UNIDADE DE APRENDIZAGEM: Modelagem de tópicos – latent dirichlet allocation

Apresentação

Existem diferentes formas de realizar pesquisas em conjuntos de dados, mas como encontrar uma informação específica em um grupo de livros ou revistas publicados ao longo de décadas? E se fosse preciso resumir os temas desses documentos com uma linha do tempo do conhecimento construído sobre determinado assunto? Com o objetivo de criar ferramentas e aplicações que utilizam a inteligência artificial para resolver essas questões, são utilizados os modelos de tópicos.

Entre os modelos de tópicos mais populares está o modelo de Alocação Latente de Dirichlet (*Latent Dirichlet Allocation* – LDA), que serviu como base para criar muitos outros modelos probabilísticos. O próprio LDA é baseado em um método estatístico chamado de Distribuição de Dirichlet, criado pelo matemático alemão Johann Peter Gustav Lejeune Dirichlet, nos anos 1800. O objetivo da criação do modelo LDA foi possibilitar o processamento eficiente de grandes coleções de dados para tarefas como: classificação, resumo e julgamentos de similaridade e relevância.

Nesta Unidade de Aprendizagem, você vai estudar sobre conceitos de modelos de tópicos, e conhecerá com mais detalhes o modelo de Alocação Latente de Dirichlet e as suas possíveis aplicações.

Bons estudos.

Ao final desta Unidade de Aprendizagem, você deve apresentar os seguintes aprendizados:

- Definir os conceitos e as aplicações da modelagem de tópicos.
- Analisar o algoritmo *Latent Dirichlet Allocation* (LDA).
- Desenvolver programas empregando LDA.

Desafio

Os algoritmos que utilizam modelagem de tópicos podem ser empregados para a identificação de temas em qualquer tipo de dados textuais, genéticos ou de imagens. Tanto em artigos ou em uma lista de mensagens, quando aplicados, possibilitam descobrir a estrutura temática dos dados. Geralmente, os tópicos do resultado do uso desse modelo são representados por uma lista de termos com maior probabilidade de relacionamento com o tópico.

Considere a situação a seguir:

A imagem a seguir possui audiodescrição. Para acessar o recurso, clique aqui

Nesse contexto, para atender à necessidade dessa cliente:

a) Escreva uma amostra de 10 documentos (frases) que poderiam fazer parte do contexto acima. Depois, liste esses documentos em uma instrução de formação de um *corpus* a ser utilizado em um modelo LDA.

b) Escreva uma instrução Python/Gensim para manter no resultado apenas as palavras que aparecem mais de duas vezes.

Infográfico

Os algoritmos LDA podem ser utilizados em diferentes fontes de dados, sejam textos, *links* ou imagens. Para compreender a sua funcionalidade e o funcionamento, é interessante acompanhar a sua aplicação em amostras de documentos de texto, e assim conhecer todas as etapas necessárias para o seu bom desempenho.

No Infográfico a seguir, você vai ver um exemplo de preparação de documentos e aplicação de um modelo LDA, a fim de verificar passo a passo o seu funcionamento.

A imagem a seguir possui audiodescrição. Para acessar o recurso, clique aqui

LIMPEZA DE DOCUMENTOS

para aplicação do LDA

O LDA é um modelo probabilístico generativo de um *corpus*. A ideia básica de sua aplicação é que os documentos sejam representados como combinações aleatórias sobre tópicos latentes, em que cada tópico é caracterizado por uma distribuição de palavras.
O objetivo dessa explicação é apresentar as etapas principais da preparação de dados para o uso de LDA.

Para iniciar, é necessário realizar a importação de uma **amostra de documentos**:

```
doc_a = "Brócolis é bom para comer. Meu irmão gosta de comer brócolis, mas minha mãe não."
doc_b = "Minha mãe passa muito tempo dirigindo para levar meu irmão para o treino de beisebol."
doc_c = "Alguns especialistas em saúde sugerem que dirigir pode causar aumento da tensão e pressão arterial."
doc_d = "Muitas vezes sinto pressão para ter um bom desempenho na escola, mas minha mãe não parece incentivar meu irmão a fazer melhor."
doc_e = "Profissionais de saúde dizem que brócolis é bom para sua saúde."

# compilar documentos de amostra em uma lista
doc_set = [doc_a, doc_b, doc_c, doc_d, doc_e]
```

Depois da importação dos documentos a serem analisados, é comum efetuar uma **limpeza dos documentos**, que envolve os seguintes processos:

- ***tokenizing*** (tokenização – converter um documento em seus elementos atômicos);
- ***stopping*** (parada – remover palavras sem sentido);
- ***stemming*** (mesclar palavras com significado equivalente).

Tokenizing:

```
from nltk.tokenize import RegexpTokenizer
tokenizer = RegexpTokenizer(r'\w+')
raw = doc_a.lower()
tokens = tokenizer.tokenize(raw)
>>> print(tokens) ['brócolis', 'é', 'bom', 'para', 'comer', 'meu', 'irmão', 'gosta', 'de', 'comer', 'brócolis', 'mas', 'minha', 'mãe', 'não']
```

Aqui, a primeira frase da amostra de documentos foi **separada em palavras ou "tokens"**.

