Analisador Textual

- Alexandre Albano
- Filipe Salgado

UGC – User Generated Content

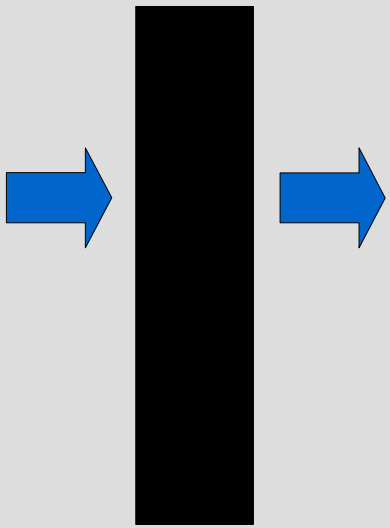
- Hoje em dia os usuários podem gerar conteúdo de diversas maneiras:
  - Blogs;
  - Enviando mensagens para outras pessoas;
  - Respondendo ou colocando perguntas em fóruns;
  - Artigos;
  - E uma série de outros mecanismos.
- Como extrair informação de todo esse conteúdo?

Objetivo

- Calcular um vetor a partir de elementos textuais não estruturados

Japão apresenta tela OLED que poderá ser base para TV "dobrável"

Pesquisadores da emissora de televisão pública japonesa NHK desenvolveram uma tela flexível de 5,8 polegadas com tecnologia OLED (diodo emissor orgânico de luz).



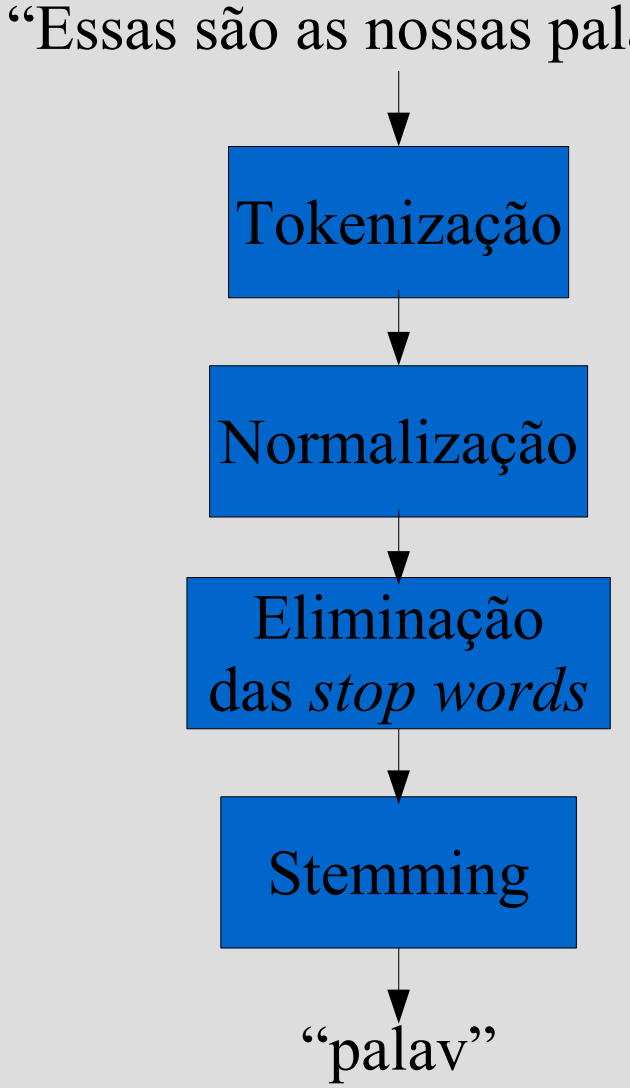
[emissora, emissor, 0.324441]
[tela, tel, 0.286131691759607]
[oled, oled, 0.28613169175O7]
[japão, japa, 0.235702260384]
[ser, ser, 0.235702260395584]
[apresenta, apresent, 0.235584]
[tv, tv, 0.2357022603955158 4]
[base, bas, 0.235702260395 84]
[dobrável, dobravel, 0.235584]
...

Analisador Textual

- Ferramenta que nos permite extrair termos e seus pesos associados para construir uma representação vetorial dos termos de um texto.

