DATA ANALYTICS

Unit 5: Advanced Techniques

Swati Pratap Jagdale
Department of Computer Science and Engineering

DATA ANALYTICS

Unit 5: Latent Semantic Analysis (LSA)

Swati Pratap Jagdale

Department of Computer Science and Engineering

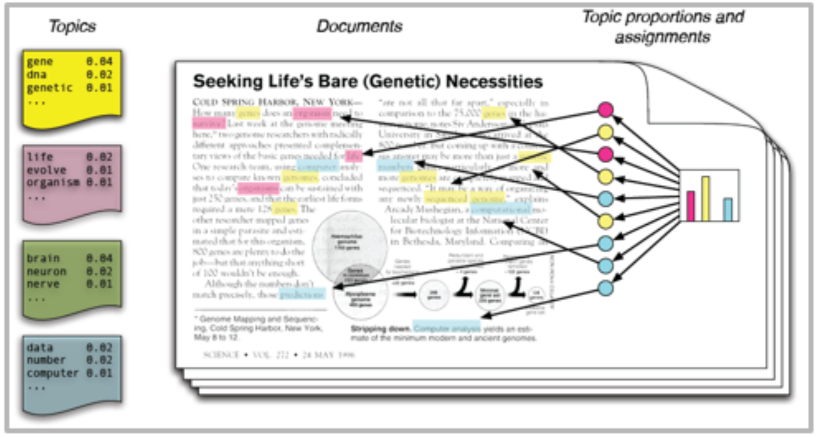
Latent Semantic Analysis (LSA)

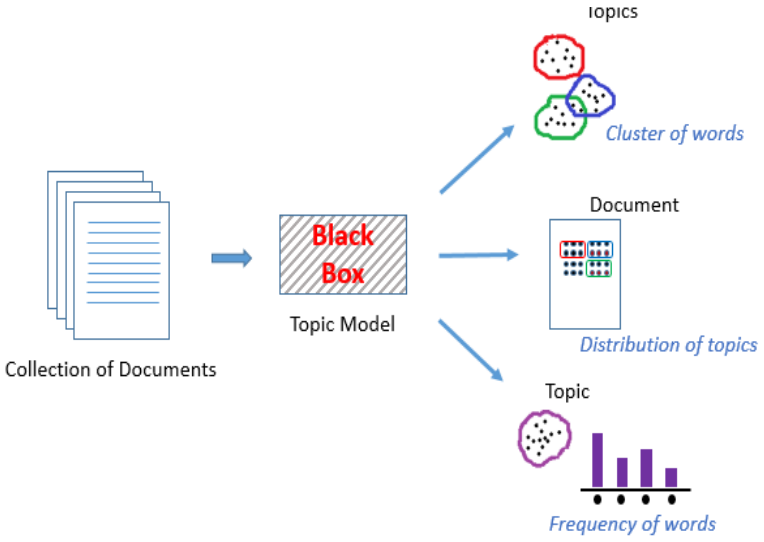
- Content-based recommendation systems: How do we extract keywords or ‘topics’ (latent patterns) in large documents (news articles, movie plots, book blurbs, relevant job descriptions, etc.) to create summaries or retrieve meaningful information?

- Latent Semantic Analysis, or LSA, is one of the foundation techniques in topic modeling.

- **What is a topic model?**
  An unsupervised technique to discover topics across various text documents.

  Every topic is defined by the proportion of different words it contains.





Latent Semantic Analysis (LSA)

