

Exercício

Faça um classificador que leia os tweets do arquivo Tweets.csv e classifique sentenças fornecidas pelo usuário como 'positiva', 'neutra' ou 'negativa'. Requisitos:

- Utilizar tokenization;
- Utilizar bag of words;
- Classificar usando bayes;
- Airline_sentiment, text

$$P(A | B) = \frac{P(B | A) \cdot P(A)}{P(B)}$$

A, B = events

$P(A|B)$ = probability of A given B is true

$P(B|A)$ = probability of B given A is true

$P(A), P(B)$ = the independent probabilities of A and B

Exercício

Agora aplique a técnica de stopwords no seu script e analise a diferença entre os resultados

$$P(A | B) = \frac{P(B | A) \cdot P(A)}{P(B)}$$

A, B = events

$P(A|B)$ = probability of A given B is true

$P(B|A)$ = probability of B given A is true

$P(A), P(B)$ = the independent probabilities of A and B

Exercício para dia 17 equipes de até 3

Os dados estão no arquivo imdb.csv;

Limpar dados com stopwords , regex, stem ou lemma, (nltk ou spacy)

Criar um classificador (não bayes)

Calcular a acurácia do seu classificador com testes

```
from nltk.sentiment import SentimentIntensityAnalyzer
```

```
sia = SentimentIntensityAnalyzer()
```

```
sia.polarity_scores("Wow, NLTK is really powerful!")
```

```
{'neg': 0.0, 'neu': 0.295, 'pos': 0.705, 'compound': 0.8012}
```

Comparar os resultados (acurácia) do sia com os do seu classificador

Um notebook do jupyter postado em seu github (link na atividade do ava)