Database Content Exploration and Exploratory Analysis of User Queries

Orestis Gkorgkas

Norwegian University of Science and Technology

August 12, 2015
Database Content Exploration

- When users visit a database they often do not have a specific item in mind but only a vague idea about it
  - Bob Dylan’s rock albums
  - An economic phone with large storage, good camera and long battery life
- Vast amount of data often overwhelm the users
- Users have difficulty locating items that suit their needs
Preference Queries in Exploratory Search

- Preferences are expressed using keywords or by indicating interest on specific features
- Preference queries can be ambiguous
  - Large search space
  - Numerous results → User confusion
- Implicit restrictions
  - Search space is limited to a single category of items
- Preference queries often do not provide a wide overview of the available information
Impact to visibility

- Users need to pose multiple queries
  - De-motivating experience
  - Users quit trying
- The abundance of data prevents users from viewing all options
  - Preference queries cannot address this problem in its entirety
- The number of available options reduces the visibility of the database content
- Need to enhance the visibility of the database objects
  - Help and motivate users explore and understand the available content
Research focus

- **Database content exploration techniques**
  - Organize the search results and present to the user a coherent overview of the available options

- **Exploratory analysis of user queries**
  - Analyze user queries to find items or features with high potential impact
  - Interesting items can motivate users to explore the database and give them an initial insight of the available content
    - Items that are constantly highly ranked for many users
    - Items that are attractive to users with diverse preferences
  - Identify features that will increase the visibility of a service or a company
Research Questions

Enhancement of the visibility of the objects stored in large databases through exploratory search and analysis

RQ 1 How can we summarize keyword search results on structured data and provide users a coherent overview of the information relevant to the query?

RQ 2 How can we efficiently explore a large collection of combinable objects?

RQ 3 How can we efficiently identify objects that are constantly highly ranked by a large number of users over a specific time-period?

RQ 4 How can we efficiently identify groups of products that attract customers with diverse preferences?

RQ 5 How can a description of a service or product be enriched to improve its visibility?
Grouping and summarizing keyword search results

RQ 1
How can we summarize keyword search results on structured data and provide users a coherent overview of the information relevant to the query?
Keyword Search in Structured Data

### Person

<table>
<thead>
<tr>
<th>id</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Ben Affleck</td>
</tr>
<tr>
<td>p2</td>
<td>Matt Damon</td>
</tr>
</tbody>
</table>

### Movie

<table>
<thead>
<tr>
<th>id</th>
<th>title</th>
</tr>
</thead>
<tbody>
<tr>
<td>m1</td>
<td>Good Will Hunting</td>
</tr>
<tr>
<td>m2</td>
<td>Dogma</td>
</tr>
<tr>
<td>m3</td>
<td>Project Greenlight</td>
</tr>
</tbody>
</table>

### Joined Trees of Tuples (JTT)

#### as Actor

<table>
<thead>
<tr>
<th>aid</th>
<th>pid</th>
<th>mid</th>
</tr>
</thead>
<tbody>
<tr>
<td>a1</td>
<td>p1</td>
<td>m1</td>
</tr>
<tr>
<td>a2</td>
<td>p1</td>
<td>m2</td>
</tr>
<tr>
<td>a3</td>
<td>p2</td>
<td>m1</td>
</tr>
<tr>
<td>a4</td>
<td>p2</td>
<td>m2</td>
</tr>
</tbody>
</table>

#### as Writer

<table>
<thead>
<tr>
<th>wid</th>
<th>pid</th>
<th>mid</th>
</tr>
</thead>
<tbody>
<tr>
<td>w1</td>
<td>p1</td>
<td>m1</td>
</tr>
<tr>
<td>w2</td>
<td>p2</td>
<td>m1</td>
</tr>
</tbody>
</table>

#### as Producer

<table>
<thead>
<tr>
<th>prid</th>
<th>pid</th>
<th>mid</th>
</tr>
</thead>
<tbody>
<tr>
<td>pr1</td>
<td>p1</td>
<td>m3</td>
</tr>
<tr>
<td>pr2</td>
<td>p2</td>
<td>m3</td>
</tr>
</tbody>
</table>

**Keyword query**

“Ben Affleck Matt Damon”
Keyword Search in Structured Data

- Problems with keyword queries
  - Ambiguity
  - Numerous results of different structure provide little insight to the users
- Highlight the key information and help users understand the results
  - Organize the results into groups
  - Provide a summary with the most important terms of each group
Using time aware attributes to organize the results

Frequently data are connected with time-aware attributes

- Combine the temporal and the textual data to group the data search results in thematic periods
- Create a summary for each period
- Present the user summarized thematic periods related to his/her query
Finding the age of a Joined Tree of Tuples