Stopping:

```
from stop_words import get_stop_words

# criar uma lista de palavras de parada em português
stopwords = get_stop_words('portuguese')

# remover palavras de parada dos tokens
stopped_tokens = [i for i in tokens if not i in en_stop]
>>> print(stopped_tokens)
['brócolis', 'bom', 'comer', 'irmão', 'gosta', 'comer', 'brócolis', 'mãe']
```

Agora, foi criada uma lista de palavras de parada no dicionário em português, e **removidas essas palavras dos "tokens" criados na etapa anterior**.

Stemming:

```
from nltk.stem.porter import PorterStemmer

# Criar p_stemmer da classePorterStemmer
p_stemmer = PorterStemmer()

# stem token
texts = [p_stemmer.stem(i) for i in stopped_tokens]
>>> print(stemmed_tokens)
['brócolis', 'bom', 'comer', 'irmão', 'gosta', 'comer', 'brócolis', 'mãe']
```

Por fim, as palavras, ou *tokens*, com significado parecido são mescladas.

Agora, o conteúdo está pronto para ser empregado no modelo LDA.

Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Conteúdo do Livro

Com o aumento da quantidade de informações que são armazenadas atualmente, sejam de mídias sociais ou de *sites* de notícias, por exemplo, existe a necessidade de aprimorar as técnicas ou os modelos de busca de dados. Já existem diferentes formas de pesquisar na Internet, como, por exemplo, os *sites* com ferramentas de buscas, que por vezes são limitados para encontrar uma informação específica e dependem da seleção do usuário. Por esse motivo que foram criados e vêm sendo aprimorados os modelos de tópicos, que utilizam diferentes técnicas e métricas matemáticas para realizar pesquisas específicas em coleções de dados.

Você sabia que *corpus* é a maneira como são chamados os conjuntos de documentos na linguagem de coleções e também na modelagem de tópicos?

No capítulo Modelagem de tópicos – *Latent Dirichlet Allocation*, da obra *Processamento de Linguagem Natural*, você verá essa e outras definições de conjuntos de dados que são importantes para a modelagem de tópicos.

Boa leitura.

Os elementos gráficos deste capitulo possuem audiodescrição. Para acessar o recurso, clique aqui

PROCESSAMENTO DE LINGUAGEM NATURAL

Roni Francisco Pichetti

SAGAH
SOLUÇÕES EDUCACIONAIS INTEGRADAS

Modelagem de tópicos — *latent Dirichlet allocation*

Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Definir os conceitos e as aplicações da modelagem de tópicos.
- Analisar o algoritmo *latent Dirichlet allocation* (LDA).
- Desenvolver programas empregando o LDA.

Introdução

Os modelos de tópicos auxiliam na organização e na recuperação de dados em grupos de informações, que podem ser divididas em diferentes categorias, como palavras, documentos e coleções de documentos. Nesse sentido, tais modelos preocupam-se em resumir ou filtrar as informações de acordo com temas, a fim de auxiliar na categorização de informações, de imagem, de textos acadêmicos ou de diferentes *sites* da internet.

Neste capítulo, você estudará sobre os conceitos e as aplicações da modelagem de tópicos e conhecerá o modelo *latent Dirichlet allocation* (LDA), bem como instruções na linguagem de programação Phyton utilizadas para colocá-lo em prática.

1 Modelagem de tópicos: conceitos e aplicações

O conhecimento coletivo atual costuma ser digitalizado e armazenado, por exemplo, na forma de notícias, páginas da internet, artigos científicos, livros, imagens, som, vídeo ou mídias sociais, porém, com tal quantidade de informação armazenada, torna-se mais difícil encontrar o que realmente é necessário em uma busca, levando à necessidade de novas ferramentas computacionais para auxiliar a organizar, pesquisar e entender essa grande quantidade de informações. Geralmente, as informações *on-line* são requeridas por duas formas básicas: ferramentas de busca ou *hiperlinks* (BLEI, 2012).

Para realizar uma pesquisa, digita-se uma ou mais palavras-chave com o objetivo de encontrar um conjunto de documentos relacionados a elas. Então, são apresentados como resultado documentos que nem sempre têm relação direta com o que se deseja encontrar; é preciso analisar os documentos ou informações selecionadas para finalmente encontrar o resultado almejado. E, embora as ferramentas de busca *on-line* estejam cada vez mais poderosas, sempre há algo a melhorar.

Uma maneira de melhorar as pesquisas por informações digitais é procurar o seu tema, quando se torna possível tanto buscar informações mais específicas quanto mais amplas dentro de um tema, incluindo ainda o acesso ao histórico de como esses temas mudaram ao longo do tempo ou como estão conectados entre si. Assim, em vez de empregar somente palavras-chave, a pesquisa inicia com o tema de interesse e com a análise dos documentos relacionados a esse tema (BLEI, 2012).

Por exemplo, uma pesquisa sobre a história completa de uma revista brasileira, publicada por um período maior que 10 anos. Em um nível amplo, alguns temas podem corresponder a seções específicas da revista, como "alimentação" ou "exercícios físicos". O tema de interesse poderia ser ampliado ou especificado para "práticas para melhorar a saúde de pessoas com problemas cardíacos". Depois, verificando todas as publicações sobre o assunto, seria possível acompanhar como esse tema específico mudou ao longo do tempo, nos últimos 10 anos, e o resultado, com essa exploração, apontaria os artigos originais relevantes para o tema.

