

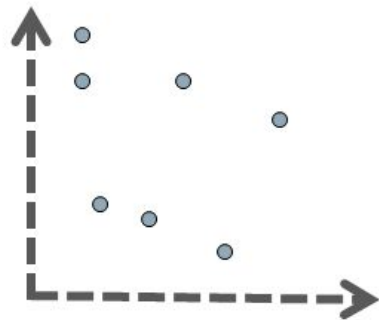
Natural Language Processing (CS5803)

Lecture 2
(Representing Text)

Recap: Drawback of TF-IDF Scheme?

- Remember?
- Solution:
 - Compact representations
 - Capturing relations between elements in consideration

Learning Representations of Words



- Represent each word with a **low-dimensional vector**
- Word similarity = vector similarity
- Key idea: Observe **surrounding words** of every word
- How do we get the vectors?
- We **learn** the vectors (or embeddings)
 - **Representation learning**
 - **Word embedding**



Use statistical cues

- Use Singular value Decomposition

$$A=U\Sigma V^T$$

- The columns of U are orthogonal eigenvectors of AA^T
- The columns of V are orthogonal eigenvectors of A^TA
- Eigenvalues $\lambda_1 \dots \lambda_r$ of AA^T are the eigenvalues of A^TA

$$\sigma_i = \sqrt{\lambda_i}$$
$$\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$$

Illustration of Singular Value Decomposition

$$\underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_A = \underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}}_U \underbrace{\begin{bmatrix} \bullet & & & & \\ & \bullet & & & \\ & & \bullet & & \\ & & & \bullet & \\ & & & & \bullet \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_{V^T}$$

Diagram illustrating the Singular Value Decomposition (SVD) of a matrix A (5x3). The matrix A is decomposed into three matrices: U (5x5), Σ (5x5), and V^T (3x3). The matrix U has a yellow shaded 5x2 submatrix. The matrix Σ has a yellow shaded 3x2 submatrix. The matrix V^T is a 3x3 matrix of stars.

$$\underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}}_A = \underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_U \underbrace{\begin{bmatrix} \bullet & & & & \\ & \bullet & & & \\ & & \bullet & & \\ & & & \bullet & \\ & & & & \bullet \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{bmatrix}}_{V^T}$$

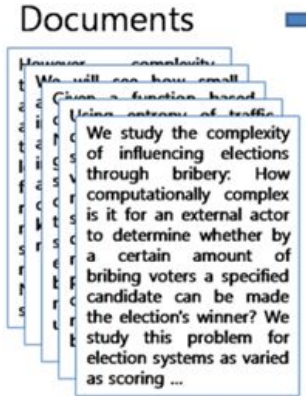
Diagram illustrating the Singular Value Decomposition (SVD) of a matrix A (3x5). The matrix A is decomposed into three matrices: U (3x3), Σ (3x3), and V^T (5x5). The matrix U is a 3x3 matrix of stars. The matrix Σ has a yellow shaded 2x2 submatrix. The matrix V^T has a yellow shaded 3x2 submatrix.

SVD for Textual Data: Summary

- A: Term-Document Incidence Matrix (Size: $m \times n$)

$$A = U \Sigma V^T$$

- Terms can be represented using the entries in U_k
 - We can work with $k \ll m$ dimensions.
- Documents can be represented using the entries in V_k
 - We can work with $k \ll n$ dimensions.