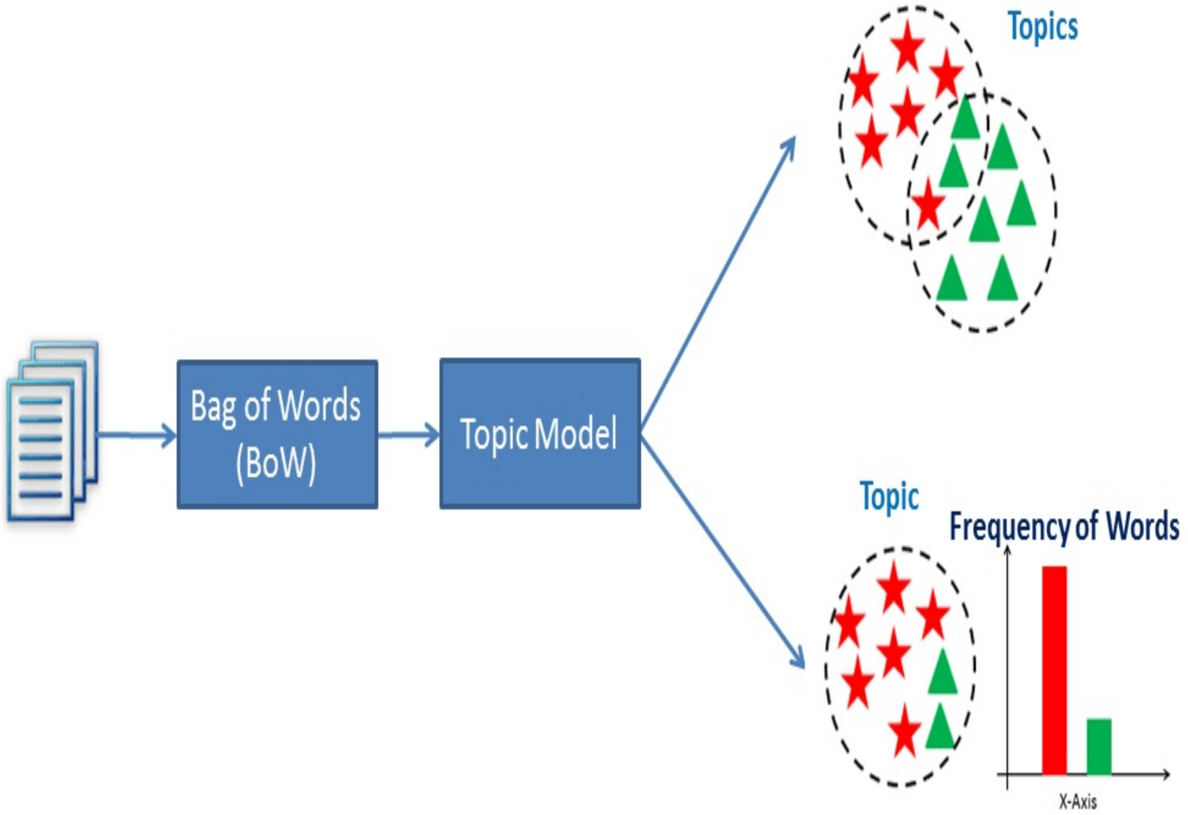
To discover the hidden topics from given documents

Discovering topics are beneficial for various purposes such as for clustering documents, organizing online available content for information retrieval and recommendations.

Multiple content providers and news agencies are using topic models for recommending articles to readers.

Latent Semantic Analysis (LSA)- Topic Modeling

- Topic Modeling automatically discover the hidden themes from given documents.
- It is an unsupervised text analytics algorithm that is used for finding the group of words from the given document.
- These group of words represents a topic.
- There is a possibility that, a single document can associate with multiple themes. for example, a group words such as 'patient', 'doctor', 'disease', 'cancer', and 'health' will represents topic **'healthcare'.**



