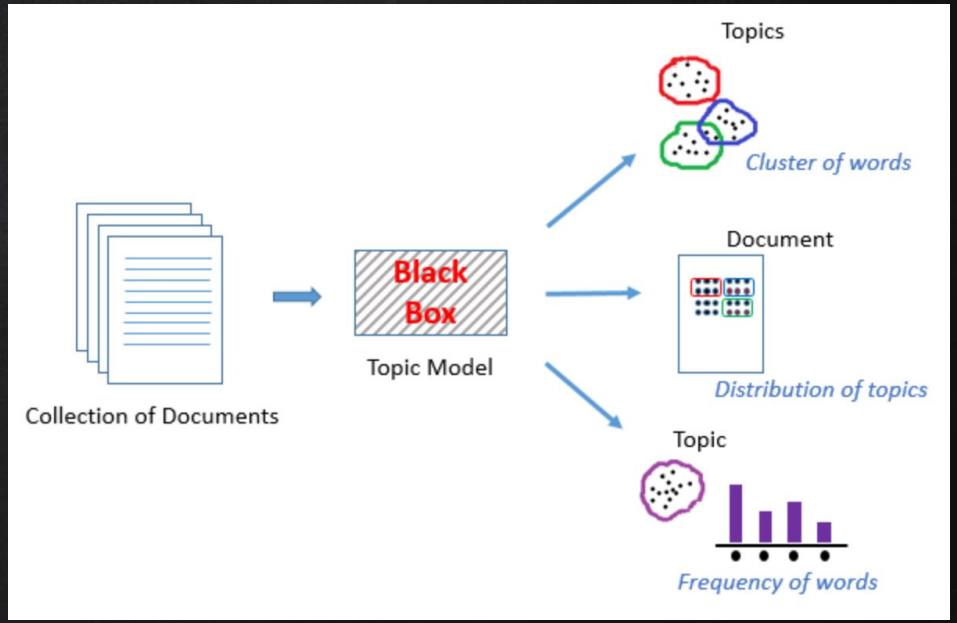
# Topic Modelling Using Latent semantic analysis


# INTRODUCTION

# What is a topic model?

A Topic Model can be defined as an unsupervised technique to discover topics across various text documents. These topics are abstract in nature, i.e., words which are related to each other form a topic. Similarly, there can be multiple topics in an individual document.

For the time being, let's understand a topic model as a black box in the diagram.



# When is topic modelling used

Imagine  arranging similar books together. Now suppose you have to perform a similar task with a few digital text documents. You would be able to manually accomplish this, as long as the number of documents is manageable. But what happens when there's an impossible number of these digital text documents?

Topic modeling helps in exploring large amounts of text data, finding clusters of words, similarity between documents, and discovering abstract topics. As if these reasons weren't compelling enough, topic modeling is also used in search engines wherein the search string is matched with the results.


# Latent Semantic Analysis (LSA)

# Overview of Latent Semantic Analysis (LSA)

All languages have their own intricacies and nuances which are quite difficult for a machine to capture (sometimes they're even misunderstood by us humans!). This can include different words that mean the same thing, and also the words which have the same spelling but different meanings.

For example, consider the following two sentences:

1. I liked his last **novel** quite a lot.
2. We would like to go for a **novel** marketing campaign.

In the first sentence, the word 'novel' refers to a book, and in the second sentence it means new or fresh.

We can easily distinguish between these words because we are able to understand the context behind these words. However, a machine would not be able to capture this concept as it cannot understand the context in which the words have been used. This is where Latent Semantic Analysis (LSA) comes into play as it attempts to leverage the context around the words to capture the hidden concepts, also known as topics.

So, simply mapping words to documents won't really help. What we really need is to figure out the hidden concepts or topics behind the words. LSA is one such technique that can find these hidden topics.