	D1	D2	D3	D4	D5
complexity	2		3	2	3
algorithm	3			4	4
entropy	1			2	
traffic		2	3		
network		1	4		

Term-document matrix

SVD on BBC Dataset

Top terms per cluster:

- **Cluster 0:** mobile, phone, broadband, digital, people, technology, phones, tv, bt, said
- **Cluster 1:** show, tv, said, series, star, musical, bbc, us, film, comedy
- **Cluster 2:** economy, growth, economic, dollar, said, rate, rates, us, year, bank
- **Cluster 3:** lord, lords, said, blunkett, blair, home, government, secretary, law, house
- **Cluster 4:** games, game, software, said, microsoft, users, people, computer, search, virus
- **Cluster 5:** music, band, album, rock, song, best, chart, number, singer, said
- **Cluster 6:** said, government, would, eu, people, uk, party, minister, public, police
- **Cluster 7:** film, best, films, oscar, festival, awards, actor, award, director, actress
- **Cluster 8:** said, company, shares, firm, us, oil, market, sales, profits, bank
- **Cluster 9:** labour, election, blair, brown, party, howard, tax, chancellor, said, tory

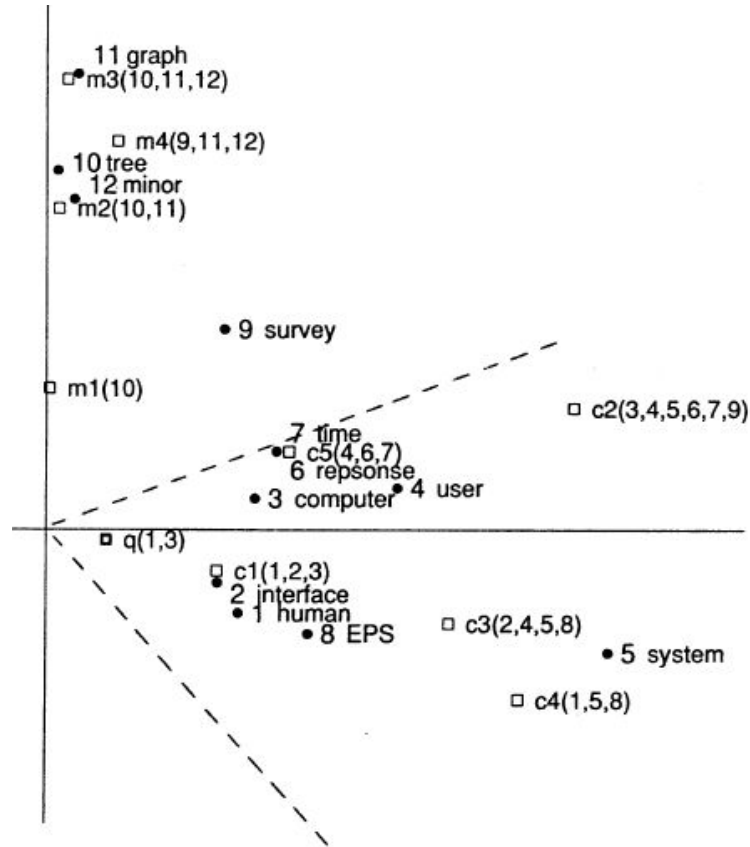
SVD Example

Technical Memo Example

Titles:

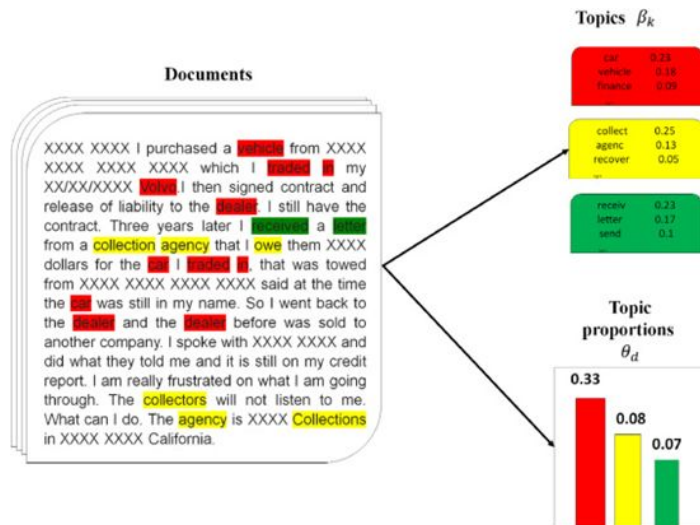
- c1: *Human machine interface for Lab ABC computer applications*
 - c2: *A survey of user opinion of computer system response time*
 - c3: *The EPS user interface management system*
 - c4: *System and human system engineering testing of EPS*
 - c5: *Relation of user-perceived response time to error measurement*
-
- m1: *The generation of random, binary, unordered 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*