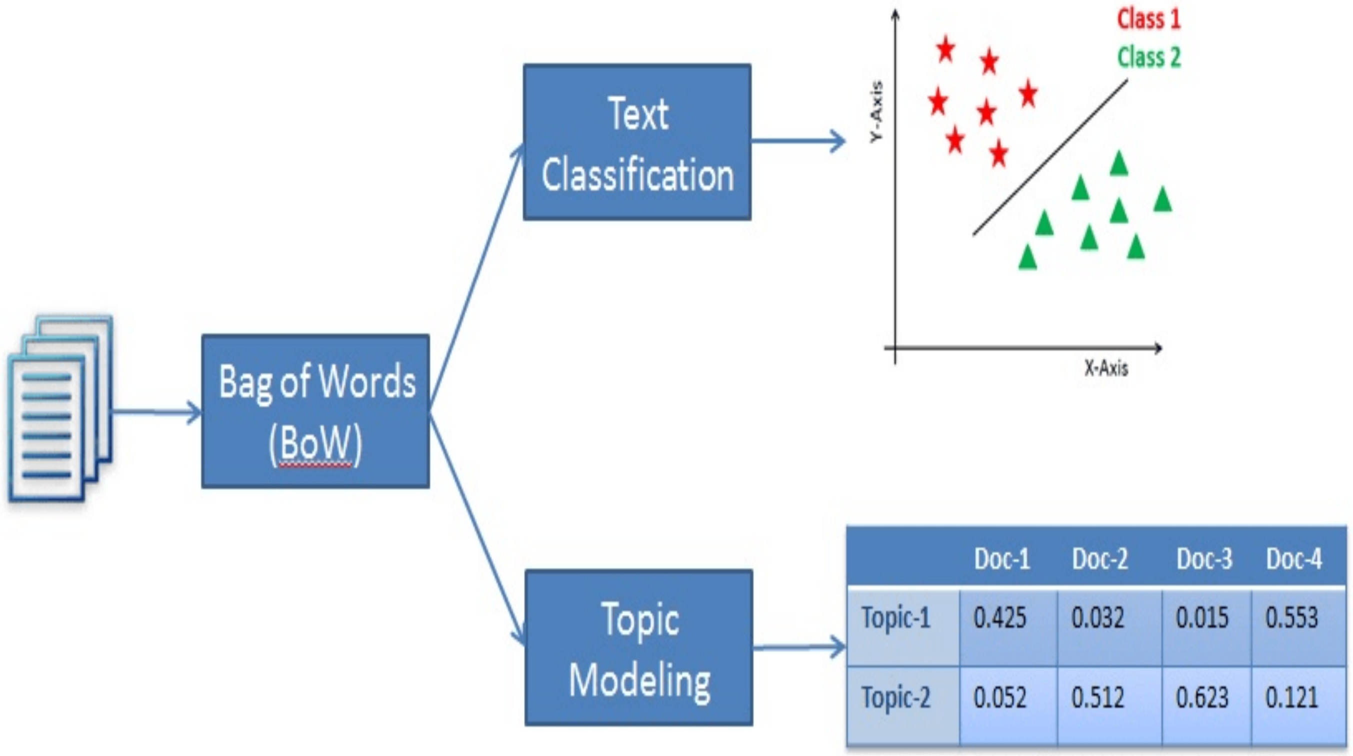
Comparison Between Text Classification and topic modeling

Text classification is a supervised machine learning problem, where a text document or article classified into a pre-defined set of classes.

Topic modeling is the process of discovering groups of co-occurring words in text documents. These group co-occurring related words makes "topics".

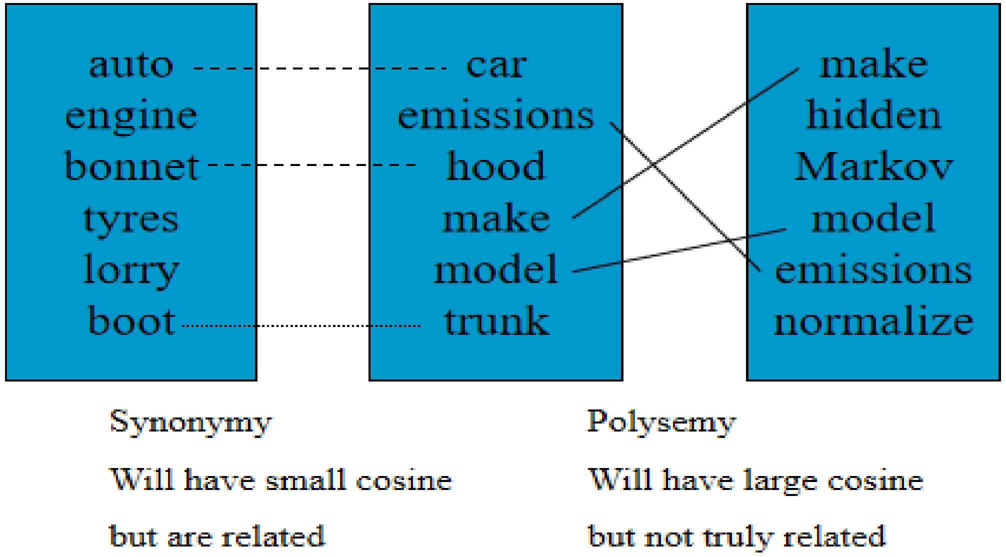
It is a form of unsupervised learning, so the set of possible topics are unknown.

