Latent Semantic Indexing: A Probabilistic Analysis

Seminar "Theoretical Topics in Data Science"

Vahe Eminyan

October 28, 2023
RWTH Aachen University

This seminar paper gives an analysis of the scientific paper "Latent Semantic Indexing: A Probabilistic Analysis" by Papadimitriou et al.
Latent semantic indexing (LSI) is one of the techniques used for information retrieval. LSI uses a mathematical technique called Singular Value Decomposition (SVD) applied to the term-document matrix. It is used to discover the hidden (latent) structure in the text data.
LSI is widely used in practice and has shown strong empirical results. However, due to the large size of term-document matrices, the computation can be slow. These lead to two interesting questions: why does LSI perform so well, and how can we speed up computations? Papadimitriou et al. address both of these questions. They provide a theoretical justification for LSI's effectiveness, considering a special type of term-document matrix.
Additionally, to expedite computations, they suggest initially mapping the large matrices into low-dimensional spaces using random projections and then applying LSI to this reduced-dimensional projection. They prove that the matrix obtained via random projection followed by LSI recovers almost as much as the matrix obtained by direct LSI, with high probability.
In this seminar paper, we will consider all relevant information, providing a detailed analysis of LSI by random projection.

1 Introduction

Retrieving information from given data has been an important aspect for a long time. Due to the development of information systems and the Internet, huge amounts of data are available. Handling these huge amounts of data is a great challenge, so many techniques have been developed to cope with the work process. Consider a scenario where we have a dataset comprising millions of documents structured as a term-document matrix and a user submits a query. Instead of just finding the documents that include the words of the query, the goal is to find documents that align with the query semantically. This semantic understanding ensures more accurate and relevant search results, even in extensive datasets. Latent Semantic Indexing (LSI) is one of the information retrieval techniques. It uses Singular Value Decomposition (SVD) to represent the term-document matrix as a product of three matrices. By representing the term-document matrix in such a way we want to find the underlying (latent, hidden) semantical topics (also called concepts) of the term-document matrix. With the help of such decomposition, we can map the documents and queries to a lower dimensional space and compare them not only syntactically but also semantically. (Section 2 provides more detailed information about LSI and SVD).
LSI has shown strong empirical results. However, prior to the paper of Papadimitriou et al., there was no satisfactory explanation for its success (Why does LSI find the documents corresponding to the semantics of the query with high accuracy). They prove a theorem that under certain constraints LSI will always find the correct topic of the given query with high probability. Section 4 of this seminar paper elaborates on this theorem.
Despite its effectiveness, the computational time of LSI is very long. Papadimitriou et al. introduce a two-step LSI, in which we first map the original term-document matrix into a lower dimensional space by using random projections and then apply LSI. The paper proves that the matrix obtained via random projection followed by LSI recovers almost as much as the matrix obtained by direct LSI, with high probability. Section 5 provides a detailed analysis of LSI by random projection and presents the proof for this statement.
In the last section, we will draw a conclusion and give a short summary of this seminar paper.

2 LSI Background

The corpus is defined as a set of documents. Each document is a vector of length n from $\mathbb{R}^n$. Each position of the document describes a mathematical function in terms of ith term of the entire term space. The function can, for example, be the frequency of the ith term, or just 0-1 (1 if the term appears in the document, 0 otherwise).
Let $A \in \mathbb{R}^{n \times m}$ be a matrix of rank r. The Singular Value Decomposition (SVD) is a mathematical technique that represents the matrix as a product of three matrices.

$$A = UDV^T.$$

Let $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r$ be the eigenvalues of the matrix AA^T (called singular values of A), then $D = diag(\sigma_1, ..., \sigma_r) \in \mathbb{R}^{r \times r}$. $U \in \mathbb{R}^{n \times r}$ is the matrix representing the eigenvectors of the matrix AA^T and $V \in \mathbb{R}^{m \times r}$ the eigenvectors of the matrix A^TA. The columns of U and V are orthonormal. It is shown that every matrix can be represented in such a form [ZITATTTTT]
In LSI the matrix $A \in \mathbb{R}^{n \times m}$ represents a term-document matrix of n terms and m documents. Each row of the matrix corresponds to a term and each column to a document.
LSI keeps only k-largest singular values of the matrix considering only the first k columns of matrices U and V. I.e.