Porém, pesquisas em arquivos eletrônicos não são realizadas desse modo: mesmo com a grande quantidade de informações disponíveis na internet, o homem não consegue realizar esse tipo de pesquisa, leitura e estudo para obter o resultado descrito no exemplo. Por isso, pesquisadores de aprendizado de máquina desenvolveram a modelagem probabilística de tópicos, ou modelagem de tópicos, que se refere a um conjunto de algoritmos que visam a pesquisar sobre grandes arquivos de documentos com informações temáticas (BLEI, 2012). Na Figura 1, podemos observar a manipulação e a organização de dados por meio do aprendizado de máquina. Na primeira etapa, as informações, ou tópicos, não estão organizadas, e, na segunda, inicia-se uma organização que se completa na terceira fase.

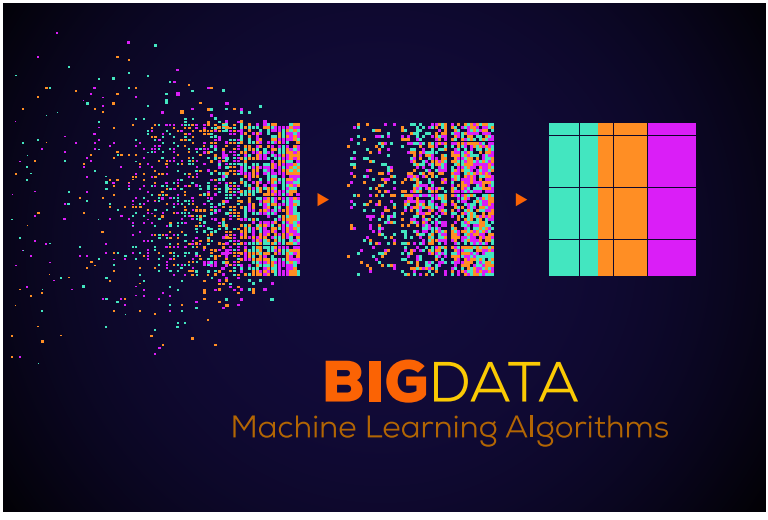

Figura 1. Algoritmos de aprendizado de máquina.
Fonte: SkillUp/Shutterstock.com.

Na modelagem de tópicos, cada documento é retratado como uma combinação de tópicos, em que cada tópico é considerado um conjunto de termos, ambos com probabilidades associadas. Assim, cada tópico extraído da coleção tem termos mais relevantes, ou seja, cada documento tem tópicos mais essenciais para a pesquisa, conforme as respectivas probabilidades (NOLASCO; OLIVEIRA, 2016).

Por exemplo, em uma coleção de documentos, pode-se esperar que termos como "bola" e "time" estejam mais associados e em maior quantidade ao tópico "esporte" do que ao tópico "gastronomia". Por sua vez, talvez "gastronomia" contenha mais termos como "receita" e "ingredientes", enquanto termos gerais como "minutos", "tempo" e "pessoas" teriam chance igual de aparecer em ambos os tópicos. Nesse contexto, cada tópico extraído pode ser representado inicialmente por seus termos mais comuns e, depois, os documentos são agrupados conforme o respectivo tópico.

Os algoritmos de modelagem de tópicos são métodos estatísticos que analisam as palavras dos textos originais para descobrir os temas que os percorrem, como esses temas estão conectados entre si e como eles mudam ao longo do tempo, sem exigir leitura prévia dos documentos, já que os tópicos emergem da análise dos textos originais. Essa modelagem permite descobrir, organizar, agrupar e extrair documentos eletrônicos em uma escala muito maior do que a capacidade humana de leitura e análise de resultados (BLEI, 2012).

Nesse sentido, os algoritmos de modelagem de tópicos são utilizados para descobrir os principais temas que permeiam uma coleção grande e não estruturada de documentos, tendo a função, portanto, de organizar a coleção de acordo com os temas descobertos. Eles podem ser aplicados a coleções maciças de documentos e adaptados a vários tipos de dados, como para encontrar padrões em dados genéticos, documentos acadêmicos e técnicos, imagens, notícias e em mídias sociais (BLEI, 2012).

Depois da aplicação da modelagem de tópicos, geralmente o resultado consiste em um conjunto de termos que indicam os temas de uma coleção. Porém, a apresentação de um conjunto de termos de maneira externa ao seu contexto original pode dificultar a compreensão, tornando necessária uma maior interpretação semântica (avaliação do significado) dos tópicos para identificar o tema de modo mais eficiente. Essa interpretação é colocada em prática por meio da rotulagem de tópicos, pela qual se pode definir cada tópico como um conjunto mais explicativo de termos (NOLASCO; OLIVEIRA, 2016).

A rotulagem de tópicos permite mostrar aos usuários os tópicos com significado mais coerente em relação à pesquisa efetuada, o que diminui a dependência de conhecimentos especializados sobre a coleção ou o domínio, imprescindíveis para interpretar esses tópicos (NOLASCO; OLIVEIRA, 2016). Cabe deixar claro que o resultado da modelagem em tópicos não garante uma exclusividade de tópicos corretos de cada documento, porém existem grandes chances de os documentos com ideias equivalentes de tópicos ficarem agrupados juntos.

2 Algoritmo *latent Dirichlet allocation*

De acordo com Nolasco e Oliveira (2016), o modelo de alocação latente de Dirichlet [*latent Dirichlet allocation* (LDA)] é um dos modelos de tópicos mais populares, tendo servido como base para criar muitos outros modelos probabilísticos. Seu nome se refere ao matemático alemão Johann Peter Gustav Lejeune Dirichlet (1805-1858), por conta do uso de seu método estatístico

chamado de distribuição de Dirichlet. O matemático exerceu forte influência sobre o desenvolvimento dessa ciência ao longo de vida científica e acadêmica, tendo entre seus muitos alunos Eisenstein (1823-1852) e Kronecher (1823-1891) (BEGEHR *et al*., 2012).

