Latent Semantic Indexing

Seminar “Theoretical Topics in Data Science”

Vahe Eminyan

vahe.eminyan@rwth-aachen.de

20.11.2023

Overview

Motivation

- Large datasets, often organized in tabular form, represented as matrices
  - Term-document matrix representing word occurrence in documents
  - Movie-user matrix representing watched movies of users
- Interesting aspects
  - Find documents semantically associated with a query
  - Recommend a new movie to a user

	Doc 1	Doc 2	...	Doc m
Term 1	0	1	...	1
Term 2	1	0	...	1
...	...	...	...	...
Term n	1	0	...	0

Documents

Terms $\begin{pmatrix} 0 & 1 & \dots & 1 \\ 1 & 0 & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \dots & 0 \end{pmatrix}$

$n \times m$

Latent Semantic Indexing

- LSI as an information retrieval method
- Finds the latent (hidden) semantic structure of textual data
- Represent term-document matrix as product of three matrices: term-topic, topic-topic and topic-document matrix
- Answer queries with help of these matrices
- Based on singular value decomposition of the matrix

Singular Value Decomposition (SVD) [3]

- Any n by m matrix can be factored into

$$A_{n\times m} = U_{[n\times r]} D_{[r\times r]} (V_{[m\times r]})^T = \text{(orthogonal)(diagonal)(orthogonal)}.$$

- U: left singular vectors (n terms and r topics)
- V: right singular vectors (m documents and r topics)
- D: Singular values $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r$ in decreasing order ($r \times r$ diagonal matrix representing the ”importance” of each topic, where r rank of matrix A)
- Vector notation

$$A = UDV^T = \sum_{i=1}^{r} \sigma_i u_i v_i^t$$

Singular Value Decomposition (SVD) Example: Matrix A with rank $r = 3$

Terms / Documents

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} -0.48 & -0.79 & -0.11 \cdot 10^{-14} \\ -0.58 & 0.16 & 0.71 \\ \mathbf{-0.34} & \mathbf{0.56} & 0.42 \cdot 10^{-15} \\ -0.56 & 0.16 & -0.71 \end{pmatrix} \times \begin{pmatrix} 2.1 & 0 & 0 \\ 0 & 1.26 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

U: Term-Topic similarity

D: Topic "importance"

$$\times \begin{pmatrix} -0.5 & \mathbf{-0.71} & -0.5 \\ -0.5 & \mathbf{0.71} & -0.5 \\ 0.71 & 0.67 \cdot 10^{-15} & -0.711 \end{pmatrix}$$

V^T: Topic-Document similarity

Latent Semantic Indexing based on SVD

- LSI considers A_k the rank k approximation of A (I.e. keep only k most relevant topics)
- In the example $k = 2$
- Map a query to k dimensional space with U_k and then apply cosine similarity to find similar documents in $D_k V_k^T$

$$\underset{\text{Terms}}{} \underbrace{\begin{pmatrix} 1.0 & 0.01 & 1 \\ 0.51 & 1.01 & 0.51 \\ 0.0 & 1.01 & 0.0 \\ 0.49 & 0.98 & 0.49 \end{pmatrix}}_{A_k} = \underbrace{\begin{pmatrix} -0.48 & -0.79 \\ -0.58 & 0.16 \\ \mathbf{-0.34} & \mathbf{0.56} \\ -0.56 & 0.16 \end{pmatrix}}_{U_k} \times \underbrace{\begin{pmatrix} 2.1 & 0 \\ 0 & 1.26 \end{pmatrix}}_{D_k} \times \underbrace{\begin{pmatrix} -0.5 & \mathbf{-0.71} & -0.5 \\ -0.5 & \mathbf{0.71} & -0.5 \end{pmatrix}}_{V_k^T}$$

Documents

Term-Topic similarity

Topic ”importance”

Topic-Document similarity

Latent Semantic Indexing based on SVD

Theorem (Eckart and Young [1])

Among all $n \times m$ matrices C of rank at most k, A_k is the one that minimizes $\|A - C\|_F^2 = \sum_{i,j}(A_{ij} - C_{ij})^2$, where F denotes the Frobenius norm of a matrix.

$$\underset{\text{Terms}}{} \underbrace{\begin{pmatrix} 1.0 & 0.01 & 1 \\ 0.51 & 1.01 & 0.51 \\ 0.0 & 1.01 & 0.0 \\ 0.49 & 0.98 & 0.49 \end{pmatrix}}_{A_k} \approx \underbrace{\begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}}_{A} \begin{matrix} \text{Documents} \\ \text{Terms} \end{matrix}$$

Original Paper Overview and Emphasized Aspect

In this section

- Two interesting questions Papadimitriou et al. investigated [2]
  - Why does LSI perform well (why does it find the documents semantically related to each other)
  - How can we speed up the computation
- We will focus on the second question

In this section

- In this section we will investigate the question ”How we can speed up the computation”: Informal formulation of the main theorem of this section (Theorem 5 original paper)
- Introduction of theorems and lemmas that are necessary for the proof of the main theorem
- Introduction: the main theorem (Theorem 5 original paper)
- Proof of the main theorem (Theorem 5 original paper)
- Computational savings achieved by LSI by random projection

References


C. Reinsch J. H. Wilkinson.
Handbook for Automatic Computation.
Springer Berlin, Heidelberg, volume ii: linear algebra edition, 1971.


Christos H. Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh Vempala.
Latent semantic indexing: A probabilistic analysis.
Journal of Computer and System Sciences, 61(2):217–235, 2000.
URL: https://www.sciencedirect.com/science/article/pii/S0022000000917112, doi:10.1006/jcss.2000.1711.


Gilbert Strang.
Linear Algebra and Its Applications.
Cengage Learning, 4th edition edition, 2005.

References

The End