DATA ANALYTICS

Unit 5: Advanced Techniques

Swati Pratap Jagdale
Department of Computer Science and Engineering

DATA ANALYTICS

Unit 5: Latent Semantic Analysis (LSA)

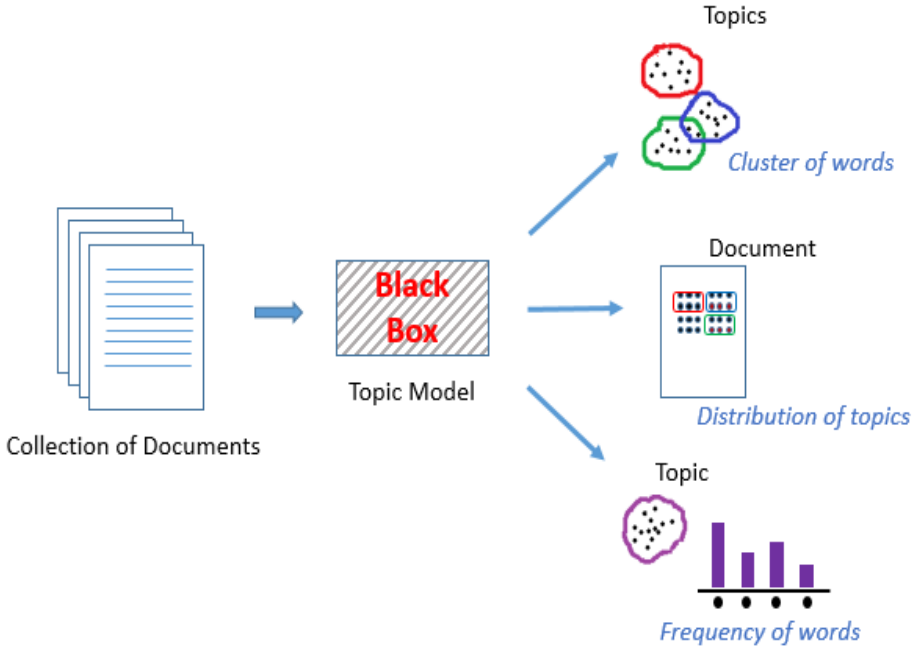
Swati Pratap Jagdale

Department of Computer Science and Engineering

Latent Semantic Analysis (LSA)

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

- **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, as illustrated in the below figure:



- This black box (topic model) forms clusters of similar and related words which are called topics. These topics have a certain distribution in a document, and every topic is defined by the proportion of different words it contains.

Latent Semantic Analysis (LSA)

What is a topic model?

- it is a process to automatically identify topics present in a text object and to derive hidden patterns exhibited by a text corpus. Thus, assisting better decision making.
- Topic Modelling is different from rule-based text mining approaches that use regular expressions or dictionary based keyword searching techniques. It is an unsupervised approach used for finding and observing the bunch of words (called “topics”) in large clusters of texts.