Com base no método estatístico de distribuição de Dirichlet, os cientistas Blei, NG e Jordan propuseram, em 2003, o algoritmo de LDA para modelagem de tópicos. O objetivo do estudo foi permitir o processamento eficiente de grandes coleções, preservando os relacionamentos estatísticos essenciais úteis para tarefas básicas, como classificação, detecção de novidades, resumo e julgamentos de similaridade e relevância (BLEI; NG; JORDAN, 2003).

Para Blei, Ng e Jordan (2003), é importante conhecer o significado de alguns termos utilizados de maneira diferente da usual na linguagem de coleções para compreender a modelagem de tópicos: palavra ou termo, a unidade básica de dados, definida como um item de um vocabulário indexado por {1..., V}; documento, uma sequência de N palavras denotadas por $w = \{w_1, w_2, ..., w_N\}$, onde w_N é a última palavra da sequência; *corpus* (plural *corpora*), um conjunto de M documentos denotados por $w = \{w_1, w_2, ..., w_M\}$.

Nesse contexto, o LDA representa um modelo probabilístico generativo de um *corpus*, cuja ideia básica é de que os documentos sejam representados como combinações aleatórias sobre tópicos latentes, em que cada tópico se caracteriza por uma distribuição de palavras (BLEI; NG; JORDAN, 2003).

Para Nolasco e Oliveira (2016), as pesquisas de modelagem de tópicos em documentos de textos iniciaram com o desenvolvimento da técnica chamada análise de semântica latente [*latent semantic analysis* (LSA)], na qual se utiliza a álgebra linear para decompor um *corpus* nos temas que o constituem, por meio de uma matriz de contagem de frequência dos termos. Uma evolução do LSA com uso de fórmulas probabilísticas surgiu posteriormente, a denominada indexação probabilística de semântica latente [*probabilistic latent semantic indexig* (pLSI)]. Pode-se dizer que o modelo LDA foi baseado tanto no LSA quanto no pLSI.

O LDA e outros modelos de tópicos integram o campo de pesquisa de modelagem probabilística, na qual os dados são tratados como originados de um processo generativo com variáveis ocultas, que, por sua vez, define uma distribuição da probabilidade conjunta utilizada para computar a distribuição de variáveis ocultas durante uma observação de variáveis. As variáveis observadas nesse processo são as palavras dos documentos e as variáveis ocultas referem-se à estrutura de tópicos (BLEI, 2012).

A premissa do modelo LDA reside no fato de que os documentos exibem vários tópicos. Por exemplo, considere o artigo retratado na Figura 2: intitulado "*Seeking Life's Bare (Genetic) Necessities*" ["Procurando as necessidades básicas (genéticas) da vida"], trata do uso da análise de dados para determinar o número dos genes que um organismo precisa para sobreviver. Estão destacadas diferentes palavras utilizadas no artigo.

Palavras sobre análise de dados, como "computador" e "previsão" (*computer* e *predictions*), são destacadas em azul; aquelas sobre biologia revolucionária, como "vida" e "organismo" (*life* e *organisms*), em rosa; e aquelas sobre genética, como "sequenciados" e "genomas" (*sequenced* e *genomes*), em amarelo. Destacando a repetição das palavras do artigo, é possível verificar que ele combina genética, análise de dados e biologia evolutiva em diferentes proporções. Além disso, saber que esse artigo combina esses tópicos auxilia a situá-lo em uma coleção de artigos científicos (BLEI, 2012).

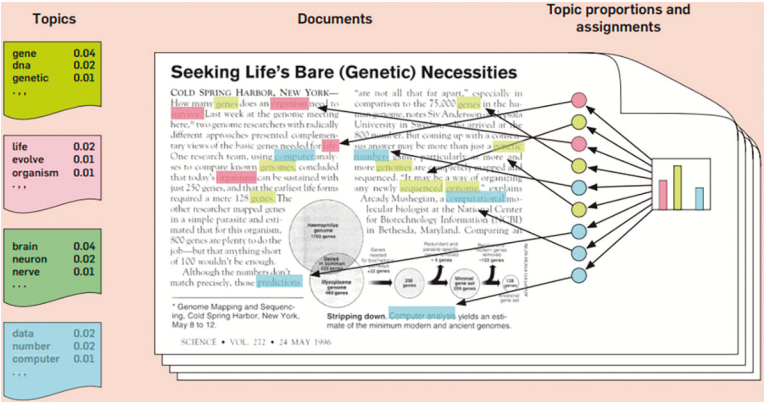

Figura 2. Suposições do LDA.

Fonte: Adaptada de Blei (2012).

É necessário definir formalmente um tópico para ser distribuído em um vocabulário fixo. Por exemplo, o tópico de "genética" apresenta palavras sobre genética com alta probabilidade e o de biologia evolutiva tem palavras sobre biologia evolutiva com alta probabilidade. Assume-se que esses tópicos sejam especificados antes da geração de dados. Cada documento exibe os tópicos em proporções diferentes (etapa 1) e cada palavra em cada documento é extraída

de um dos tópicos (etapa 2), em que o tópico selecionado é escolhido na distribuição por documento sobre os tópicos (etapa 3) (BLEI, 2012).

