Latent Dirichlet Allocation (LDA)

LDA

- A statistical model for discovering the abstract *topics* in ***topic modeling***.

What is topic modeling?

Topic modeling is a method for ***unsupervised*** classification of documents, similar to clustering on numeric data,

which finds some natural groups of items (topics) even when we're not sure what we're looking for.

Why topic modeling?

Topic modeling provides methods for automatically organizing, understanding, searching, and summarizing large electronic archives.

It can help with the following:

- discovering the hidden themes in the collection.
- classifying the documents into the discovered themes.
- using the classification to organize/summarize/search the documents.

LDA

- It is one of the most popular topic modeling methods.
- Each document is made up of various words, and each topic also has various words belonging to it.
- The aim of LDA is to find topics a document belongs to, based on the words in it.

Example:

- Let’s say we have 2 topics that can be classified as *CAT_related* and *DOG_related.*
- A topic has probabilities for each word, so words such as *milk*, *meow*, and *kitten*, will have a higher probability in the *CAT_related* topic than in the *DOG_related* one.
- The *DOG_related* topic, likewise, will have high probabilities for words such as *puppy*, *bark*, and *bone.*
- If we have a document containing the following sentences:
- “*Dogs* like to *chew* on *bones* and fetch sticks”.
  “*Puppies* drink *milk*.”
  “Both like to *bark*.”

- We can easily say it belongs to topic *DOG_related* because it contains words *such as Dogs*, *bones, puppies*, and *bark*.
- Even though it contains the word *milk* which belongs to the topic *CAT_related*, the document belongs to *DOG_related* as more words match with it.

How does LDA work?

There are 2 parts in LDA:

- The ***words that belong to a document***, that we already know.
- The ***words that belong to a topic*** or the probability of words belonging into a topic, that we need to calculate.

Algorithm to find the latter

- Go through each document and randomly assign each word in the document to one of ***k*** topics (*k* is chosen beforehand).
- For each document ***d***, go through each word ***w*** and compute :
- **p(topic *t* | document *d*)**: the **proportion of words in document *d* that are assigned to topic *t***. Tries to capture how many words belong to the topic *t* for a given document *d*. Excluding the current word.
- **p(word *w*| topic *t*)**: **the proportion of assignments to topic *t* over all documents that come from this word *w***. Tries to capture how many documents are in topic *t* because of word *w*.

- Update the probability for the word *w* belonging to topic *t*, as

p(word w with topic t) = p(topic t | document d) * p(word w | topic t)