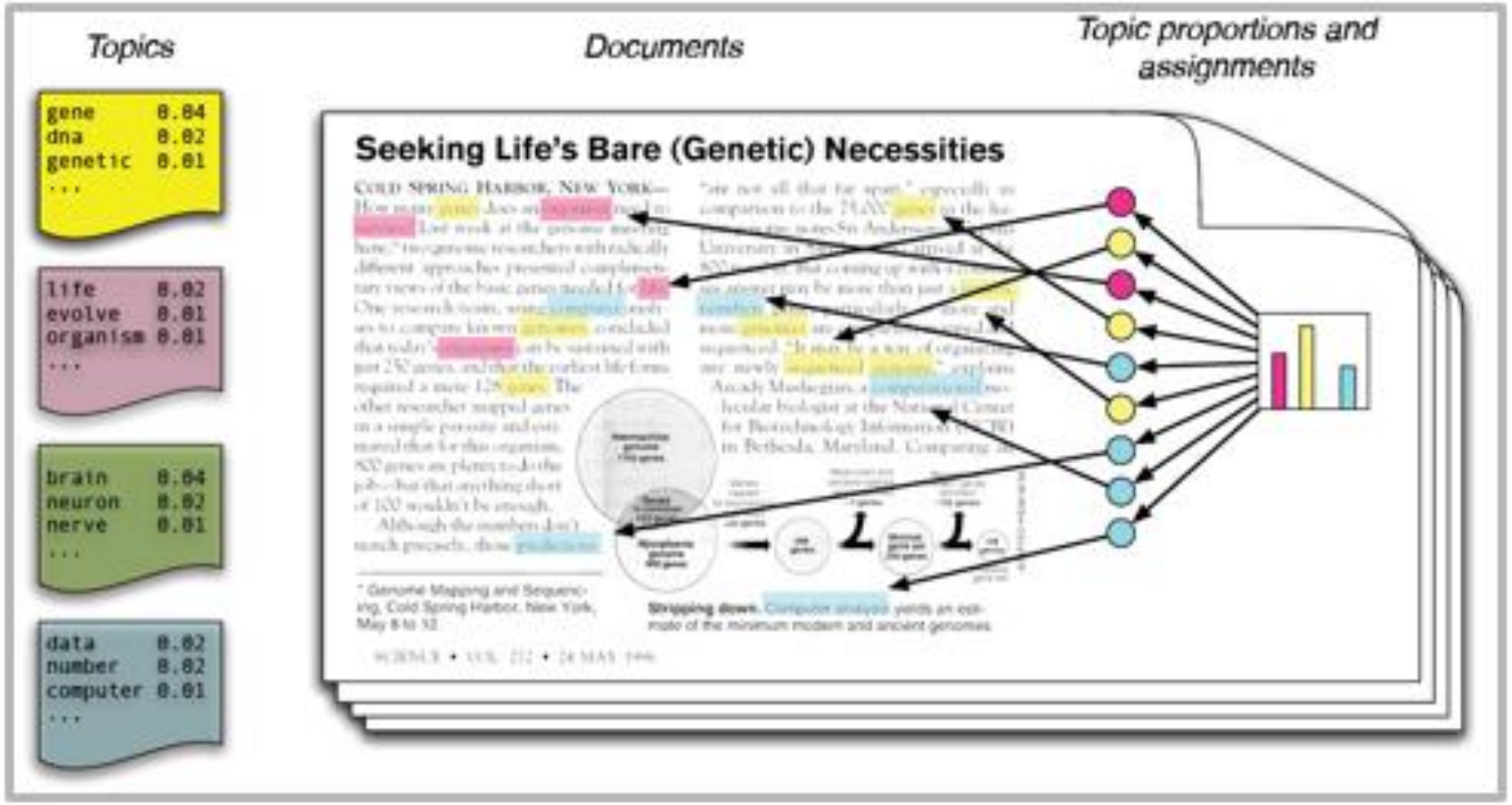
Latent Semantic Analysis (LSA)

- **What is a topic model?**

- Topic modeling helps in exploring large amounts of text data, finding clusters of words, similarity between documents, and discovering abstract topics.
- topic modeling is also used in search engines wherein the search string is matched with the results.
- Topic Models are very useful for the purpose for document clustering, organizing large blocks of textual data, information retrieval from unstructured text and feature selection.
- For Example – New York Times are using topic models to boost their user – article recommendation engines. Various professionals are using topic models for recruitment industries where they aim to extract latent features of job descriptions and map them to right candidates. They are being used to organize large datasets of emails, customer reviews, and user social media profiles.

Latent Semantic Analysis (LSA)

- What is a topic model?



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.