No artigo da Figura 2, a distribuição por tópicos coloca: probabilidade em genética (*gene*, *DNA*, *genetic*), que está em amarelo; análise de dados (*data*, *number*, *computer*), em azul; e biologia evolutiva (*life*, *evolve*, *organism*), em rosa — e cada palavra é extraída de um desses três tópicos. O próximo artigo da coleção será sobre análise de dados e neurociência, cuja distribuição sobre tópicos colocaria probabilidade nesses dois tópicos. Essa é a característica que distingue o modelo LDA dos demais, visto que todos os documentos da coleção compartilham o mesmo conjunto de tópicos, mas cada documento exibe esses tópicos em proporções diferentes (BLEI, 2012).

Portanto, o uso do LDA supõe que documentos com tópicos similares usam grupos similares de palavras e que tópicos podem ser encontrados ao procurar grupos de palavras que ocorrem juntas frequentemente nos documentos da coleção de documentos.

O LDA é definido pelas suposições estatísticas que faz sobre o *corpus*. Uma área ativa da pesquisa de modelagem de tópicos é como estender essas suposições para descobrir uma estrutura mais sofisticada nos textos. Segundo Blei (2012), o modelo LDA se baseia nas suposições a seguir.

- A ordem das palavras no documento não importa: embora essa suposição não seja realista, mostra-se razoável que o único objetivo consista em descobrir a estrutura semântica do curso dos textos. Para objetivos mais sofisticados, como a geração de idiomas, não é apropriado.
- A ordem dos documentos não importa: essa suposição pode não ser realista ao analisar coleções de longa duração que se estendem por anos ou séculos, nas quais é possível assumir que os tópicos mudam com o tempo.
- O número de tópicos é assumido, conhecido e corrigido: o número de tópicos é determinado pela coleção durante uma inferência posterior e novos documentos podem exibir tópicos anteriormente não vistos.

Assim, os modelos de tópicos probabilísticos, como o LDA, podem ser descritos como um conjunto de algoritmos que fornecem uma solução estatística para o problema de gerenciar grandes arquivos de documentos. Com os recentes avanços científicos no suporte ao aprendizado de máquina não supervisionado, os modelos de tópicos prometem constituir um componente importante para resumir e entender a crescente quantidade de informações digitais armazenadas.

3 Algoritmos com LDA

Os algoritmos que utilizam modelagem de tópicos podem ser empregados para identificar temas em qualquer tipo de dados textuais, genéticos ou de imagens. Tanto em artigos quanto em uma lista de mensagens, quando aplicados, possibilitam descobrir a estrutura temática dos dados. Geralmente, os tópicos do resultado do uso desse modelo são representados por uma lista de termos com maior probabilidade de relacionamento com o tópico. E os termos mais comuns presentes na base de dados analisada são diluídos pelas probabilidades, pois não têm relação direta com nenhum tema (NOLASCO; OLIVEIRA, 2016).

Diferentes linguagens de programação disponibilizam bibliotecas para implementação de modelagem de tópicos, como a "lda-c" para a linguagem C, a "mallet" para Java e a "gensim" para Python. Em virtude de seu caráter probabilístico, não se torna necessário realizar nenhum pré-processamento no *corpus* para a utilização de uma dessas linguagens, já que o procedimento passa por uma tokenização do conteúdo textual. Assim, as palavras comuns são irrelevantes para o resultado final e os demais processamentos podem ser utilizados para melhorar o desempenho (NOLASCO; OLIVEIRA, 2016).

Analisar o conteúdo de mídias sociais, mensagens de texto, e-mails ou qualquer outra fonte em que o conteúdo do texto seja informal e sujeito a grandes variações representa um desafio para a modelagem de tópicos, visto que ela necessita de determinada consistência textual dos dados. De qualquer maneira, todos esses meios de comunicação desenvolveram formas consistentes de transmitir mensagens, com *hashtag* ou vocabulário específico que se transforma em um novo padrão, tornando possível analisar esses dados (NOLASCO; OLIVEIRA, 2016).

A seguir, será apresentado um exemplo de utilização do modelo LDA na linguagem Phyton, disponível na documentação da biblioteca "gensim", o qual está dividido em 10 partes descritas sequencialmente intercaladas com suas respectivas interpretações.

Exemplo

1 — Exemplo de documento:

```
document = "Human machine interface for lab abc computer
applications"
```

2 — Exemplo de *corpus*:

```
text_corpus = [
  "Human machine interface for lab abc computer applications",
  "A survey of user opinion of computer system response
time",
  "The EPS user interface management system",
  "System and human system engineering testing of EPS",
  "Relation of user perceived response time to error
measurement",
  "The generation of random binary unordered trees",
  "The intersection graph of paths in trees",
  "Graph minors IV Widths of trees and well quasi ordering",
  "Graph minors A survey",
]
```

Fonte: Gensim (2020).

Para facilitar o entendimento do exemplo, é necessário ter em mente o conceito de alguns termos utilizados: documento corresponde a algum texto; *corpus* é uma coleção de documentos; vetor é uma representação matemática de um documento; e modelo representa um algoritmo utilizado para transformar vetores de uma representação para outra. Na parte 1 do exemplo, há um tipo de documento, visto que este pode ser, por exemplo, uma frase, um parágrafo ou uma notícia. Já a parte 2 demonstra um *corpus*, uma coleção de frases (GENSIM, 2020).

