Review of LSA, PLSA and LDA

- Jiayuan Li (jiayuan8@illinois.edu)

In this article, I will give a brief overview of three difference Natural Language Understanding algorithms, LSA, PLSA and LDA. The overview will focus on how these algorithms works and how are they related to each other.

The three algorithms aims to identify topics in each articles. They all treat each essay as a bag-of-words and uses different approaches to identify the latent topics in the essay. However, they also share some similarities.

LSA

First, we will introduce the approach of LSA (Latent Semantic Analysis). LSA starts from a matrix call Document-Term-Matrix(DTM). Each row of a DTM represents a single document and each column of DTM represents a distinct word. For example, an entry (i, j) with $DTM_{i,j} = 4$ means the word represented by index j has occurrences of 4 times in document i. All three methods aims to decompose the DTM into a document-topic matrix and a topic-term matrix in order to accomplish the task.

LSA utilizes a technique called singular value decomposition (SVD) to decompose the DTM matrix, which can be represented as $DTM = U\Sigma V^t$. Assume the shape of the DTM matrix is $m \times n$, then the shape of U, Σ and V will be $m \times m$, $m \times n$ and $n \times n$. Moreover, we will have only the diagonal entries of Σ as non-zero values and $\Sigma_{i,i}$ is monotonic decreasing as i increases.

To find the most significant t topics in the collections of document, we can use a variant of SVD called truncated SVD. The general idea is to only keep the first t columns of U, the first t columns of V and the first t rows and columns of Σ. Then we will have U, Σ, V as matrices with shape $m \times t, t \times t$ and $n \times t$ respectively. The first matrix U is a document-topic matrix where each row represents a document and each column represents a topic. The matrix V^T is a topic-term matrix where each row represents a topic and each column represents a word. The diagonal entries of matrix Σ is the singular value of each topic. V can help us identify what words is each topic exposed to and how significant the exposure is. U can help us identify what topics is each topic exposed to and how significant the exposure is.

The LSA method is quite intuitive and easy to implement. However, it neglect the potential connection between words. For example, words "stock" and "equity" have very similar meanings whereas in the LSA model they are treated as two independent words and may lead same topic being separated to different distinct topics. Moreover, then entries in U and V may contain negative values which may produce problems in interpretability.

pLSA

Now, we will introduce pLSA (probabilistic Latent Semantic Analysis). Compare to traditional LSA, pLSA introduces a joint probability $p(d, w)$ that models the probability of having a word w and a document d together. The probability can be calculated given the formula (1) in the following steps:

$$p(d,w) = p(d) \cdot p(w|d) \quad (1) \qquad p(w|d) = \sum_{z \in Z} p(w|z) \cdot p(z|d) \quad (2)$$

$$\rightarrow p(d,w) = p(d) \cdot \sum_{z \in Z} p(w|z) \cdot p(z|d) = \sum_{z \in Z} p(z) \cdot p(d|z) \cdot p(w|z) \quad (3)$$

In the formula (2), we have z representing a topic and $p(w|z)$ indicate the probability that word w is in topic z and $p(z|d)$ indicate the probability that document d has topic z.

To draw similarity between LSA and pLSA, we can take a look at formula (3). With formula (1) and (2), we can rewrite $p(d,w)$ in formula in the form of (3). Given (3), we can notice that $p(d,w)$ is rearranged to a SVD form that $p(d,w)$ can correspond to the DTM matrix in LSA, $p(d|z)$ can correspond to the document-topic matrix U in LSA, $p(z)$ can correspond to the singular value matrix Σ in LSA, and $p(w|z)$ can correspond to the word-topic matrix V in LSA.

The model can be solved by an iterative optimization method called expectation-maximization (EM). The parameters we want to solve for are the probability $p(w|z)$ and $p(z|d)$. By following formula (2). We can define $p(w|z) = \phi_{(z,w)}$ and $p(z|d) = \theta_{(d,z)}$. The notation Φ and Θ are matrices formed by $\phi_{(z,w)}$ and $\theta_{(d,z)}$, and ϕ_z and θ_d are vectors formed by their corresponding elements. For the vocabulary V, we should have that $\sum_{w \in V} \phi_{(z,w)} = 1$ for each $z \in Z$. For the topics set T, we should have $\sum_{z \in T} \theta_{(d,z)} = 1$ for each document d. The, the generative algorithm can be described as following:

For each document d and each token position i, We can generate topic $z \sim \text{Multinomial}(\theta_d)$ and word $w \sim \text{Multinomial}(\phi_z)$. Then the probability that the i^{th} token of document d is w and the overall joint distribution of the data set can be written as

$$p(d_i = w|\Phi, \theta_d) = \sum_{z \in T} \phi_{(z,w)} \cdot \theta_{(d,w)} \quad \text{and} \quad p(W|\Phi, \Theta) = \prod_{d \in D} \prod_{i \in N_d} \sum_{z \in T} \phi_{(z,w)} \cdot \theta_{(d,w)}$$

Then the maximization problem can be reduced to

$$\arg\max_{\Phi \ \Theta} \Big[\log p(W|\Phi, \Theta) + \sum_{d \in D} \lambda_d (1 - \sum_{z \in T} \theta_{(d,z)}) + \sum_{z \in T} \sigma_z (1 - \sum_{w \in V} \phi_{(z,w)}) \Big]$$

where the second and the third terms comes from Lagrange Multipliers that guarantees the multinomial parameters to be ranged in 0 and 1. By following the EM algorithm, we can have the optimal Φ and Θ.

pLSA can have better performance than LSA algorithm generally. However, the number of parameters grows as the number of document grows which may result in efficiency problems. Meanwhile, it still has the problem that LSA has. The model itself also cannot assign probability to new document d_{new}.

LDA in Brief

Now, we will introduce LDA (Latent Dirichlet Allocation) method. In short, LDA is an Bayesian improvement of pLSA which can assign probability to any new document haven't seen. LDA also assigns each document-topic and term-topic pairs with an estimation of distribution from Dirichlet Priors. (Dirichlet distribution is a multivariate distribution with a vector parameter α).

To form the problem more formally, we recall the optimization process we demonstrated above for Φ and Θ. In pLSA, Φ and Θ are parameters that are obtained only from the maximization problem in the dataset. In LDA, we instead add the step that draw $\theta_d \sim \text{Dirichlet}(\alpha)$ and $\phi_z \sim \text{Dirichlet}(\beta)$ before the step of drawing $z \sim \text{Multinomial}(\theta_d)$ and $w \sim \text{Multinomial}(\phi_z)$. Therefore, we can see that the LDA can be generalized to new documents haven't seen in the dataset more easily because of we can always sample from the Dirichlet prior.

Reference

[1] https://arxiv.org/pdf/1212.3900.pdf

[2] https://towardsdatascience.com/topic-modeling-with-lsa-plsa-lda-nmf-bertopic-top2vec-a-comparison-5e6ce4b1e4a5

[3] https://medium.com/nanonets/topic-modeling-with-lsa-psla-lda-and-lda2vec-555ff65b0b05

[4] http://lsa.colorado.edu/