Prof. Dr. Andreas Hotho,
M.Sc. Janna Omeliyanenko
Lecture Chair X for Data Science, Universität Würzburg

11. Exercise for "Sprachverarbeitung und Text Mining"

04.02.2022

1 Knowledge Questions

1. In the lecture, the Dirichlet Distribution was introduced. Describe the role of the Dirichlet parameter $\overrightarrow{\alpha}$ in your own words. What are the effects of changing these parameter?

   - Entries in vector $\overrightarrow{\alpha}$ can be seen as pseudo counts that represent how many times each outcome has been observed.
   - The Dirichlet distribution, as a prior distribution, then generates probability distributions from which the pseudo counts could originate.
   - The higher the pseudo counts of a random variable x_1, the more often the Dirichlet distribution generates a probability distribution with a high pseudo count of x_1.
     $\rightarrow$ The Dirichlet distribution randomly draws probability distributions, favoring the distributions that better explain the pseudo counts $\overrightarrow{\alpha}$.

2. What are the input and output of the Gibbs Sampling Algorithm?

   - Input: corpus D, number of topics K, Dirichlet parameters $\overrightarrow{\alpha}$ and $\overrightarrow{\beta}$
   - Output: Learned distributions θ_d and ϕ_k

3. How can LDA be used to cluster words in a corpus? What do the resulting clusters contain?

- Given Dirichlet parameters $\overrightarrow{\alpha}$, $\overrightarrow{\beta}$, number of topics K and corpus D.
- Apply Gibbs sampling until convergence.
- Compute ϕ_k from the resulting topic-term frequencies n_k^t.
  $\rightarrow \phi_k$ contains a probabilistic clustering, which says how often each term occurs in topic/cluster k.
- Resulting clusters contain terms/words that are assigned to semantically similar subject areas/topics.

2 LDA – Generative process

1. In the lecture we introduced the generative process of LDA to create a document. Describe this generative process in your own words.

   - Inputs: Dirichlet parameters $\overrightarrow{\alpha}$, $\overrightarrow{\beta}$, document length n, number of topics K.
   - Goal: Create a document d with n words
   - Procedure:
     - Sample ϕ_k for each topic $k \in K$ from α
     - Sample θ_d from β
     - For all positions i in document d:
       * Sample a topic z_i from θ_d
       * Sample a term w_i from ϕ_{z_i}

2. You want to create a document according to the idea of the generative LDA process. The following document-topic distribution θ_{d_1} and topic-term distributions ϕ_k for topics $k \in (animals, sports, interest)$ have already been sampled from the Dirichlet distributions with parameters $\overrightarrow{\beta}$ and $\overrightarrow{\alpha}$:

$$\theta_{d_1} = \begin{pmatrix} p(animals) & = & 0.4 \\ p(sport) & = & 0.35 \\ p(interest) & = & 0.25 \end{pmatrix} \qquad \phi_{animal} = \begin{pmatrix} p(cat) & = & 0.3 \\ p(dog) & = & 0.2 \\ p(mouse) & = & 0.25 \\ p(ball) & = & 0.05 \\ p(play) & = & 0.1 \\ p(like) & = & 0.1 \end{pmatrix}$$

$$\phi_{sport} = \begin{pmatrix} p(cat) & = & 0.0 \\ p(dog) & = & 0.02 \\ p(mouse) & = & 0.0 \\ p(ball) & = & 0.4 \\ p(play) & = & 0.5 \\ p(like) & = & 0.08 \end{pmatrix} \qquad \phi_{interest} = \begin{pmatrix} p(cat) & = & 0.07 \\ p(dog) & = & 0.04 \\ p(mouse) & = & 0.01 \\ p(ball) & = & 0.13 \\ p(play) & = & 0.15 \\ p(like) & = & 0.6 \end{pmatrix}$$

Generate the document d_1 with 5 words using these distributions. You can use random.org[1] to generate random numbers, or the function

```
numpy.random.choice(['pick', 'one', 'please'], p=[0.3, 0.5, 0.2])
```

of the Python library numpy. Specify the resulting document, as well as the topics of each word.

Sample topics and words according to task 2.1, given θ_d and ϕ_k.

Possible example sentence completely generated in Python:
(For reproducibility seed of the random generator: numpy.random.seed(234))

Topic z_i	animal	interest	animal	interest	sport
Term w_i	cat	like	mouse	like	play

[1] `https://www.random.org/decimal-fractions/`

3 Gibbs Sampling for LDA

Consider the following two documents from which stop words have already been removed:
D_1: `Lecture LDA Gibbs Exam Institute Fun`
D_2: `Fun Exam LDA Gibbs`
Given is the following assignment of the topics to the words.

Topic	A	B	C	A	A	C
Word	Lecture	LDA	Gibbs	Exam	Institute	Fun

Topic	C	A	B	B
Word	Fun	Exam	LDA	Gibbs

1. Create a topic-term matrix that contains the corresponding counts over both documents.

	A	B	C
Lecture	1	0	0
LDA	0	2	0
Gibbs	0	1	1
Exam	2	0	0
Institute	1	0	0
Fun	0	0	2

2. Based on your matrix, calculate the topic distribution $\vec{z}$.

$\vec{z} = (4, 3, 3)$

3. Calculate the document-topic distribution $\vec{z}_d$ for the first document D_1.

$\vec{z}_d = (3, 1, 2)$

4. Given the previous results, perform a step of Gibbs Sampling for the word $w_1 =$ `Gibbs` in the first document D_1 as shown in the lecture. The Apriori counts for $\vec{\alpha}$ and $\vec{\beta}$ are set to $\vec{1}$. Which topic would you most probably assign to `Gibbs` after the step?

   After deleting the current topic from `Gibbs`, the counts are updated first:

	A	B	C
Lecture	1	0	0
LDA	0	2	0
Gibbs	0	1	**0**
Exam	2	0	0
Institute	1	0	0
Fun	0	0	2

   $\vec{z} = (4, 3, \mathbf{2})$ $\vec{z}_d = (3, 1, \mathbf{1})$

   Then the new 'probabilities' are calculated for each topic:

$$p(z_1 = A) \sim \frac{0+1}{4+6\cdot 1} \cdot \frac{3+1}{6+3-1} = 0.05$$
$$p(z_1 = B) \sim \frac{1+1}{3+6\cdot 1} \cdot \frac{1+1}{6+3-1} = 0.06$$
$$p(z_1 = C) \sim \frac{0+1}{2+6\cdot 1} \cdot \frac{1+1}{6+3-1} = 0.03$$

   These new 'probabilities' are normalized:

$$p(z_1 = A) = \frac{0.05}{0.05+0.06+0.03} = 0.36$$
$$p(z_1 = B) = \frac{0.06}{0.05+0.06+0.03} = 0.43$$
$$p(z_1 = C) = \frac{0.03}{0.05+0.06+0.03} = 0.21$$

   $\Rightarrow$ In this iteration, we would assign `Gibbs` the topic B.