Exemplo

3 — Etapa de pré-processamento:

```
# Crie um conjunto de palavras frequentes
stoplist = set('for a of the and to in'.split(' '))
# Coloque em minúscula cada documento, divida-o por espaço
em branco e filtre as palavras-chave
texts = [[word for word in document.lower().split() if
word not in stoplist]
         for document in text_corpus]

# Conte frequências de palavras
from collections import defaultdict
frequency = defaultdict(int)
for text in texts:
    for token in text:
        frequency[token] += 1

# Mantenha apenas as palavras que aparecerem mais de uma vez
processed_corpus = [[token for token in text if
frequency[token] > 1] for text in texts]
pprint.pprint(processed_corpus)
```

Resultado:

```
[['human', 'interface', 'computer'],
 ['survey', 'user', 'computer', 'system', 'response',
'time'],
 ['eps', 'user', 'interface', 'system'],
 ['system', 'human', 'system', 'eps'],
 ['user', 'response', 'time'],
 ['trees'],
 ['graph', 'trees'],
 ['graph', 'minors', 'trees'],
 ['graph', 'minors', 'survey']]
```

4 — Criação de código exclusivo (ID) para cada palavra do *corpus*:

```
from gensim import corpora
dictionary = corpora.Dictionary(processed_corpus)
print(dictionary)
```

Resultado:

```
Dictionary(12 unique tokens: ['computer', 'human', 'in-
terface', 'response', 'survey']...)
```

5 — O *corpus* processado do exemplo 2 tem 12 palavras únicas; para verificar o ID de cada palavra, utiliza-se:

```
pprint.pprint(dictionary.token2id)
```

Fonte: Gensim (2020).

A parte 3 do exemplo corresponde ao pré-processamento apresentado de maneira simplificada, no qual são removidas apenas algumas palavras comuns utilizadas com mais frequência ou as que aparecem apenas uma vez no *corpus*. Esse processo corresponde à tokenização dos dados, pois divide os documentos em palavras. Na etapa seguinte, a parte 4, atribui-se um código identificador único para cada palavra, o que é útil para um *corpus* com uma grande quantidade de palavras. Para verificar o código de cada palavra, pode ser utilizada a instrução da etapa 5 (GENSIM, 2020).

Exemplo

6 — O *corpus* da parte 2 também pode ser transformado em vetor, para o qual se emprega:

```
bow_corpus = [dictionary.doc2bow(text) for text in
processed_corpus]
pprint.pprint(bow_corpus)
```

Resultado:

```
[[(0, 1), (1, 1), (2, 1)],
 [(0, 1), (3, 1), (4, 1), (5, 1), (6, 1), (7, 1)],
 [(2, 1), (5, 1), (7, 1), (8, 1)],
 [(1, 1), (5, 2), (8, 1)],
 [(3, 1), (6, 1), (7, 1)],
 [(9, 1)],
 [(9, 1), (10, 1)],
 [(9, 1), (10, 1), (11, 1)],
 [(4, 1), (10, 1), (11, 1)]]
```

Fonte: Gensim (2020).

Na parte 6, realiza-se a vetorização dos documentos, para que eles possam ser manipulados matematicamente, correspondendo a uma das maneiras de vetorizar o documento. A vetorização pode ser aplicada utilizando perguntas e respostas, como verificar quantas vezes cada palavra aparece no *corpus* ou quantos parágrafos de que dispõe o documento, cuja resposta é representada pelo código gerado previamente. Assim, a sequência de respostas torna-se o vetor do documento (GENSIM, 2020).

Exemplo

7 — Com o *corpus* vetorizado, aplica-se o modelo:

```
from gensim import models

# treinar o modelo
tfidf = models.TfidfModel(bow_corpus)

# transformar a cadeia "menores de sistema ou system minors"
words = "system minors".lower().split()
print(tfidf[dictionary.doc2bow(words)])
```

Resultado:

```
[(5, 0.5898341626740045), (11, 0.8075244024440723)]
```

Fonte: Gensim (2020).

A parte 7 do exemplo apresenta um modelo simples: a aplicação de um modelo acompanha a transformação de uma representação do documento para outra, e o modelo aprende os detalhes da transformação do documento em um vetor de treinamento, realizado na leitura de um *corpus* específico. O modelo apresentado é o tf-idf, que transforma o vetor da etapa anterior em um espaço vetorial em que a contagem de frequência de palavras é ponderada de acordo com a raridade de cada uma no *corpus*. O resultado do modelo é uma lista de tuplas — a primeira é o código (ID) do *token*, e a segunda a ponderação tf-idf (GENSIM, 2020).

Exemplo

8 — Preparar o *corpus* para a consulta de similaridade:

```
from gensim import similarities
index = similarities.SparseMatrixSimilarity(tfidf[bow_cor-
pus], num_features=12)
```

9 — Consultar a similaridade de todos os documentos do *corpus*:

```
query_document = 'system engineering'.split()
query_bow = dictionary.doc2bow(query_document)
sims = index[tfidf[query_bow]]
print(list(enumerate(sims)))
```

Resultado:

```
[(0, 0.0), (1, 0.32448703), (2, 0.41707572), (3, 0.7184812),
(4, 0.0), (5, 0.0), (6, 0.0), (7, 0.0), (8, 0.0)]
```

10 — Organizar os resultados da similaridade:

```
for document_number, score in sorted(enumerate(sims),
key=lambda x: x[1], reverse=True):
    print(document_number, score)
```

Resultado:

```
3 0.7184812
2 0.41707572
1 0.32448703
0 0.0
4 0.0
5 0.0
6 0.0
7 0.0
8 0.0
```

Fonte: Gensim (2020).

