CS5560 Knowledge Discovery and Management

Problem Set 5

July 3 (T), 2017

Name:

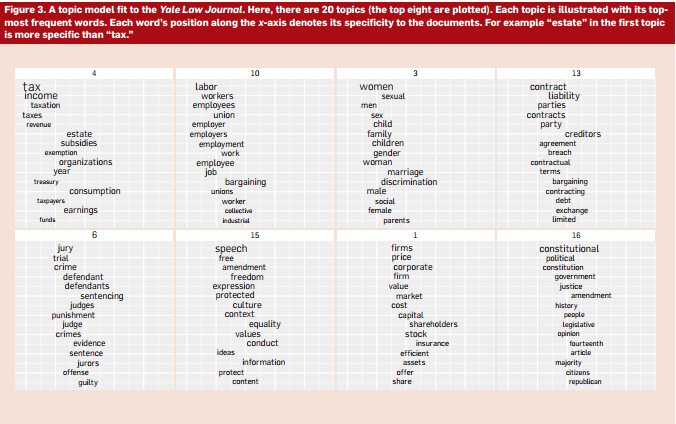
Class ID:

1. LDA

Read the following articles to learn more about LDA

* <https://algobeans.com/2015/06/21/laymans-explanation-of-topic-modeling-with-lda-2/>
* <http://engineering.intenthq.com/2015/02/automatic-topic-modelling-with-lda/>

Consider the topics discovered from Yale Law Journal. (Here the number of topics was set to be 20.) Topics about subjects like about discrimination and contract law.



1. Describe the overall process to generate such topics from the corpus.

Ans.

LDA(latent dirichlet association) is a technique that automatically discovers topics that these documents contains. To evaluate the LDA model, one document is taken and is split into two. The first half is fed into LDA to compute the topics composition, from that composition then the word distribution is eliminated. This distribution is t hen compared with the word distribution of 2nd half of the document, a measure of distance is extracted. It makes use of gibbs sampling for this process.

In LDA each document can be viewed as a mixture of various toips and assigned to it vin lda.

Ex. K=2,set topics CAT or DOG.

Words in the documents such as meow, kitten can be classified as CAT and puppies, bark, Labrador might be classified as DOG based on distance measurement.

1. Draw a knowledge graph for Topic 3 in Yale Law Journal (The First Figure).

Ans.A picture containing text, whiteboard

Description generated with very high confidence

1. Each topic is illustrated with its topmost frequent words. Each word’s position along the x-axis denotes its specificity to the documents. For example “estate” in the first topic is more specific than “tax.” (the second figure). Describe how to determine the generality or specificity of the terms in a topic.

Ans. When we deal with a huge amount of data on a daily basis. Much of this is textual as we try to get an intimate understanding of articles, web pages, blog posts and opinions. Fortunately, **text mining**, a branch of data science, gives us a hand.

In this post we’re going to describe how topics can be automatically assigned to text documents; this process is named, unsurprisingly, **topic-modelling**. It works like fuzzy (or soft) clustering since it’s not a strict categorisation of the document to its dominant topic.

Let’s start with an example: the optimal topic-modelling outcome for Shakespeare’s Romeo & Juliet would be a composition of topics of circa 50% tragedy and 50% romance. Surely, topics like social networks, football and indian cuisine don’t appear in the play, so their weights would be all 0%.

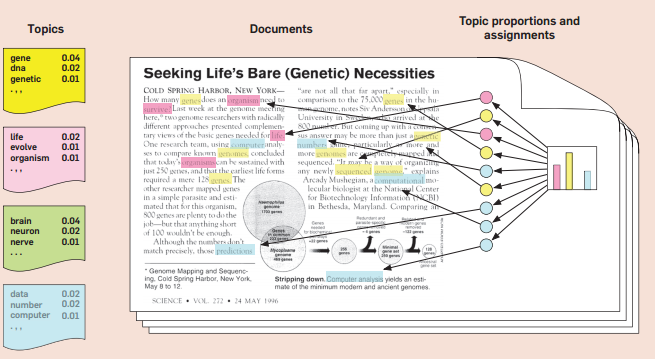
One of the most advanced algorithms for doing topic-modelling is **Latent Dirichlet**

**Allocation** (or **LDA**). This is a probabilistic model developed by Blei, Ng and Jordan in 2003. LDA is an iterative algorithm which requires only three parameters to run: when they’re chosen properly, its accuracy is pretty high. Unfortunately, one of the required parameters is the number of topics: exactly as happens with K-means, this requires a deep a-priori knowledge of the dataset.

A good measure to evaluate the performance of LDA is **perplexity**. This measure is taken from information theory and measures how well a probability distribution predicts an observed sample. To evaluate the LDA model, one document is taken and split in two. The first half is fed into LDA to compute the topics composition; from that composition, then, the word distribution is estimated. This distribution is then compared with the word distribution of the second half of the document. a a measure of distance is extracted. Thanks to this measure, in practice, perplexity is often used to select the best number of topics of the LDA model.

Under the hood, LDA models both the topics-per-document and the topic-per-word distribution as Dirichlet distributions (that’s why it appears in its name). By using a Markov Chain Monte Carlo (MCMC) method to sample and approximate the underlying Markov Chain stationary distribution (called **Gibbs sampling**), the whole process is iterative, pretty fast, convergent and accurate

1. Describe the inference algorithm that was used in LDA.



In LDA, they used observed variables to infer the hidden structure.

Step 1: Assign the value k, how many topics to extract.