Topic modeling can be used to solve the text classification problem. Topic modeling will identify the topics presents in a document" while text classification classifies the text into a single class.



Latent Semantic Analysis (LSA)

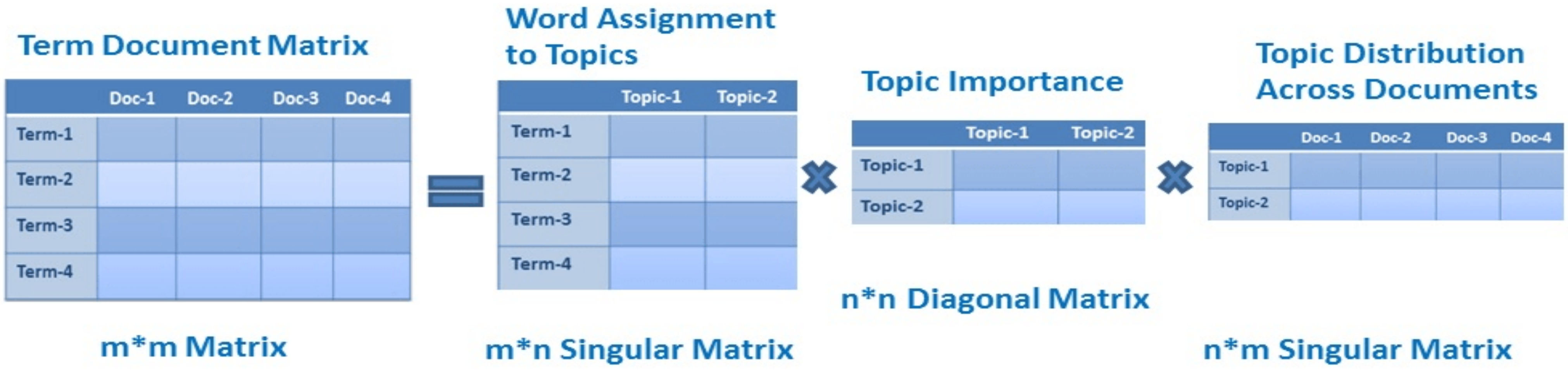
- Example: Vector Space Model
  (from Lillian Lee)



Latent Semantic Indexing was proposed to address these two problems with the vector space model for Information Retrieval

Latent Semantic Analysis (LSA)

LSA (Latent Semantic Analysis) also known as LSI (Latent Semantic Index) LSA uses bag of word(BoW) model, which results in a term-document matrix(occurrence of terms in a document). Rows represent terms and columns represent documents. LSA learns latent topics by performing a matrix decomposition on the document-term matrix using Singular value decomposition. LSA is typically used as a dimension reduction or noise reducing technique.



Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA)

- **A Simple Example:** Technical Memo Titles

Topic: Human Computer Interaction (HCI)
c1: *Human* machine *interface* for ABC *computer* applications
c2: A *survey* of *user* opinion of *computer system response time*
c3: The *EPS user interface* management *system*
c4: *System* and *human system* engineering testing of *EPS*
c5: Relation of *user* perceived *response time* to error measurement

Topic: Graph theory (conceptually disjoint from HCI)
m1: The generation of random, binary, ordered *trees*
m2: The intersection *graph* of paths in *trees*
m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
m4: *Graph minors*: A *survey*

Complete paper with detailed notes

Latent Semantic Analysis (LSA)