Na oitava parte do exemplo, é apresentada a instrução para preparar o *corpus* para a consulta de similaridade. Depois, nas etapas 9 e 10, realiza-se a consulta de similaridade e organizam-se os resultados obtidos (para facilitar a sua interpretação), respectivamente. Pode-se dizer que, na saída, o documento 3 tem uma pontuação de similaridade de 0,718, o que é igual a 78%, enquanto o documento 2 apresentou uma pontuação de 42%, e assim por diante (GENSIM, 2020).

Link

Ao longo deste capítulo, você pôde observar diferentes conceitos sobre a modelagem de tópicos e LDA. Acessando o *link* a seguir, você conseguirá conhecer mais conceitos e exemplos sobre o uso do modelo LDA.

https://qrgo.page.link/cjT8j

Referências

BEGEHR, H. *et al. Mathematics in Berlin*. Berlin: Birkhäuser, 2012.

BLEI, D. M. Surveying a suite of algorithms that offer a solution to managing large document archives. *Communications of the ACM*, [*s. l.*], v. 55, n. 4, 2012.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. *Journal of Machine Learning Research*, [*s. l.*], v. 3, p. 993–1022, 2003.

GENSIM. *Core concepts*. 2020. Disponível em: https://radimrehurek.com/gensim/auto_examples/core/run_core_concepts.html#sphx-glr-auto-examples-core-run-core-concepts-py. Acesso em: 06 mar. 2020.

NOLASCO, D.; OLIVEIRA, J. Modelagem de tópicos e criação de rótulos: identificando temas em dados semi-estruturados e não-estruturados. *In*: OGASAWARA, E.; VIEIRA, V. (org.). *Tópicos em gerenciamento de dados e informações*. Salvador: Sociedade Brasileira de Computação, 2016. cap. 4. p. 87–112. Disponível em: http://sbbd2016.fpc.ufba.br/e-book/minicursos.pdf. Acesso em: 07 mar. 2020.

Fique atento

Os *links* para *sites* da Web fornecidos neste capítulo foram todos testados, e seu funcionamento foi comprovado no momento da publicação do material. No entanto, a rede é extremamente dinâmica; suas páginas estão constantemente mudando de local e conteúdo. Assim, os editores declaram não ter qualquer responsabilidade sobre qualidade, precisão ou integralidade das informações referidas em tais *links*.

Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.

Conteúdo:

SAGAH
SOLUÇÕES EDUCACIONAIS INTEGRADAS

Dica do Professor

Na modelagem de tópicos, cada documento é considerado uma combinação de tópicos, em que cada tópico é um conjunto de termos,
os dois com probabilidades associadas. Assim, cada tópico que for extraído da coleção contém termos mais relevantes.

Nesta Dica do Professor, você verificará alguns conceitos adicionais sobre o modelo de tópicos LDA e um conceito prático de sua aplicação.

As imagens do vídeo a seguir possuem audiodescrição. Para acessar o recurso, clique aqui

Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Exercícios

1) **O modelo de tópicos auxilia na realização de buscas em um grande volume de dados. Nesse sentido, pode-se dizer que a forma de pesquisar utilizada na modelagem de tópicos, mais eficiente se comparada a ferramentas de busca da Internet, é feita por meio de:**

A) Conceitos-chave.

B) Palavras-chave.

C) Temas de interesse.

D) Documentos.

E) Histórico.

2) **A modelagem de tópicos é um conjunto de algoritmos que visam a pesquisar sobre grandes arquivos de documentos com informações temáticas. Assinale a alternativa em que uma das características da modelagem de tópicos é descrita de forma correta:**

A) Os algoritmos de modelagem de tópicos exigem leitura prévia dos documentos analisados.

B) Os algoritmos de modelagem de tópicos impõem que os temas estarão conectados entre si.

C) Os algoritmos de modelagem de tópicos verificam como os temas mudam ao longo do tempo.

D) Os algoritmos de modelagem de tópicos são métodos genéticos que avaliam palavras e temas.

E) Os algoritmos de modelagem de tópicos são utilizados para descobrir temas secundários.

3) **A rotulagem de tópicos permite mostrar aos usuários os tópicos com significado mais coerente em relação à pesquisa efetuada. Nesse sentido, assinale a alternativa que apresenta corretamente uma das finalidades do uso da rotulagem de tópicos:**

A) Interpretar melhor o significado dos tópicos resultantes da pesquisa.

B) Definir cada tema como um conjunto mais explicativo de tópicos.

C) Aumentar a dependência de conhecimentos especializados sobre a coleção.

D) Apresentar um conjunto de termos de maneira externa ao seu contexto original.

E) Mostrar os tópicos com significado complexo em relação à pesquisa efetuada.

4) **O modelo LDA é um dos modelos de tópicos mais populares, o qual serviu como base para criar muitos outros modelos probabilísticos. Nesse sentido, assinale a alternativa na qual são listados corretamente a(s) técnica(s) e/ou o modelo(s) nos quais o LDA foi baseado:**

A) Alocação Latente de Dirichlet, Análise de Semântica Latente e Indexação Probabilística de Semântica.