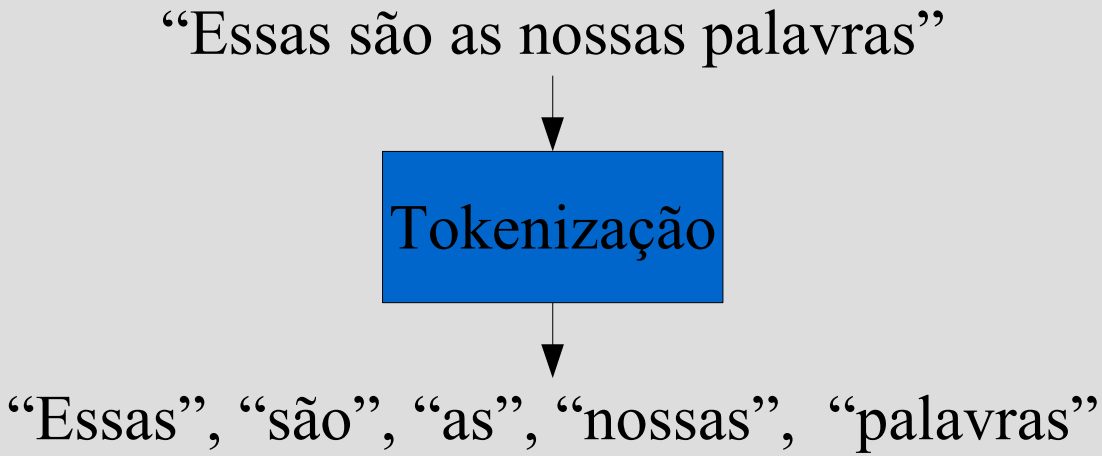
- Essa representação vetorial dos termos pode ser usada para:
  - Gerar metadados sobre o usuário;
  - Criar tag clouds;
  - Construir modelos preditivos;
  - Formar uma base para desenvolver um mecanismo de recomendação baseado em conteúdo;
  - Etc

Analisador Textual



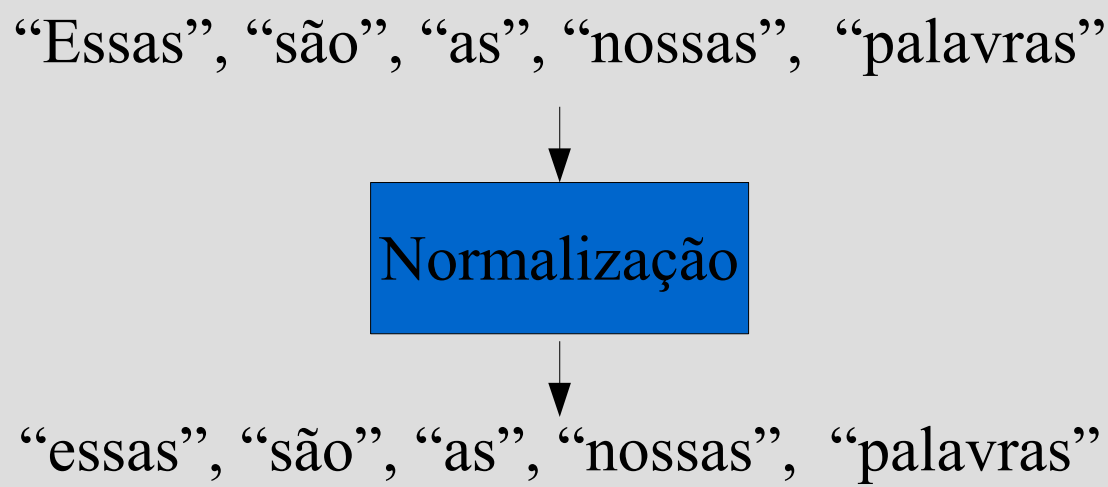
Tokenização

- Interpreta o texto para transformá-lo em termos.
- Analisadores mais sofisticados conseguem extrair frases.



Normalização

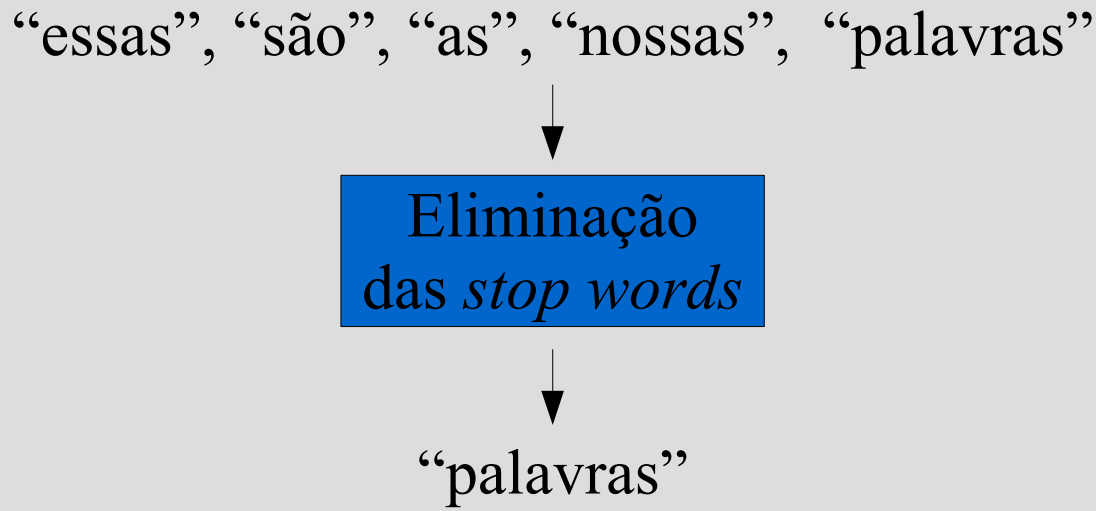
- Conversão do texto para letras minúsculas.