- ## A Simple Example

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

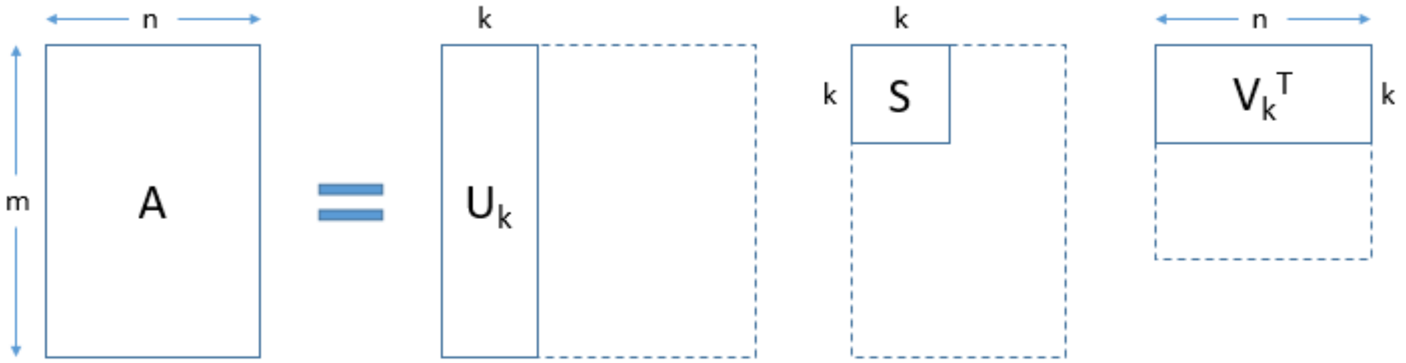
A word by context matrix, ***A***, formed from the titles of five articles about human-computer interaction and four about graph theory. Cell entries are the number of times that a word (rows) appeared in a title (columns) for words that appeared in at least two titles.

r (human,user) = -.38 r (human,minors) = -.29

Singular Value Decomposition

- The mxn term-document matrix is subject to singular value decomposition

$$A = USV^T$$



- Rank-reduced Singular Value Decomposition (SVD) performed on matrix, all but the k highest singular values are set to 0; this produces a k-dimensional approximation of the original matrix (in least-squares sense) this is the “semantic space”
- Compute similarities between entities in semantic space (usually with cosine)

Latent Semantic Analysis (LSA)

- A Simple Example

{U} =

0.22	-0.11	0.29	-0.41	-0.11	-0.34	0.52	-0.06	-0.41
0.20	-0.07	0.14	-0.55	0.28	0.50	-0.07	-0.01	-0.11
0.24	0.04	-0.16	-0.59	-0.11	-0.25	-0.30	0.06	0.49
0.40	0.06	-0.34	0.10	0.33	0.38	0.00	0.00	0.01
0.64	-0.17	0.36	0.33	-0.16	-0.21	-0.17	0.03	0.27
0.27	0.11	-0.43	0.07	0.08	-0.17	0.28	-0.02	-0.05
0.27	0.11	-0.43	0.07	0.08	-0.17	0.28	-0.02	-0.05
0.30	-0.14	0.33	0.19	0.11	0.27	0.03	-0.02	-0.17
0.21	0.27	-0.18	-0.03	-0.54	0.08	-0.47	-0.04	-0.58
0.01	0.49	0.23	0.03	0.59	-0.39	-0.29	0.25	-0.23
0.04	0.62	0.22	0.00	-0.07	0.11	0.16	-0.68	0.23
0.03	0.45	0.14	-0.01	-0.30	0.28	0.34	0.68	0.18

Latent Semantic Analysis (LSA)

- A Simple Example

{S} =

3.34								
	2.54							
		2.35						
			1.64					
				1.50				
					1.31			
						0.85		
							0.56	
								0.36

Latent Semantic Analysis (LSA)