SVD Example



LDA

- Built on top of the notion of topics
 - There are many topics
 - Each document is a mixture of topics
 - The topics themselves are unknown or latent
-
- Document is a collection of words
 - Which words are present, depends on the topics in the document
-
- Need a mechanism to
 - Identify the topics
 - Connect words with topics
 - Explain the documents with these learnings



LDA

Topics

gene 0.04
dna 0.02
genetic 0.01
...

life 0.02
evolve 0.01
organism 0.01
...

brain 0.04
neuron 0.02
nerve 0.01
...

data 0.02
number 0.02
computer 0.01
...

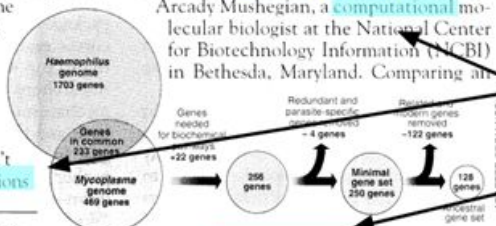
Documents

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions

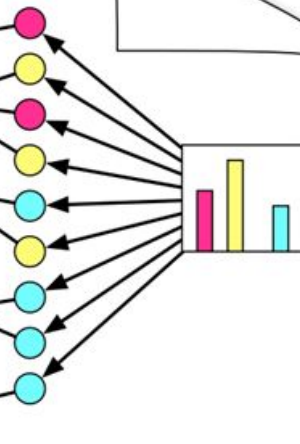
"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an



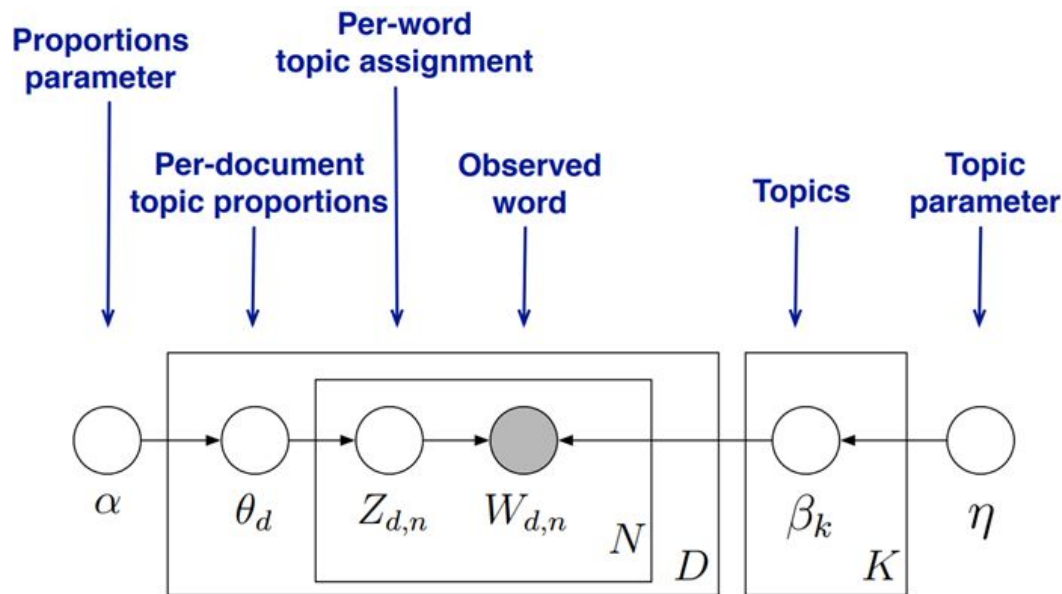
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