# Role of linear algebra

# In

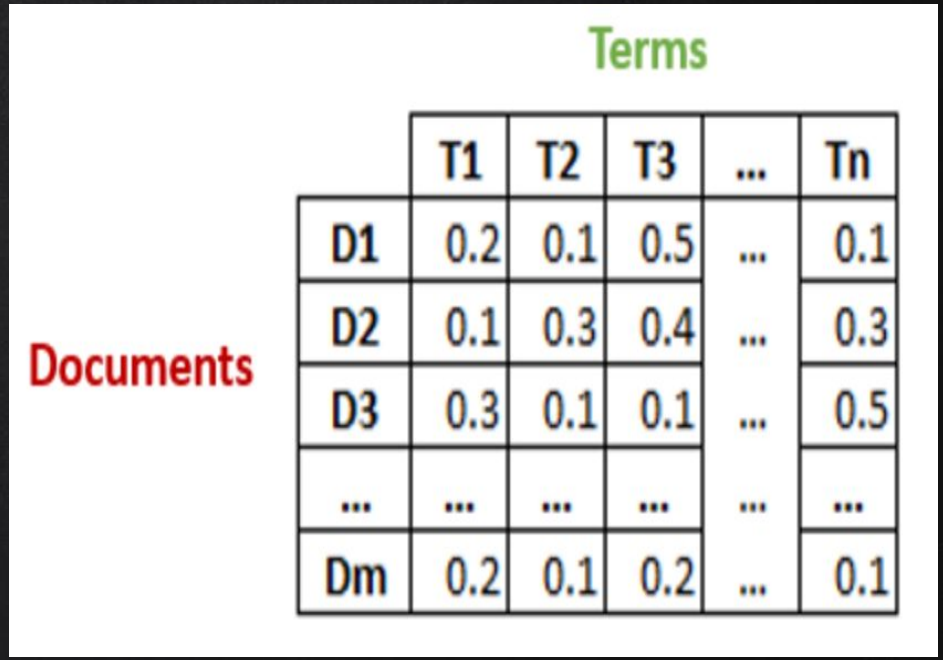
# latent semantic analysis

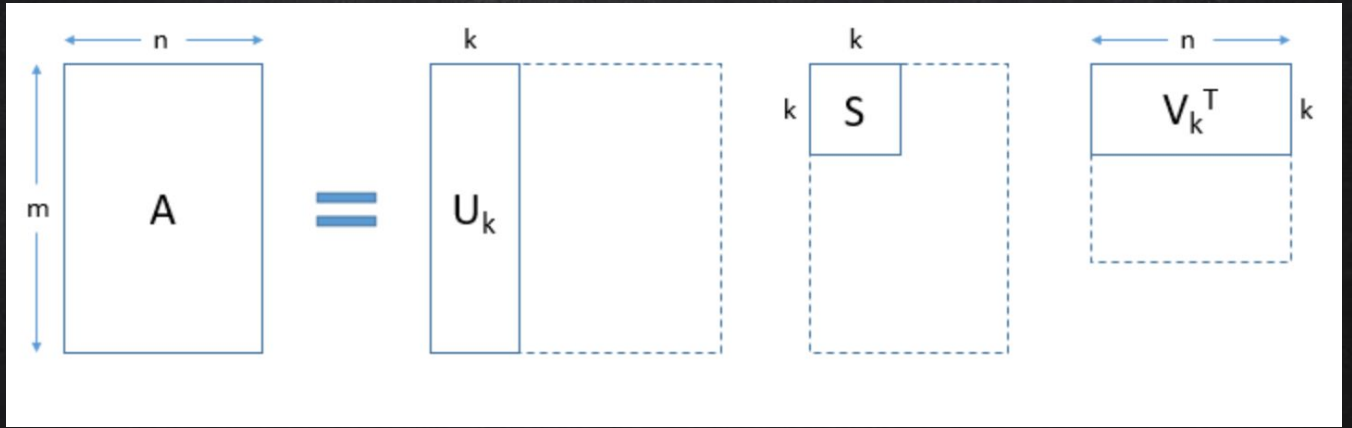
# Steps involved in the implementation of LSA

Let's say we have **m** number of text documents with **n** number of total unique terms (words). We wish to extract **k** topics from all the text data in the documents. The number of topics, k, has to be specified by the user.

- A document-term matrix of shape **m x n** having TF-IDF scores **as shown in the diagram** is generated
- Then, we will reduce the dimensions of the above matrix to **k** (no. of desired topics) dimensions, using singular-value decomposition (SVD).
- SVD decomposes a matrix into three other matrices. Suppose we want to decompose a matrix A using SVD. It will be decomposed into matrix U, matrix S, and $V^T$ (transpose of matrix V).

$$A = USV^T$$






- Each row of the matrix $U_k$ **(document-term matrix)** is the vector representation of the corresponding document. The length of these vectors is k, which is the number of desired topics. Vector representation for the terms in our data can be found in the matrix $V_k$ **(term-topic matrix)**.
- So, SVD gives us vectors for every document and term in our data. The length of each vector would be **k**. We can then use these vectors to find similar words and similar documents using the cosine similarity method.

# Links