Step 2: Algorithm assigns temporary topic to envy word

Step 3: In an iterative way, algorithm will check and update the topics assigned base do

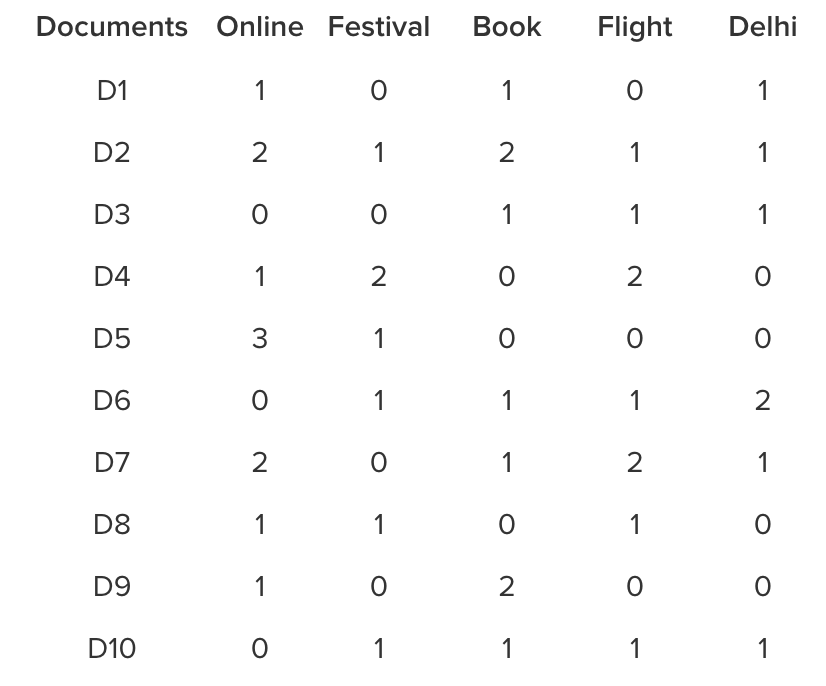
* How prevalent is the word across topics?
* How prevalent are topics in the document?

1. K-means clustering vs. LDA

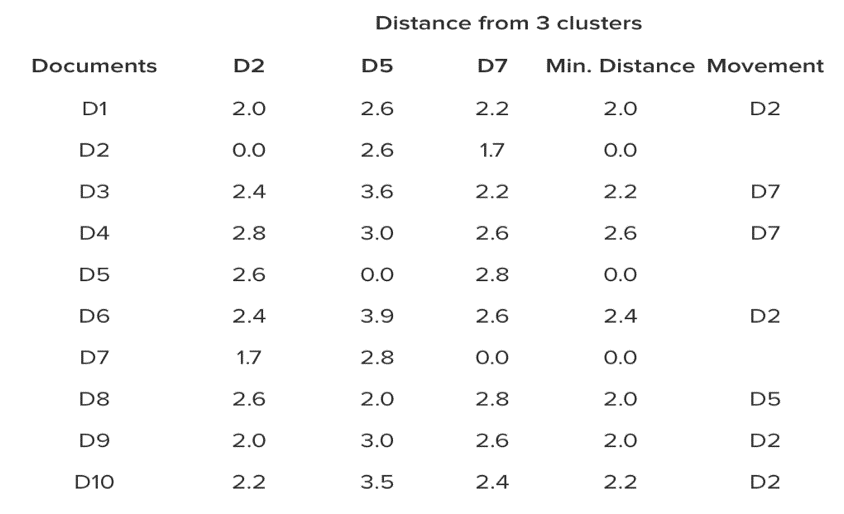
Read the K-means clustering for text clustering from <https://www.experfy.com/blog/k-means-clustering-in-text-data>

1. Describe the steps how the following 10 documents have moved into 3 different clusters using clustered using k-means (K=3).

**Document/Term Matrix**



**Distance Matrix**



Ans.

A close up of text on a whiteboard

Description generated with high confidence

* Hence, 10 documents have moved into 3 different clusters. Instead of Centroids, Medoids are formed and again distances are re-calculated to ensure that the documents who are closer to a medoid is assigned to the same cluster.
* Medoids are used to build the story for each cluster.

But there is still one important question remaining: How do you choose the optimal number of clusters?

One approach would be to use the Elbow method to choose the optimal number of clusters. This is based on plotting the cost function for various number of clusters and identifying the breakpoints. If adding more clusters is not significantly reducing the variance within the cluster, one should stop adding more clusters. Although this method cannot give you the optimal number of clusters as an exact point, it can give you an optimal range.

1. Describe the difference (pro and con) of k-means clustering and the LDA topic discovery model.

Ans.Time Complexity

K-means is linear in the number of data objects i.e. O(n)O(n), where *n* is the number of data objects. The time complexity of most of the hierarchical clustering algorithms is quadratic i.e. O(n2)O(n2). Therefore, for the same amount of data, hierarchical clustering will take quadratic amount of time. Imagine clustering 1 million records?

# Shape of Clusters

K-means works well when the shape of clusters are hyper-spherical (or circular in 2 dimensions). If the natural clusters occurring in the dataset are non-spherical then probably K-means is not a good choice.

# Repeatability

K-means starts with a random choice of cluster centers, therefore it may yield different clustering results on different runs of the algorithm. Thus, the results may not be repeatable and lack consistency. However, with hierarchical clustering, you will most definitely get the same clustering results.

Off course, K-means clustering requires prior knowledge of K (or number of clusters), whereas in hierarchical clustering you can stop at whatever level (or clusters) you wish.