Topic proportions and assignments

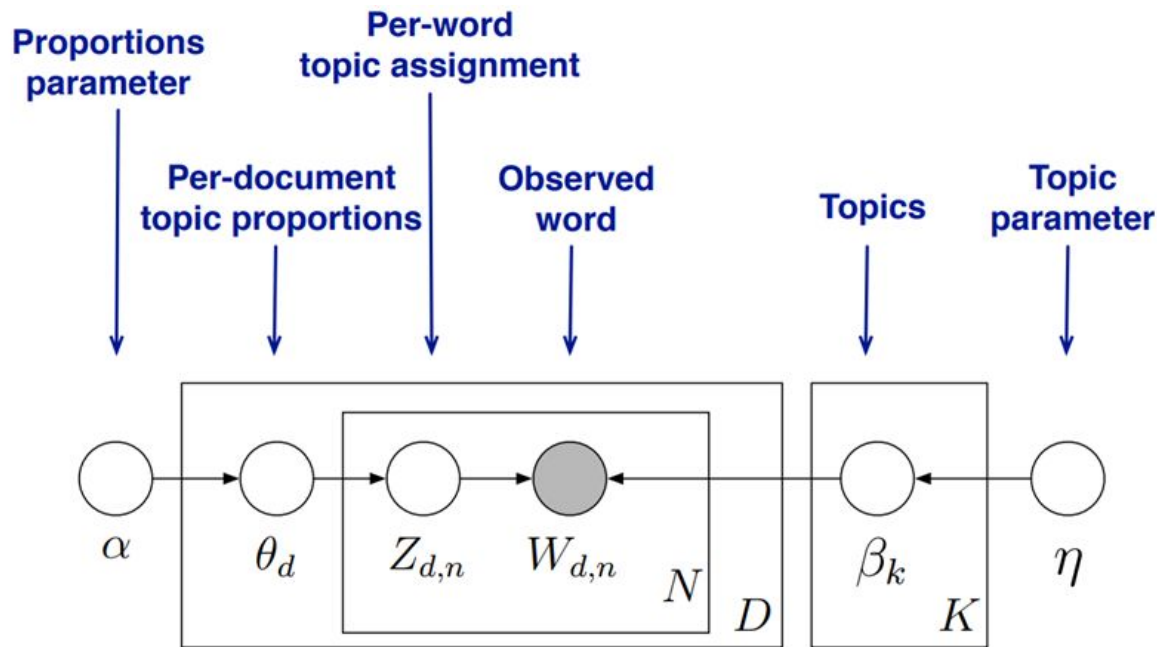


LDA



- α and η are hyperparameters
- θ , z and β are model parameters
- θ : Topic distribution in the document
- z : Per-topic word assignment
- β : Per-topic word distribution

LDA



$$p(\beta, \theta, \mathbf{z}, \mathbf{w}) = \left(\prod_{i=1}^K p(\beta_i | \eta) \right) \left(\prod_{d=1}^D p(\theta_d | \alpha) \prod_{n=1}^N p(z_{d,n} | \theta_d) p(w_{d,n} | \beta_{1:K}, z_{d,n}) \right)$$

LDA

$$p(\theta | \vec{\alpha}) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1}$$

```
import numpy as np  
def f(theta1, theta2, theta3):  
    s = np.random.dirichlet ((theta1, theta2, theta3), 5)  
    for i in range(0,len(s)):  
        print(i , ": %2.2f %2.2f %2.2f" % (s[i][0], s[i][1], s[i][2]))
```

```
f(10,10,10)  
print("---")
```

For more on generating and plotting Dirichlet Distribution, please see:

- <https://towardsdatascience.com/dirichlet-distribution-a82ab942a879>
- <https://numpy.org/doc/stable/reference/random/generated/numpy.random.dirichlet.html>

Related reading: <http://jrmeyer.github.io/machinelearning/2017/08/18/mle.html>