$$A_k = U_k D_k V_k^T$$

which is an approximation of the original matrix A. In order to find the most similar documents for the submitted query we map the query to the k-dimensional space by using the matrix U_k. Afterward, we compare the k-dimensional projection with k-dimensional representations of the documents in the corpus. For this comparison, we can use cosine similarity. Thus we only need to store the three matrices U_k, D_k, and V_k. At this point, we can see the saving of the storage. In order to save the matrix A entirely we need to memorize $n \cdot m$ entries. In contrast to that, we need only $n \cdot k + k \cdot k + m \cdot k$ entries, which is much smaller than $n \cdot m$, because k is much smaller than both n and m.
The approximation A_k of rank k is the matrix that minimizes the Frobenius norm of the of $A - A_k$. Formally:

Theorem 1 (ZITATTT)**.** *Among all $n \times m$ matrices C of rank at most k, A_k is the one that minimizes $||A - C||_F^2 = \sum_{i,j}(A_{ij} - C_{ij})^2$.*

I.e. this approximation preserves all relative distances, now we only need to say why it finds queries and documents that are semantically related.

3 Probabilistic Corpus Model

Because LSI relies on uncovering the statistical characteristics of a corpus, it is essential to begin with a well-defined probabilistic model of the corpus. In this section, we introduce the probabilistic corpus model. In the rest of the paper, we assume that the corpus is generated from a corpus model.

Definition 2. The universe U is the set of all terms.

Definition 3. A topic is a probability distribution on U.

The logical interpretation of this definition is the following. Assume we have a set of documents where one partition is about computer science and the other is about nature. Hence there would be two hidden topics for the whole set of documents. Hence the one topic will probably include terms like computer, software and algorithm, and the second will include the terms forest, ocean and earth. Thus we can interpret a topic as a probability distribution on the entire set of the terms.

Another important aspect is the authorship style, it influences the frequencies of words and hence the documents.

Definition 4. A style is a $|U| \times |U|$ stochastic matrix (a matrix with nonnegative entries and row sums equal to 1), denoting the way whereby style modifies the frequency of terms.

After introducing the previous three definitions, now we can define the probabilistic corpus model.

Definition 5. A corpus model $\mathscr{C}$ is a quadruple $\mathscr{C} = (U, \mathscr{T}, \mathscr{S}, D)$, where U is the universe of terms, $\mathscr{T}$ is a set of topics, and $\mathscr{S}$ is a set of styles, and D a probability distribution on $\hat{\mathscr{T}} \times \hat{\mathscr{S}} \times \mathbb{Z}^+$, where by $\hat{\mathscr{T}}$ we denote the set of all convex combinations of topics in $\mathscr{T}$, by $\hat{\mathscr{S}}$ the set of all convex combinations of styles in $\mathscr{S}$, and by $\mathbb{Z}^+$ the set of positive integers (the integers represent the lengths of documents).

I.e. a corpus generated from the introduced corpus model is a set of documents in which every document is generated as follows: we sample a triple from D. It includes fixed $\hat{T} \in \mathscr{T}$, $\hat{S} \in \mathscr{S}$, and a fixed length ℓ of the document. Then we use $\hat{T}$ to sample the terms of the document ℓ times.

4 Brief Mention of Analysis of LSI's Good Performance

In this part, I will provide the most significant information of paragraph 4 of the original paper.
Main idea: Under certain conditions and assumptions, it can be shown why LSI performs well.

5 LSI by Random Projection

In order to reduce the computational time we use dimensionality reduction and then apply LSI. We prove theorem 5 of the paper which maintains that the matrix obtained via random projection followed by LSI recovers almost as much as the matrix obtained by direct LSI, with high probability.

6 Conclusion and Summary

In this section, we will draw conclusions based on the findings presented in the seminar paper and give a brief summary of the paper.

References