Feature Selection

Recuperação da Informação

**Sumário**

[1. Introdução 3](#_Toc80544014)

[2. Estudo 3](#_Toc80544015)

[3. Estudar e analisar bigrama e trigrama 3](#_Toc80544016)

[4. Estudar a biblioteca Gensim com Python 3](#_Toc80544017)

[5. Testar exemplos 3](#_Toc80544018)

[6. Referências 4](#_Toc80544019)

# **Introdução**

**Aula dia ():** Aula

# **Estudo**

**Livro:** Manning, Raghavan, Schutze - Introduction to information retrieval (2008), páginas 251 a 258 (275-)

**Livro:** Aggarwal - Machine Learning for Text (2018), páginas 117 a 122

# **Estudar e analisar bigrama e trigrama**

**xxx**

# **Estudar a biblioteca Gensim com Python**

**xxx**

# **Testar exemplos**

from gensim.models.phrases import Phrases

modelo\_fraseador = Phrases(frases, min\_count = min\_c, threshold = threshold,

scoring = scoring)

min\_c = 120, threshold = 0.6, scoring = 'npmi'

trabalhar com notícias de 2016 a 2021

[segunda-feira 15:53] Frederico Shu

Depois de treinar o modelo fraseador:

fraseador = Phraser(modelo\_fraseador) # exportar e fechar o modelo para economizar RAM

fraseador.save(arqfr) # gravar o modelo treinado

fraseador = Phraser.load(arqfr) # carregar fraseador

documento\_analisado = fraseador[documento]

A ideia então seria implementar o fraseador e estudar seleção de atributos

# **Referências**

<https://www.amazon.com.br/Introduction-Information-Retrieval-English-Christopher-ebook/dp/B00AHTN5JM>