06 Text Mining

IC 22/23

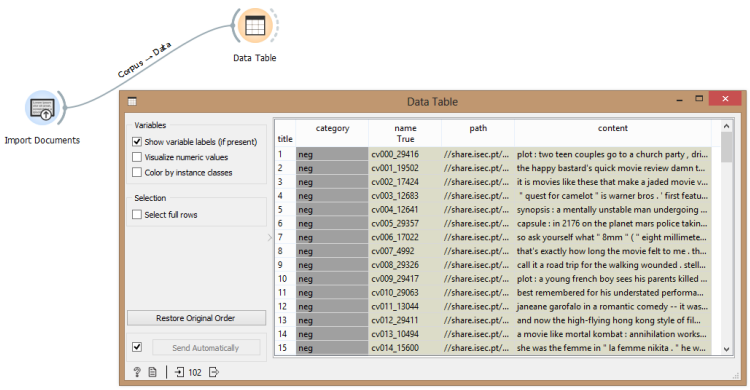
Text Mining

- Tipos de Atributos ou variáveis (features)
  - Contínuas
    - descrevendo quantidade.
  - Categóricas
    - descrevendo classes (número fixo, obtidas a partir de uma lista).
  - Texto
    - variáveis como “string”.

- Classificação de Texto
  - Como representar e apresentar ao classificador?
    - Benchmark : “Sentiment Analysis of Movie Reviews”
      - Conjunto de revisões - textos, classificados como revisão “positiva” ou “negativa”.
      - http://www.cs.cornell.edu/People/pabo/movie-review-data/
    - Tarefa: dado um texto de revisão, queremos classificar a revisão como “positiva” ou “negativa” com base no conteúdo do texto.

Text Mining

- ….
  - [2]



3

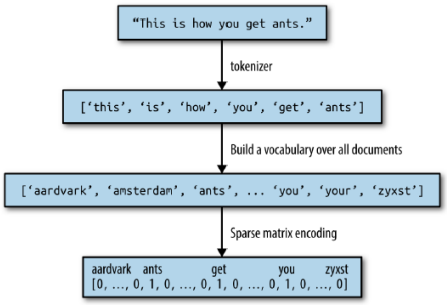
Representação

- “Bag of Words”
  - A forma mais simples, mas eficaz, e comumente usada de representação texto:
    - Considera apenas o número de vezes que cada palavra aparece em cada texto, não considerando a estrutura dos documentos.
    - O cálculo da representação para um conjunto de documentos consiste usualmente em nas três etapas:
      - ***Tokenization*** - Dividir cada documento em palavras – designadas de *tokens*.
      - **Construção de vocabulário** – formar um vocabulário com todas as palavras que aparecem no conjunto, ordenado por exemplo alfabeticamente.
      - ***Codificação*** - Para cada documento, “contar” quantas vezes cada uma das palavras do vocabulário aparece nesse referido documento.

4

Representação

- ...
  - [1]




```
bards_words =["The fool doth think he is wise,",
              "but the wise man knows himself to be a fool"]

from sklearn.feature_extraction.text import CountVectorizer
vect = CountVectorizer()
vect.fit(bards_words)

print("Vocabulary size: {}".format(len(vect.vocabulary_)))
print("Vocabulary content:\n {}".format(vect.vocabulary_))
```

```
Vocabulary size: 13
Vocabulary content:
 {'the': 9, 'himself': 5, 'wise': 12, 'he': 4, 'doth': 2, 'to': 11, 'knows': 7,
  'man': 8, 'fool': 3, 'is': 6, 'be': 0, 'think': 10, 'but': 1}
```

```
bag_of_words = vect.transform(bards_words)

print("Dense representation of bag_of_words:\n{}".format(
    bag_of_words.toarray()))
```

```
Dense representation of bag_of_words:
[[0 0 1 1 1 0 1 0 0 1 1 0 1]
 [1 1 0 1 0 1 0 1 1 1 0 1 1]]
```

5

Representação

- ...
  - Uma forma de retirar "palavras não informativas" é descartando palavras que são muito frequentes – duas metodologias:
    - Lista específica (por língua) de palavras irrelevantes – "stopwords"

```
['above', 'elsewhere', 'into', 'well', 'rather', 'fifteen', 'had', 'enough',
 'herein', 'should', 'third', 'although', 'more', 'this', 'none', 'seemed',
 'nobody', 'seems', 'he', 'also', 'fill', 'anyone', 'anything', 'me', 'the',
 'yet', 'go', 'seeming', 'front', 'beforehand', 'forty', 'i']
```

    - Lista de palavras que aparecem com muita frequência em todos os documentos (depende do conjunto de textos)

6

Representação

- …
  - tf–idf "*term frequency–inverse document frequency*"
    - O princípio do método baseia-se em dar maior "peso" a termos que apareçam frequentemente num determinado documento, mas não em muitos documentos do conjunto