DATA ANALYTICS

Latent Semantic Analysis (LSA)

- **The Problem**

- Information Retrieval in the 1980s
- Given a collection of documents: retrieve documents that are relevant to a given query
- Match terms in documents to terms in query
- The vector space method
  - term (rows) by document (columns) matrix, based on occurrence
  - translate into vectors in a vector space
    - one vector for each document
  - cosine to measure distance between vectors (documents)
    - small angle = large cosine = similar
    - large angle = small cosine = dissimilar

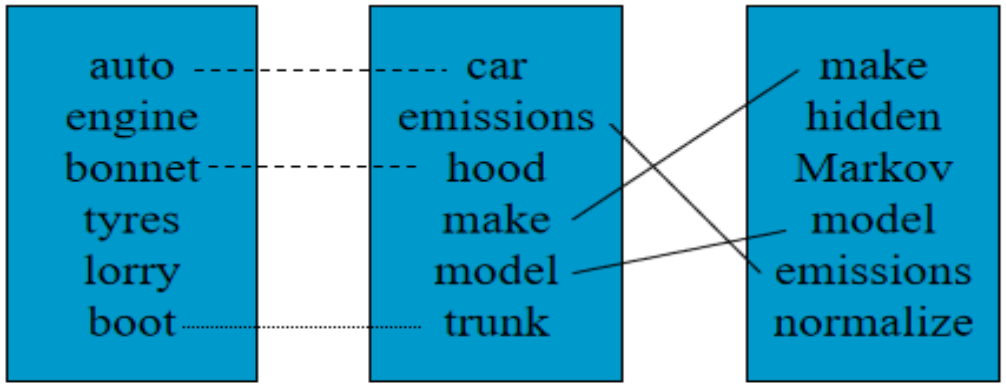
Latent Semantic Analysis (LSA)

- Standard measures in IR
  Precision: portion of selected items that the system got right
  Recall: portion of the target items that the system selected
- Two problems that arose using the vector space model:
  synonymy: many ways to refer to the same object, e.g. car and automobile
    - leads to poor recall

  polysemy: most words have more than one distinct meaning, e.g.model, python, chip
    - leads to poor precision

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)

- **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.

Generate a document-term matrix of shape m x n

Terms

Documents	T1	T2	T3	...	Tn
D1	0.2	0.1	0.5	...	0.1
D2	0.1	0.3	0.4	...	0.3
D3	0.3	0.1	0.1	...	0.5
...	...	...	...	...	...
Dm	0.2	0.1	0.2	...	0.1

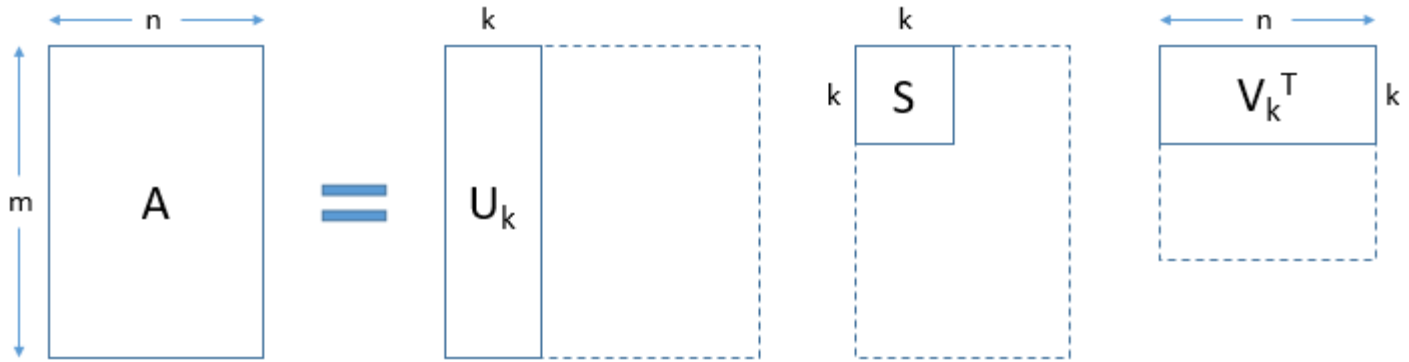
Latent Semantic Analysis (LSA)

- **Steps involved in the implementation of LSA**

Then, we will reduce the dimensions of the above matrix to k (no. of desired topics) dimensions, using singular-value decomposition (SVD).