Eliminação de *stop words*

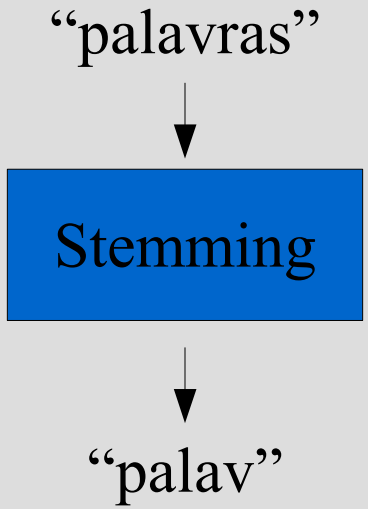
- Eliminação dos termos que aparecem com muita frequência nos textos.
- Varia de língua para língua.
- Existem listas de *stop words* na Internet.
- Dependendo do analisador, é possível acrescentar *stop words*.

Eliminação das *stop words*



Stemming

- Conversão dos termos em suas raízes.
- Eliminação de plural.
- Também é dependente da linguagem.



O que foi feito

- Foi criada uma Façade para o código contido em [Alag-src], com 6 métodos abrangendo 4 funcionalidades básicas.
- Foram parametrizadas listas de frases, sinônimos e *stop words* (sendo que a lista de *stop words* padrão é uma lista “pt-BR”).
- Foi criado um pacote com os jar's necessários.

Funcionalidades básicas

- `TagMagnitudeVector analisar (String texto)`
- Devolve um vetor de termos representado pela classe `TagMagnitudeVector`.
- Esse vetor consiste no mapeamento dos termos (Tag) do `texto` em pesos (TagMagnitude).
- Nesse caso, os pesos são a frequência das palavras.

Funcionalidades básicas

- `TagMagnitudeVector analisar (String descricaoCurta, String texto)`
- Devolve um vetor de termos, como no exemplo anterior.
- Porém, neste caso, os pesos dos termos presentes em `descricaoCurta` recebem peso maior.

Funcionalidades básicas

- `TagMagnitudeVector analisar (String descricaoCurta, String texto, List<String> outrosTextos)`

- Devolve um vetor de termos, conforme o exemplo anterior.

- Diminui o peso dos termos que também ocorrem na lista `outrosTextos`, usando *tf*idf* para o cálculo dos pesos.

TF x IDF

- *tf* indica a frequência de um termo em um documento.

- *idf* estima o uso do termo na coleção inteira. Quanto menos usado for o termo, maior o idf.

$$w_{i,d} = tf \times idf = tf_{i,d} \times \log(n / df_i)$$

$tf_{i,d}$ = frequência do termo i no documento j

n = número total de documentos

df_i = número de documentos que contém o termo i

Funcionalidades básicas

- `TagMagnitudeMatrix analisar(List<String> textos, Boolean consideraIdf)`

- Devolve um conjunto de vetores de termos. Esse conjunto é representado pela classe `TagMagnitudeMatrix`.

- Cada vetor desse conjunto consiste no mapeamento de termos (Tag) em pesos (TagMagnitude) de um elemento da lista `textos`.

Conclusão

- Muito conteúdo sendo gerado pelos usuários
- Temos que extrair todo tipo de informação desse conteúdo que possa agregar valor ao nosso site ou aplicação
- Através do Analisador Textual é possível saber quais os termos mais relevantes de textos não-estruturados
- A partir desses termos podemos gerar metadados sobre os usuários, tag clouds, modelos preditivos, ...

Exemplo

- http://pcgerosa.ime.usp.br:8180/analisadorTextual

Perguntas

?????

Referências

- [ALAG] ALAG, S. (2009), Collective Intelligence in Action, ISBN 1933988312.
- [ALAG-src] Collective Intelligence in Action – código - http://www.manning.com/alag/ciia-src.zip
- [IIR] Introduction to Information Retrieval - http://www-csli.stanford.edu/~hinrich/information-retrieval-book.html