MINERAÇÃO DE TEXTOS (Parte 4) - ESCORE DE SENTIMENTOS

Livro: The Amateur Emigrant de Robert Louis Stevenson

Usando o pacote AFINN, podemos medir a intensidade das palavras positivas e negativas, ou seja, o quão positiva ou negativa elas são, numa avaliação que varia de -5 (para a mais negativa) até +5 (para a mais positiva).

Instalando o pacote 'AFINN' e importando as bibliotecas:

In [1]:
```
pip install afinn
```

```
Requirement already satisfied: afinn in c:\users\mmateus\anaconda3\lib\site-packages
(0.1)Note: you may need to restart the kernel to use updated packages.
```

In [2]:
```
import pandas as pd
import nltk
import re
import matplotlib.pyplot as plt
from afinn import Afinn

from nltk.corpus import stopwords
from nltk.corpus import opinion_lexicon
from nltk.tokenize import word_tokenize

nltk.download('stopwords', quiet=True)
nltk.download('opinion_lexicon', quiet=True)
nltk.download('punkt', quiet=True)

plt.style.use('ggplot')
```

Importando o texto:

In [3]:
```
df = pd.read_csv('data/TheAmateurEmigrant.txt',sep='\t')\
    .dropna()

df.head(10)
```

Out[3]:

	text
0	THE AMATEUR EMIGRANT
1	THE SECOND CABIN
2	I first encountered my fellow-passengers on th...
3	Thence we descended the Clyde in no familiar s...
4	on each other as on possible enemies. A few S...
5	already grown acquainted on the North Sea, wer...
6	their long pipes; but among English speakers d...
7	reigned supreme. The sun was soon overclouded...

	text
8	grew sharp as we continued to descend the wide...
9	falling temperature the gloom among the passen...

Preparando os dados:

In [4]:
```
# Adicionando linhas com números para dividir o texto em seções
df['line'] = range(1, len(df) + 1)

df.head()
```

Out[4]:

	text	line
0	THE AMATEUR EMIGRANT	1
1	THE SECOND CABIN	2
2	I first encountered my fellow-passengers on th...	3
3	Thence we descended the Clyde in no familiar s...	4
4	on each other as on possible enemies. A few S...	5

In [5]:
```
# Limpando, organizando e tokenizando o texto
def clean_text(text):
    text = text.lower()
    text = text.replace("'", '')
    text = re.sub(r'[^\w]', ' ', text)
    text = re.sub(r'\s+', ' ', text)
    text = text.strip()
    return text

df['text'] = df['text'].map(clean_text)
df['text'] = df['text'].map(word_tokenize)

df.head()
```

Out[5]:

	text	line
0	[the, amateur, emigrant]	1
1	[the, second, cabin]	2
2	[i, first, encountered, my, fellow, passengers...	3
3	[thence, we, descended, the, clyde, in, no, fa...	4
4	[on, each, other, as, on, possible, enemies, a...	5

In [6]:
```
df = df.explode('text').rename(columns={'text': 'token'})

df.head(10)
```

Out[6]:

	token	line
0	the	1
0	amateur	1

	token	line
0	emigrant	1
1	the	2
1	second	2
1	cabin	2
2	i	3
2	first	3
2	encountered	3
2	my	3

Machine Learning: Score de Sentimentos

- Usando o AFINN cuja escala de escore vai de -5 (muito negativo) para +5 (muito positivo)

In [7]:
```
afinn_scorer = Afinn()

df['score'] = df['token'].map(afinn_scorer.score).astype(int)
df = df[df['score'] != 0]
```

Obs: o uso do != 0 acima faz com que as palavras de escore 0 (neutro) sejam excluídas da análise.

- Tabela de Frequência do Score de Sentimentos

In [8]:
```
score_freq = df.score.value_counts().sort_index().to_frame('n')

score_freq
```

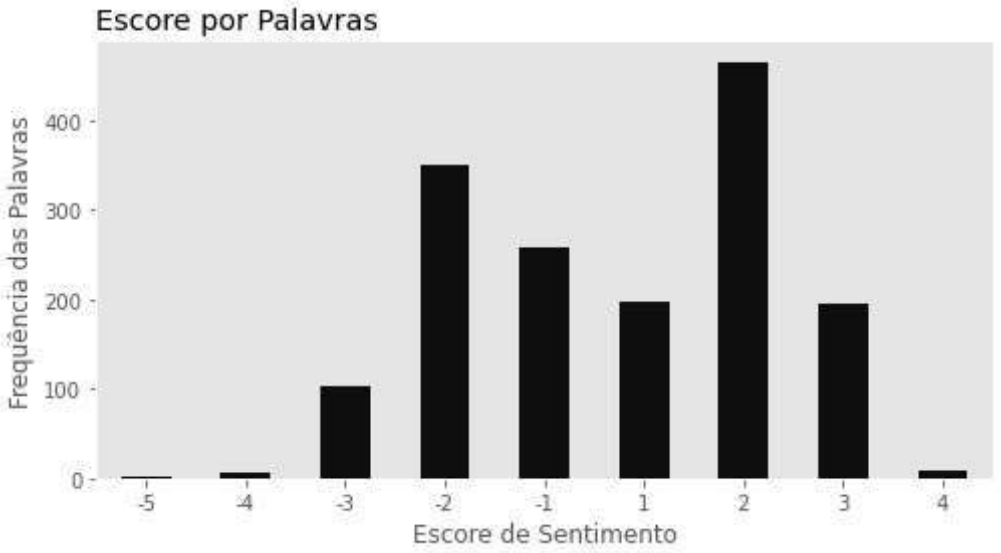
Out[8]:

	n
-5	1
-4	6
-3	103
-2	350
-1	258
1	198
2	465
3	194
4	9

Gráfico com a frequência do Score de Sentimentos

In [9]:
```
score_freq.plot.bar(
    legend=False,
```

```
    figsize=(8, 4),
    grid=False,
    color='darkblue')
plt.xlabel('Escore de Sentimento')
plt.ylabel('Frequência das Palavras')
plt.title('Escore de Sentimento por palavras', loc='left')
plt.title('Escore por Palavras', loc='left')
plt.xticks(rotation=0);
```