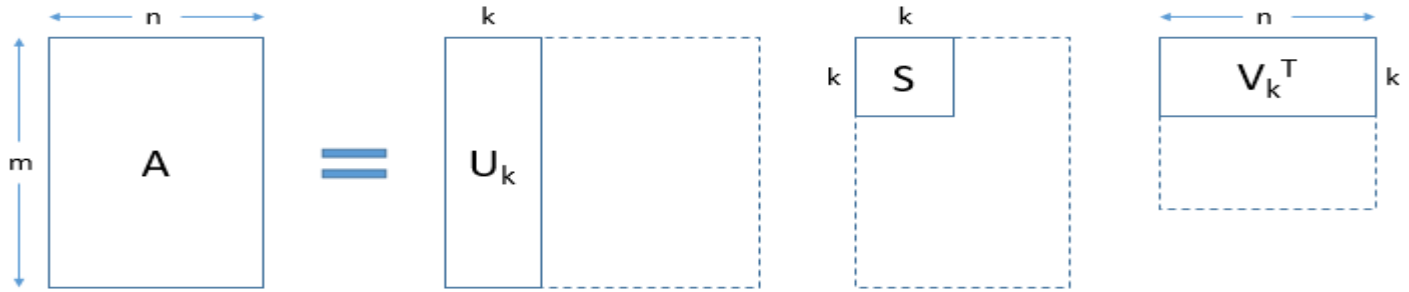
SVD decomposes a matrix into three other matrices. Suppose we want to decompose a matrix A using SVD. It will be decomposed into matrix U, matrix S, and VT (transpose of matrix V).

$$A = USV^T$$



Latent Semantic Analysis (LSA)

- **Steps involved in the implementation of LSA**



Each row of the matrix Uk (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 Vk (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.

Latent Semantic Analysis (LSA)

- **Implementation: four basic steps**

- term by document matrix (more generally term by context) tend to be sparce
- convert matrix entries to weights, typically:
  L(i,j) * G(i): local and global
  a_ij -> log(freq(a_ij)) divided by entropy for row (-sum (p logp), over p: entries in the row)
    weight directly by estimated importance in passage
    weight inversely by degree to which knowing word occurred provides information about the passage it appeared in

Latent Semantic Analysis (LSA)

- **Implementation: four basic steps**

- Rank-reduced Singular Value Decomposition (SVD) performed on matrix
  all but the k highest singular values are set to 0
  produces 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)

- **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

DATA ANALYTICS

Latent Semantic Analysis (LSA)

- **A Simple Example**

Technical Memo Titles
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

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*

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

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

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)

- A Simple Example

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

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

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

Latent Semantic Analysis (LSA)

- ## A Simple Example

LSA Titles example:
Correlations between titles in raw data

	c1	c2	c3	c4	c5	m 1	m 2	m 3
c2	- 019							
c3	0.00	0.00						
c4	0.00	0.00	0.47					
c5	- 033	0.58	0.00	- 031				
m 1	- 017	- 030	- 021	- 016	- 017			
m 2	- 026	- 045	- 032	- 024	- 026	0.67		
m 3	- 033	- 058	- 041	- 031	- 033	0.52	0.77	
m 4	- 033	- 019	- 041	- 031	- 033	- 017	0.26	0.56

0.02	
- 030	**0.44**

Correlations in first-two dimension space

c2	0.91							
c3	1.00	0.91						
c4	1.00	0.88	1.00					
c5	0.85	0.99	0.85	0.81				
m 1	- 085	- 056	- 085	- 088	- 045			
m 2	- 085	- 056	- 085	- 088	- 044	1.00		
m 3	- 085	- 056	- 085	- 088	- 044	1.00	1.00	
m 4	- 081	- 050	- 081	- 084	- 037	1.00	1.00	1.00

Latent Semantic Analysis (LSA)

- **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

PES UNIVERSITY ONLINE

THANK YOU

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