Processamento de Linguagem Natural

Aula 3: Análise de Conjuntos de Documentos

Objetivo: ao fim desta aula, o aluno será capaz de coletar e mostrar informações sobre a ocorrência de palavras ao longo de conjuntos de documentos

|  |
| --- |
| **Exercício 1** |
| *Objetivo: entender o problema do uso de palavras em textos* |
| Considere os dois textos abaixo, retirados de comentários sobre o mesmo produto postados em um site de compras:  TEXTO 1:  *Além de não ter a cor anunciada me enviou a câmera e um kit filme velho cheios de formiga, além de não se mostrar solicito para resolver a situação. Não aconselho a compra.*  TEXTO 2:  *Amei! era meu sonho ter uma, essa câmera é perfeita! já é meu xodó 3 super recomendo para os amantes de fotografia!.*  (a) Qual desses comentários é positivo? Qual é negativo?  (b) Quais palavras estão presentes em cada um dos comentários evidenciando que tratam-se de comentários positivos ou negativos?  (c) Qual seria um possível plano de ação para resolver os problemas que levaram o vendedor a ter esse comentário negativo? |

|  |
| --- |
| **Exercício 2** |
| *Objetivo: entender as limitações do TF para avaliar o conteúdo de coleções de texto* |
| É evidente que é inviável avaliar muitos comentários sobre produtos lendo todos um a um. Uma possibilidade para resumir o conteúdo de coleções de textos é usar a medida de Term Frequency (TF) para cada um dos documentos. Por exemplo:  *O produto não fez nem 10 dias e já ta com defeito. A cafeteira simplesmente parou a função de aquecimento, não funciona mais nada nela. É incrível o nível de descaso com o cliente.*  (a) Qual é a palavra com TF mais alto no trecho?  (b) Quais são as palavras mais relevantes deste trecho?  (c) Qual seria um possível plano de ação para esta marca satisfazer melhor seus clientes?  (d) Na variação a seguir, qual é a palavra com TF mais alto?  *A cafeteira não fez nem 10 dias e já ta com defeito. A cafeteira simplesmente parou a função de aquecimento, não funciona mais nada na cafeteira. É incrível o nível de descaso com os compradores de cafeteira.*  (e) O fato da palavra “cafeteira” ter sido mencionada muitas vezes modifica sua importância em relação a um possível plano de ação da empresa em relação à satisfação de seu cliente?  (f) O que deveria ter mais peso quanto a esse plano de ação: um cliente mencionar um aspecto (por exemplo: a garantia) dez vezes em seu comentário, ou dez clientes mencionarem a garantia, uma vez cada?  (g) Por que devemos assumir que um comentário é “negativo”: por que ele usou muitas vezes uma determinada palavra, ou por que determinadas palavras estão presentes no texto? |

|  |
| --- |
| **Exercício 3** |
| *Objetivo: entender como usar o Document Frequency (DF) para avaliar o conteúdo de coleções de texto* |
| Document Frequency (DF) de uma palavra é igual ao número de documentos em que ela aparece em toda a coleção.  (a) Complete a frase abaixo:  DF de uma palavra é a probabilidade de, ao selecionarmos aleatoriamente \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ do conjunto \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ , encontrarmos \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.  (b) Como a medida DF pode se relacionar ao ensaio estatístico de jogar uma moeda enviesada? |

|  |
| --- |
| **Exercício 4** |
| *Objetivo: usar a API do scikit-learn para encontrar palavras* |
| Considere o código abaixo:  from sklearn.feature\_extraction.text import CountVectorizer  s = ['meu primeiro documento é meu', 'meu segundo documento é meu também']  vectorizer = CountVectorizer()  x = vectorizer.fit\_transform(s)  print(x)  print(vectorizer.vocabulary\_)  Encontre na documentação do Scikit-Learn.  (a) O que o CountVectorizer faz?  (b) O que a variável x deve conter após a execução desse código?  (c) Como devemos interpretar o vectorizer.vocabulary\_ neste código?  (d) O que o parâmetro “binary=True” faria com nosso resultado?  (e) Como podemos informar as stopwords em nosso processo?  (f) Como o CountVectorizer separa palavras individuais?  (g) Como podemos usar essa estrutura para calcular o DF de cada palavra em toda a coleção? |

|  |
| --- |
| **Exercício 5** |
| *Objetivo: abrir um corpus de comentários* |
| Considere o código abaixo:  import pandas as pd  df = pd.read\_csv('./datasets/IMDB Dataset.csv')  df.head()  Ele abre o corpus IMDB, que contém comentários públicos sobre filmes.  (a) Quantas, e quais colunas, tem o corpus?  (b) Quantas entradas existem neste corpus?  (c) Quais são as 10 palavras com DF mais alto neste corpus?  (d) Quais são as 10 palavras com DF mais alto, somente considerando os comentários positivos?  (e) Quais são as 10 palavras com DF mais alto, somente considerando os comentários negativos? |