ARCO DO SENTIMENTO

- Divisão do texto em seções de 100 linhas e cálculo do escore de sentimento para cada seção.

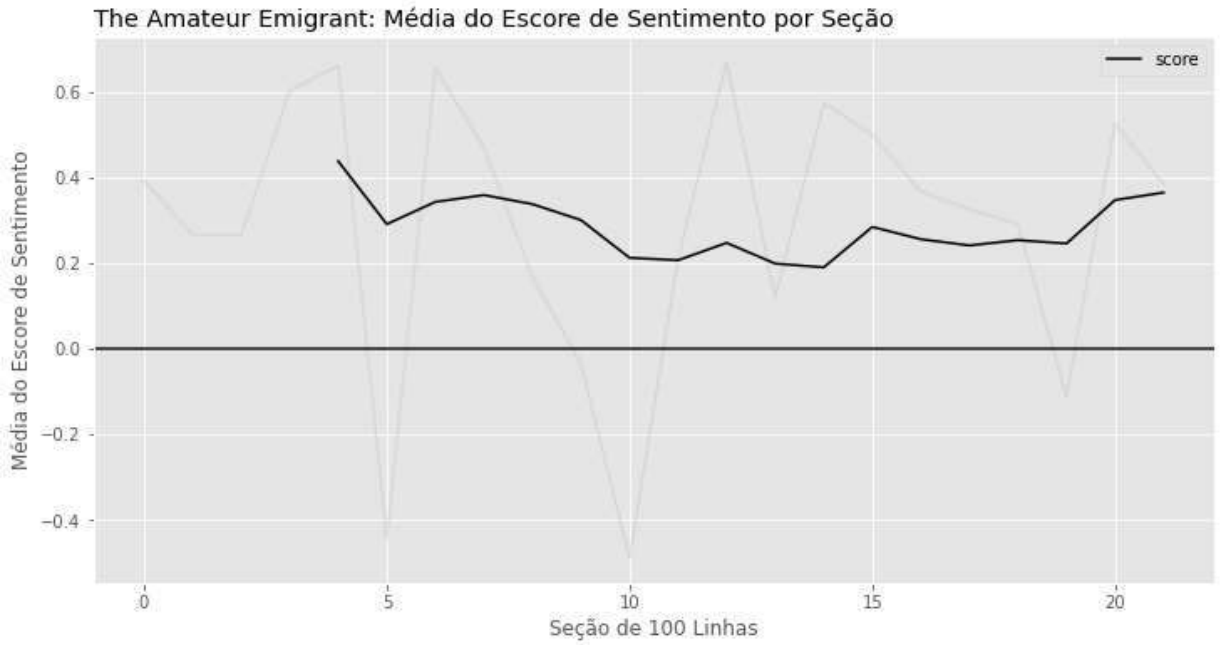
In [10]:
```
score_acc = df.groupby(df['line'] // 100)\
    .score.mean()\
    .to_frame('score')\
    .rename_axis('section')

score_acc.head(10)
```

Out[10]:

section	score
0	0.392157
1	0.266667
2	0.266667
3	0.602151
4	0.661017
5	-0.442857
6	0.654321
7	0.470588
8	0.168539
9	-0.034884

Gráfico do Escore por Seção para visualização do Arco da Narrativa

In [11]:
```
ax = score_acc.plot.line(legend=False, figsize=(12, 6), grid=False, alpha=0.5, color
score_acc.rolling(10, min_periods=5).mean().plot.line(ax=ax, color='black')
plt.xlabel('Seção de 100 Linhas')
plt.ylabel('Média do Escore de Sentimento')
plt.title('The Amateur Emigrant: Média do Escore de Sentimento por Seção', loc='left
plt.axhline(0, color='darkblue')
plt.xticks(rotation=0);
```



Usando o Escore de Sentimentos para criar um arco da narrativa da história de Stevenson, nota-se que apesar das passagens negativas dela, especialmente quando o autor relata as condições oferecidas aos viajantes da terceira classe do navio, ela tem uma conotação bem mais positiva do que negativa (arco situado acima da linha 0.0 do gráfico) do início ao fim.

Para sumarizar, o uso dos **algoritmos de mineração de textos** (Wordcloud, WordPairs, Classificação Binária e Escore de Sentimentos) na análise de "The Amateur Emigrant" foi fundamental para que a Cinetour Publishing publicasse essa história (junto com as demais do livro "Essays of Travel"). A editora usou a mesma estratégia para analisar e publicar "Travels with a Donkey", outra narrativa de viagem do mesmo autor.

Algoritmos de mineração de textos permitem que os profissionais de análise de conteúdo, como editores, possam ter uma ideia antecipada do material que têm em mãos, antes mesmo de dedicarem seu tempo a uma leitura completa e eventual publicação.