15-826: Multimedia Databases and Data Mining

Lecture #17: Text - part IV (LSI)

C. Faloutsos

Must-read Material


Outline

Goal: ‘Find similar / interesting things’
- Intro to DB
- Indexing - similarity search
- Data Mining

Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
- fractals
- text
- SVD: a powerful tool
- multimedia
- ...
Text - Detailed outline

• text
  – problem
  – full text scanning
  – inversion
  – signature files
  – clustering
  – information filtering and LSI

LSI - Detailed outline

• LSI
  – problem definition
  – main idea
  – experiments

Information Filtering + LSI

• [Foltz+, ’92] Goal:
  – users specify interests (= keywords)
  – system alerts them, on suitable news-documents
• Major contribution: LSI = Latent Semantic Indexing
  – latent (‘hidden’) concepts

Main idea
• map each document into some ‘concepts’
• map each term into some ‘concepts’

‘Concept’ := a set of terms, with weights, e.g.
  – “data” (0.8), “system” (0.5), “retrieval” (0.6) -
    > DBMS_concept
Information Filtering + LSI

Pictorially: term-document matrix (BEFORE)

<table>
<thead>
<tr>
<th></th>
<th>'data'</th>
<th>'system'</th>
<th>'retrieval'</th>
<th>'lung'</th>
<th>'ear'</th>
</tr>
</thead>
<tbody>
<tr>
<td>TR1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TR2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TR3</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>TR4</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Information Filtering + LSI

Pictorially: concept-document matrix and...

<table>
<thead>
<tr>
<th></th>
<th>'DBMS-concept'</th>
<th>'medical-concept'</th>
</tr>
</thead>
<tbody>
<tr>
<td>TR1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>TR2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>TR3</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>TR4</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

Information Filtering + LSI

... and concept-term matrix

<table>
<thead>
<tr>
<th></th>
<th>'DBMS-concept'</th>
<th>'medical-concept'</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>system</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>retrieval</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>lung</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ear</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Information Filtering + LSI

Q: How to search, eg., for ‘system’?
Information Filtering + LSI

A: find the corresponding concept(s); and the corresponding documents

<table>
<thead>
<tr>
<th>Concept</th>
<th>'DBMS-concept'</th>
<th>'medical-concept'</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>system</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>retrieval</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>lung</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>ear</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Thus it works like an (automatically constructed) thesaurus:
we may retrieve documents that DON’T have the term ‘system’, but they contain almost everything else (‘data’, ‘retrieval’)

LSI - Detailed outline

- LSI
  - problem definition
  - main idea
  - experiments
LSI - Experiments

- 150 Tech Memos (TM) / month
- 34 users submitted ‘profiles’ (6-66 words per profile)
- 100-300 concepts

LSI - Experiments

- four methods, cross-product of:
  - vector-space or LSI, for similarity scoring
  - keywords or document-sample, for profile specification
- measured: precision/recall

\[
\begin{pmatrix}
\text{data, retrieval} \\
\text{concept1, concept2}
\end{pmatrix}
\times
\begin{pmatrix}
\text{data} \\
\text{mining}
\end{pmatrix}
\]

LSI - Experiments

- Q: Who wins?
- precision
- recall

LSI - Experiments

- LSI, with document-based profiles, were better
- precision
- recall
LSI - Discussion - Conclusions

• Great idea,
  – to derive ‘concepts’ from documents
  – to build a ‘statistical thesaurus’ automatically
  – to reduce dimensionality
• Often leads to better precision/recall
• but:
  – Needs ‘training’ set of documents
  – ‘concept’ vectors are not sparse anymore

Observations
• Bellcore (→ Telcordia) has a patent
• used for multi-lingual retrieval

How exactly SVD works? (Details, next)