B) Análise de Dirichlet, Análise de Semântica Latente e Indexação Semântica de Probabilística Latente.

C) Distribuição Latente, Análise de Semântica de Dirichlet e Indexação Probabilística de Semântica Latente.

D) Distribuição de Dirichlet, Análise de Semântica Latente e Indexação Probabilística de Semântica Latente.

E) Distribuição de Dirichlet, Alocação Latente de Dirichlet e Indexação Probabilística de Dirichlet Latente.

5) **Diferentes linguagens de programação disponibilizam bibliotecas para a implementação de modelagem de tópicos. Entre elas estão a "lda-c" para a linguagem C, "mallet" para Java e "gensim" para Python. Nesse sentido, verifique a instrução Phyton a seguir:**

processed_corpus = [[token for token in text if frequency[token] > 1] for text in texts]

Agora, assinale a alternativa que descreve corretamente a função da expressão acima:

A) Contar as vezes que os termos se repetem no *corpus*.

B) Manter no resultado os termos que se repetem no *corpus*.

C) Criar um conjunto de termos frequentes no documento.

D) Definir um código exclusivo para cada palavra do documento.

E) Verificar o código de cada palavra dos documentos.

Na prática

A modelagem de tópicos pode auxiliar empresas ou pesquisadores em diferentes assuntos, seja para fins comerciais ou acadêmicos. Sua utilização resulta em resultados mais abrangentes sobre os assuntos tratados nos documentos que fazem parte da coleção de dados.

Na Prática, você vai conhecer o caso de uma *startup* que está buscando aumentar o número de usuários e de visibilidade de sua mídia social.

A imagem a seguir possui audiodescrição. Para acessar o recurso, clique aqui

GERENCIANDO UMA MÍDIA SOCIAL:
Joinha

O crescente número de acessos a diferentes mídias sociais incentivou um grupo de estudantes de um curso de Inteligência Artificial a criar uma *startup*, a IA5. **O principal produto oferecido pela IA5 é uma mídia social chamada Joinha**, pela qual seus usuários podem elaborar e compartilhar textos, fotos e mensagens com os demais usuários.

A fim de realizar uma análise sobre os assuntos que vêm sendo mais citados na mídia social, para verificar o perfil de interesse dos usuários do Joinha, **a equipe da IA5 optou pela utilização da modelagem de tópicos. O modelo utilizado foi o LDA**.

Visto que consta no termo de uso do Joinha a possível análise de dados compartilhados em modo público, com essas informações, ao longo do último ano, foi criada a amostra de documentos.

Cada postagem contém termos específicos, os quais formam um único documento. A coleção dos documentos criou o *corpus* a ser utilizado.

Após a coleta dos dados, ocorreu a importação da amostra de documentos, definindo um nome específico para cada um.

Depois, foi realizada uma limpeza prévia dos documentos, com os procedimentos de:

- tokenização (separar as postagens em *tokens*);
- remoção de caracteres não desejados (remover menções a outros usuários, URLs, emoticons, caracteres especiais);
- verificação de palavras de parada (palavras descartadas);
- *stemming* (reduzir palavras topicamente semelhantes à sua raiz).

Então, foram criados vetores com os tópicos dos documentos e o dicionário do *corpus*:

```
from gensim import corpora, models
dictionary = corpora.Dictionary(texts)
corpus = [dictionary.doc2bow(text) for text in texts]
```

E, por fim, aplicado o modelo LDA:

```
ldamodel =
gensim.models.ldamodel.LdaModel(corpus, num_topics=5, id2word = dictionary,
passes=10)
```

Esse modelo solicita a geração de, no máximo, cinco tópicos, verifica o dicionário criado na etapa anterior e define como 10 o número de voltas que o modelo passará pelo *corpus*.

Quanto maior o número de passes, mais preciso será o modelo. Muitas passagens podem ser lentas em um corpus muito grande.

Assim, foi possível criar uma linha do tempo dos **assuntos mais comentados no Joinha** em cada mês do ano analisado, bem como foi possível **relacionar o resultado alcançado com acontecimentos e notícias** mais divulgadas no país em cada período.

Com o resultado, a IA5 tem uma noção sobre quais assuntos podem ter mais visibilidade e que geraram maior engajamento de seus usuários.

Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Saiba mais

Para ampliar o seu conhecimento a respeito desse assunto, veja abaixo as sugestões do professor:

Visualização em multirresolução do fluxo de tópicos em coleções de texto

Para conhecer mais sobre a utilização e as limitações do modelo LDA, acesse a dissertação de Bruno Schneider, defendida na Fundação Getúlio Vargas, no curso de Mestrado em Modelagem Matemática da Informação. Dê atenção especial para os conceitos presentes entre as páginas 05 e 09 do documento.

Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

LSI e LDA

Para verificar mais detalhes e conceitos sobre os modelos *Latent Semantic Indexig* (LSI) e *Latent Dirichlet Allocation* (LDA), consulte o artigo a seguir, de Marcos de Souza e Renato Rocha Souza.

Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

Modelos probabilísticos de tópicos: desvendando o *Latent Dirichlet Allocation*

Para compreender a fundo os detalhes técnicos do uso do modelo LDA e de suas equações, acesse o relatório técnico criado por Thiago de Paulo Faleiros e Alneu de Andrade Lopes, da Universidade de São Paulo. Dê atenção especial ao Capítulo 2 do documento, entre as páginas 13 e 18.

Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.