- ## A Simple Example

{V} =

0.20	0.61	0.46	0.54	0.28	0.00	0.01	0.02	0.08
-0.06	0.17	-0.13	-0.23	0.11	0.19	0.44	0.62	0.53
0.11	-0.50	0.21	0.57	-0.51	0.10	0.19	0.25	0.08
-0.95	-0.03	0.04	0.27	0.15	0.02	0.02	0.01	-0.03
0.05	-0.21	0.38	-0.21	0.33	0.39	0.35	0.15	-0.60
-0.08	-0.26	0.72	-0.37	0.03	-0.30	-0.21	0.00	0.36
0.18	-0.43	-0.24	0.26	0.67	-0.34	-0.15	0.25	0.04
-0.01	0.05	0.01	-0.02	-0.06	0.45	-0.76	0.45	-0.07
-0.06	0.24	0.02	-0.08	-0.26	-0.62	0.02	0.52	-0.45

Latent Semantic Analysis (LSA)

- **Original term-document matrix**

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

After LSA

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04
computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
user	0.26	0.84	0.61	0.70	0.39	0.03	0.08	0.12	0.19
system	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
response	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
time	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
EPS	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.20	-0.11
survey	0.10	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
trees	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
graph	-0.06	0.34	-0.15	-0.30	0.20	0.31	0.69	0.98	0.85
minors	-0.04	0.25	-0.10	-0.21	0.15	0.22	0.50	0.71	0.62

r (human.user) = -.38 r (human.minors) = -.29

r (human,user) = .94 r (human,minors) = -.83

Effect of SVD on the correlation matrix

LSA Titles example:
Correlations between titles in raw data

	c1	c2	c3	c4	c5	m1	m2	m3
c2	-0.19							
c3	0.00	0.00						
c4	0.00	0.00	0.47					
c5	-0.33	0.58	0.00	-0.31				
m1	-0.17	-0.30	-0.21	-0.16	-0.17			
m2	-0.26	-0.45	-0.32	-0.24	-0.26	0.67		
m3	-0.33	-0.58	-0.41	-0.31	-0.33	0.52	0.77	
m4	-0.33	-0.19	-0.41	-0.31	-0.33	-0.17	0.26	0.56

0.02	
-0.30	**0.44**

Correlations in first-two dimension space post LSA

c2	0.91							
c3	1.00	0.91						
c4	1.00	0.88	1.00					
c5	0.85	0.99	0.85	0.81				
m1	-0.85	-0.56	-0.85	-0.88	-0.45			
m2	-0.85	-0.56	-0.85	-0.88	-0.44	1.00		
m3	-0.85	-0.56	-0.85	-0.88	-0.44	1.00	1.00	
m4	-0.81	-0.50	-0.81	-0.84	-0.37	1.00	1.00	1.00

Pros and Cons of LSA

Pros:

- LSA is fast and easy to implement.
- It gives decent results, much better than a plain vector space model.

Cons:

- Since it is a linear model, it might not do well on datasets with non-linear dependencies.
- LSA assumes a Gaussian distribution of the terms in the documents, which may not be true for all problems.
- LSA involves SVD, which is computationally intensive and hard to update as new data comes up.

References

https://www.analyticsvidhya.com/blog/2018/10/stepwise-guide-topic-modeling-latent-semantic-analysis/

https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/

http://lsa.colorado.edu/papers/dp1.LSAintro.pdf

https://www.datacamp.com/community/tutorials/discovering-hidden-topics-python

Latent Semantic Analysis (LSA)

- **Overview of Latent Semantic Analysis (LSA)**

- consider the following two sentences:

- I liked his last **novel** quite a lot.
- 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.
- 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.

Latent Semantic Analysis (LSA)

- **Singlar Value Decomposition (SVD)**

- SVD is basically a factorization of the matrix. Here, we are reducing the number of rows (which means the number of words) while preserving the similarity structure among columns (which means paragraphs).
- unique mathematical decomposition of a matrix into the product of three matrices:
  two with orthonormal columns
  one with singular values on the diagonal
- tool for dimension reduction
- similarity measure based on co-occurrence
- finds optimal projection into low-dimensional space

Latent Semantic Analysis (LSA)

- **Singlar Value Decomposition (SVD)**

- can be viewed as a method for rotating the axes in n-dimensional space, so that the first axis runs along the direction of the largest variation among the documents
  - the second dimension runs along the direction with the second largest variation
  - and so on
- generalized least-squares method

THANK YOU

Swati Pratap Jagdale
Department of Computer Science
swatigambhire@pes.edu