|  |
| --- |
| **Exercício 6** |
| *Objetivo: refletir sobre o uso de wordclouds e gráficos de barras para visualizar o conteúdo de textos* |
| Wordclouds são formas de visualizar frequências de palavras. Elas ficaram especialmente populares depois que o Flickr começou a usar tag-clouds para facilitar a navegação em suas coleções de fotos. A ideia é espalhar palavras pela tela e escrevê-las com um tamanho proporcional à sua frequência. A wordcloud para o conto “A Cartomante”, de Machado de Assis (gerada pelo site: <https://www.jasondavies.com/wordcloud/>) é a seguinte:    A partir da wordcloud, encontre:   1. Qual é a palavra mais frequente no conto 2. Qual é a terceira palavra mais frequente 3. Qual palavra é mais frequente: “cartomante” ou “rita”? 4. Responda novamente às perguntas (a) (b) e (c), mas desta vez tomando como base o gráfico de barras abaixo. 5. Em que situações a wordcloud é mais adequada que um gráfico de barras, e em que situações o gráfico de barras é mais adequado? |

|  |
| --- |
| **Exercício 7** |
| *Objetivo: entender o código para criação de um gráfico de barras em Python* |
| No código abaixo, encontre os seguintes movimentos:  (a) Colocar todos os reviews do corpus em uma lista de strings, com um elemento por review  (b) Calcular P(w), a razão entre o DF de cada palavra w e o total de documentos na coleção  (c) Ordenar palavras em ordem decrescente de P(w)  (d) Montar um gráfico de barras usando as probabilidades P(w) como alturas  (e) Associar as marcações no eixo horizontal às palavras escritas por extenso  reviews = [s for s in df['review']]  vectorizer = CountVectorizer(stop\_words = stopwords\_en, binary=True)  x = vectorizer.fit\_transform(reviews)  x\_ = np.mean(x, axis=0)  tuplas = [ (x\_[0,vectorizer.vocabulary\_[i]], i) for i in vectorizer.vocabulary\_.keys() ]  tuplas\_ordenadas = sorted(tuplas, reverse=True) # reverse=True pede uma ordenação em ordem decrescente  palavras = [ t[1] for t in tuplas\_ordenadas ]  contagens = [ t[0] for t in tuplas\_ordenadas ]  eixo\_x = np.arange(len(palavras))  n\_palavras = 60  plt.figure(figsize=(14,3))  plt.bar(eixo\_x[0:n\_palavras], contagens[0:n\_palavras])  plt.xticks(eixo\_x[0:n\_palavras], palavras[0:n\_palavras], rotation=90)  plt.ylabel('P(word)')  plt.xlabel('Words')  plt.title('Probability of finding a word in a random document from collection')  plt.savefig('pwords.png', bbox\_inches='tight')  plt.show() |

|  |
| --- |
| **Exercício 8** |
| *Objetivo: mostrar informações sobre o corpus de textos do IMDB usando gráficos de barras* |
| Crie figuras para responder às seguintes perguntas (pode ser mais de uma figura por pergunta, se for necessário):  (a) Os comentários positivos e negativos do corpus IMDB usam as mesmas palavras?  (b) Que termos técnicos do cinema (acionáveis pela equipe de produção) são mais mencionados nos comentários positivos e negativos?  (c) Em geral, a audiência presta mais atenção ao enredo, à atuação, ou à música do filme? |

|  |
| --- |
| **Mini-projeto 1** |
| *Objetivo: aplicar o conceito de DF para analisar diferentes corpus de texto* |
| Encontre outros datasets com reviews de clientes ou usuários. Alguns exemplos: Amazon Reviews Dataset ou Yelp Dataset.  Tipicamente, reviews positivos e negativos em diferentes domínios usam as mesmas palavras? Seria factível criar um dicionário amplo de palavras associadas a sentimentos positivos/negativos? |

|  |
| --- |
| **Mini-projeto 2** |
| *Objetivo: analisar o efeito do tamanho do corpus na análise de DF* |
| Estime um intervalo de confiança para o DF de cada palavra.  Como esse intervalo se comporta quando usamos datasets menores?  Quantos documentos, no mínimo, precisamos em nossos datasets de reviews para que DF seja uma estatística confiável? |

|  |
| --- |
| **Mini-projeto 3** |
| *Objetivo: criar e analisar um dataset para descobrir algum fato sobre a comunidade da qual você participa* |
| Pense em uma informação relevante sobre a comunidade que você está participando. Por exemplo, talvez você se pergunte: “o que posso fazer para tornar minha comunidade mais feliz?”, ou então: “se eu abrir uma barraca de lanches, quais seriam os lanches que devo fazer?”, ou ainda: “eu gostaria de dar aulas melhores!”. Precisamos, então, criar um corpus de textos nos quais cada sujeito escreva (ou fale?) sobre isso.  (a) Defina qual é a informação que você quer saber, e por que ela é relevante para você (é uma prospecção de *business model*? É pela curiosidade científica? Outro motivo?) – é muito importante não fazer uma pesquisa como essa apenas para confirmar algum viés de pensamento seu!  (b) Defina uma pergunta que é respondível por pessoas não-treinadas e que  (c) Teste sua pergunta com um(a) parceiro(a) de trabalho e verifique se a pergunta foi realmente entendida e se é respondível com base em fatos concretos levando a ações acionáveis. Se precisar, retorne ao passo (b).  (d) Defina quantas respostas você precisa, e também uma estratégia para coletar essas respostas.  (e) Teste toda a infra-estrutura da sua estratégia! Se precisar, re-defina a estratégia.  (f) Colete dados!  (g) Use as estratégias que aprendemos hoje para gerar a informação que você definiu no passo (a). |