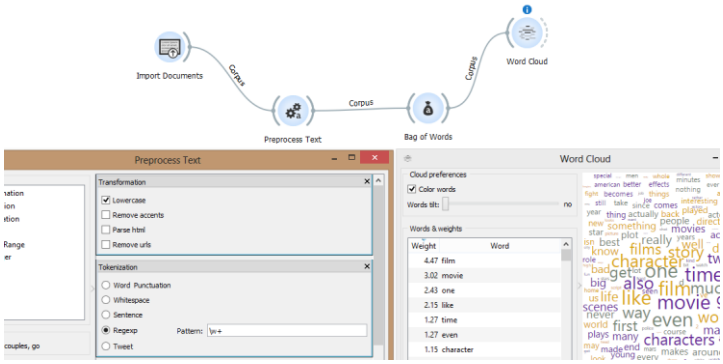
$$\text{tfidf}(w, d) = \text{tf} \log\left(\frac{N+1}{N_w+1}\right) + 1$$

- *N* - número de documentos no conjunto de treino
- *Nw* - número de documentos no conjunto de treino em que a palavra "w" aparece
- *tf* (term frequency) - número de vezes que a palavra "w" aparece no documento em análise "*d*"

7

Representação

- …



8

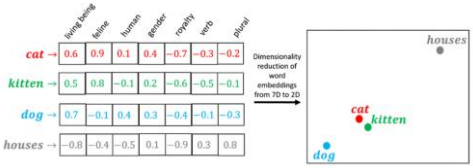
Representação

- …
  - Stemming
    - Representar cada palavra usando apenas o seu radical, exemplo:
      - “replace", "replaced", "replacement", "replaces", and "replacing“
  - Lemmatization
    - Usar um dicionário de palavras (um sistema pericial e verificado por humanos), em que o papel da palavra na frase é levado em consideração

9

Representação

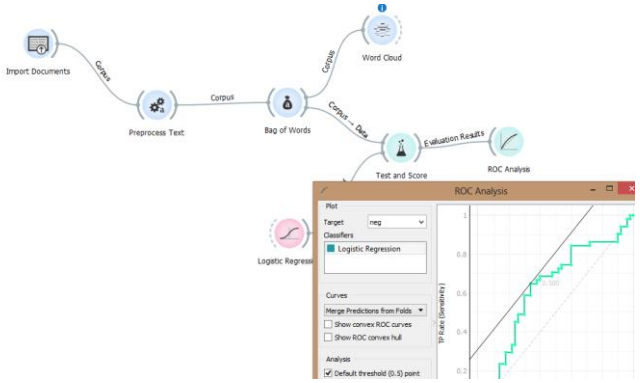
- …
  - n-Grams
    - Bag-of-Words com mais do que uma palavra [1]
      - “*it’s bad, not good at all*” ou “*it’s good, not bad at all*”
        - têm exatamente a mesma representação! - pois não é considerada a ordem das palavras.
      - Uma forma de <u>capturar o contexto</u> ao usar uma representação de “bag of words”, é não considerar apenas a contagem de *tokens* simples, mas também a contagem de pares ou tripletos de *tokens* que apareçam em conjunto. Exemplo:
        - “not like”
  - Word Embedding
    - word2vec



10

Aplicação

- …
  - [2]

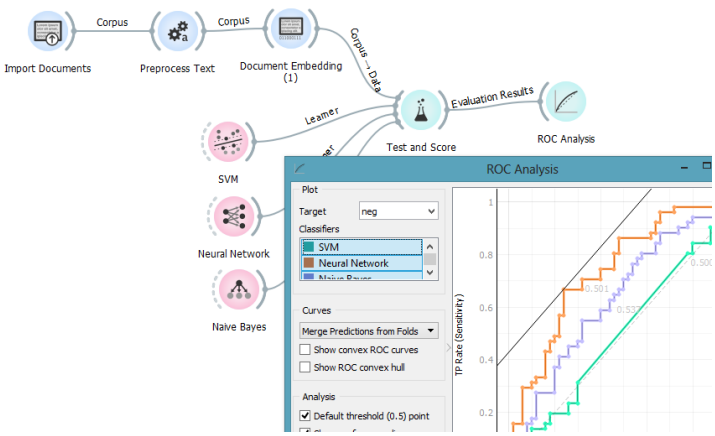


Aplicação

- …

  Word Embedding [4]
  - Permite uma representação eficiente e densa;
  - Termos semelhantes têm uma codificação semelhante;
  - Os parâmetros são o resultado de treino de uma rede neuronal.

  - https://www.tensorflow.org/text/guide/word_embeddings



Aplicação

- …
  - Ferramentas
    - https://scikit-learn.org/stable/tutorial/text_analytics/working_with_text_data.html
    - Orange-Text https://orange3-text.readthedocs.io/en/latest/
    - spacy https://spacy.io/
    - nltk https://www.nltk.org/
    - …

13

Referências

- [1] Müller, A. C., & Guido, S. (2016). *Introduction to machine learning with Python: a guide for data scientists*. " O'Reilly Media, Inc.".
- [2] https://orangedatamining.com/
- [3] https://scikit-learn.org
- [4] https://www.tensorflow.org/text/guide/word_embeddings

14