

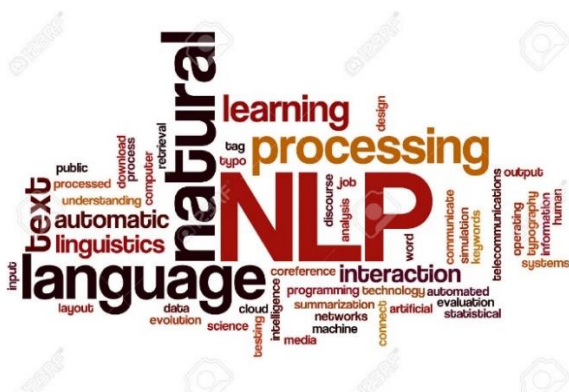
Data de Entrega: **01/10/2018** - Enviar ao email [rinaldo.ufrpe@gmail.com](mailto:rinaldo.ufrpe@gmail.com) com o assunto: [MT – LE 02]

## 2ª. LISTA DE EXERCÍCIOS

### (1) Usando O NLTK

Usando o documento " **NLP.txt**" como entrada, gere os seguintes wordclouds:

1. Um *word cloud* (ver exemplo na figura abaixo) contendo apenas os lemas dos 20 *substantivos* mais frequentes no documento.
2. Um *word cloud* contendo apenas os lemas dos 20 *verbos mais frequentes*



3. Use a lista de stopwords (**stopwords.txt**) que serve para eliminar as palavras muito frequentes presentes no documento **Corpus\_en\_NER.txt** antes de gerar o word cloud, e refaça os itens (1) e (2) acima.

As suas word clouds ficaram mais informativa em ambos os casos? Se sim, por que?

4. Gere as figuras as árvores de parsing constituinte da seguinte sentença:

*The last love letter I wrote was probably about 10 years ago.*

- a. Existe alguma sentença subordinada na frase acima?
- b. Como ela é identificada na árvore de parsing?

### Mais sobre stopwords

- <http://xpo6.com/list-of-english-stop-words/>
- <http://www.lextek.com/manuals/onix/stopwords1.html>

### Mais sobre word clouds:

- <http://www.boostlabs.com/what-are-word-clouds-value-simple-visualizations/>

## (2) Usando o CoreNLP:

---

Usando o documento **Corpus\_en\_NER.txt** como entrada, crie um pipeline no CoreNLP ([corenlp.run](http://corenlp.run)) composto pelas seguintes sub-tarefas

*. tokenize, sentence splitting, pos tagging, lemma, NER e dependency parsing.*

Baseando nas anotações geradas pelo CoreNLP, responda:

1. Quais os verbos identificados e quais seus respectivos lemmas?
  - a. O que acontece quando o verbo está no passado quanto ao seu lemma?
2. Quais as entidades nomeadas encontradas?
3. Considerando apenas as primeiras 3 sentenças do arquivo **NLP.txt**
  - a. quais os tipos distintos de dependências gramaticais encontradas?
  - b. O que cada uma delas significam?

### (3) Implementando um Sumarizador Simples de Notícias

---

Usando o corpus de notícias "**News.zip**" contendo 5 notícias como entrada, implemente um sumarizador extrativo que, para cada documento, seleciona as frases mais relevantes de acordo com o seguinte algoritmo:

1. Realizar o pré-processamento: *tokenization*, *sentence splitting* e *NER*.
2. Para cada sentença  $S_i$  de um documento, calcular um **score global** para  $S_i$  definida pela seguinte fórmula:

**Score-global-de-uma-sentença( $S_i$ ) =**

$$1 + (2 * \text{número de entidades nomeadas que } S_i \text{ contém}) / (N + \text{score(position)})$$

Onde:

- **score(position)** =  $1 - \text{index\_da\_sentença} / N$
- **Index da sentença** = número de ordem dela na notícia: 1 para a primeira sentença, 2 para a segunda sentença, etc.
- **N** = número total de sentenças da notícia

3. Ordenar as frases segundo os scores acima calculados

3. Selecionar apenas 30% das frases ordenadas acima.

**Responda:**

- a. O que você pode dizer dos sumários gerados? Eles são razoáveis para você?
- b. Eles apresentam algum problema de coerência entre as frases selecionadas usadas para gerar o sumário final?

---

### Material extra de apoio para os exercícios

Ver os links de material de apoio disponível no drive virtual da disciplina.