => Abacate
 - Transposição – troca duas letras adjacentes
 - Duivda => Duvida
 - Substituição – troca uma letra por outra
 - Mudamça => Mudança
 - Inserção – adiciona uma letra
 - Lingagem => Linguagem

POS tagging



POS tagging

- Marcação das partes da oração.
- Part of speech tagging.
- Aplicar análise sintática na frase para identificar as partes.
- Identificar:
 - Verbo
 - Sujeito
 - Substantivo

O rato roeu a roupa do rei do Roma.



[('O', 'ART'), ('rato', 'N'), ('roeu', 'V'), ('a', 'ART'), ('roupa', 'N'), ('do', 'KS'), ('rei', 'N'), ('de', 'PREP'), ('Roma', 'NPROP')]

Bibliografia

- Expressões regulares - <https://docs.python.org/pt-br/3.8/library/re.html>
- Stop words português - <https://gist.github.com/alopes/5358189>
- Correção ortográfica - <http://norvig.com/spell-correct.html>
- POS tagging - <http://nilc.icmc.usp.br/macmorpho/macmorpho-manual.pdf>



Semantix[®]

All about data

filipe.theodoro@semantix.com.br

www.semantix.com.br