Query

“Ben Affleck, Matt Damon”
Finding the age of a Joined Tree of Tuples

Query
“Ben Affleck, Matt Damon”
### Organizing the search results

<table>
<thead>
<tr>
<th>Movies</th>
<th>Plays</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_1$:</td>
<td>$m_1$</td>
<td>$a_1$</td>
</tr>
<tr>
<td>$T_2$:</td>
<td>$m_2$</td>
<td>$a_1$</td>
</tr>
<tr>
<td>$T_3$:</td>
<td>$m_3$</td>
<td>$a_1$</td>
</tr>
<tr>
<td>$T_4$:</td>
<td>$m_4$</td>
<td>$a_2$</td>
</tr>
<tr>
<td>$T_5$:</td>
<td>$m_5$</td>
<td>$a_2$</td>
</tr>
<tr>
<td>$T_6$:</td>
<td>$m_6$</td>
<td>$a_2$</td>
</tr>
<tr>
<td>$T_7$:</td>
<td>$m_7$</td>
<td>$a_3$</td>
</tr>
<tr>
<td>$T_8$:</td>
<td>$m_8$</td>
<td>$a_3$</td>
</tr>
</tbody>
</table>

**Query**

"Woody Allen female actors"
Organizing the search results

<table>
<thead>
<tr>
<th>Movies</th>
<th>Plays</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>T&lt;sub&gt;1&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;1&lt;/sub&gt; Annie Hall</td>
<td>drama 1977 W.Allen</td>
<td>m&lt;sub&gt;1&lt;/sub&gt; a&lt;sub&gt;1&lt;/sub&gt; a&lt;sub&gt;1&lt;/sub&gt; D. Keaton female 1946</td>
</tr>
<tr>
<td><strong>T&lt;sub&gt;2&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;2&lt;/sub&gt; Interiors</td>
<td>drama 1978 W.Allen</td>
<td>m&lt;sub&gt;2&lt;/sub&gt; a&lt;sub&gt;1&lt;/sub&gt; a&lt;sub&gt;1&lt;/sub&gt; D. Keaton female 1946</td>
</tr>
<tr>
<td><strong>T&lt;sub&gt;3&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;3&lt;/sub&gt; Manhattan</td>
<td>drama 1979 W.Allen</td>
<td>m&lt;sub&gt;3&lt;/sub&gt; a&lt;sub&gt;1&lt;/sub&gt; a&lt;sub&gt;1&lt;/sub&gt; D. Keaton female 1946</td>
</tr>
<tr>
<td><strong>T&lt;sub&gt;4&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;4&lt;/sub&gt; Broadway</td>
<td>Danny Rose comedy 1984 W.Allen</td>
<td>m&lt;sub&gt;4&lt;/sub&gt; a&lt;sub&gt;2&lt;/sub&gt; a&lt;sub&gt;3&lt;/sub&gt; M. Farrow female 1945</td>
</tr>
<tr>
<td><strong>T&lt;sub&gt;5&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;5&lt;/sub&gt; Cairo</td>
<td>comedy 1985 W.Allen</td>
<td>m&lt;sub&gt;5&lt;/sub&gt; a&lt;sub&gt;2&lt;/sub&gt; a&lt;sub&gt;3&lt;/sub&gt; M. Farrow female 1945</td>
</tr>
<tr>
<td><strong>T&lt;sub&gt;6&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;6&lt;/sub&gt; Hannah</td>
<td>her Sisters comedy 1986 W.Allen</td>
<td>m&lt;sub&gt;6&lt;/sub&gt; a&lt;sub&gt;2&lt;/sub&gt; a&lt;sub&gt;3&lt;/sub&gt; M. Farrow female 1945</td>
</tr>
<tr>
<td><strong>T&lt;sub&gt;7&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;7&lt;/sub&gt; Deconstructing</td>
<td>Harry comedy 1997 W.Allen</td>
<td>m&lt;sub&gt;7&lt;/sub&gt; a&lt;sub&gt;3&lt;/sub&gt; a&lt;sub&gt;3&lt;/sub&gt; J. Davis female 1945</td>
</tr>
<tr>
<td><strong>T&lt;sub&gt;8&lt;/sub&gt;</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m&lt;sub&gt;7&lt;/sub&gt; Celebrity</td>
<td>comedy 1998 W.Allen</td>
<td>m&lt;sub&gt;8&lt;/sub&gt; a&lt;sub&gt;3&lt;/sub&gt; a&lt;sub&gt;3&lt;/sub&gt; J. Davis female 1945</td>
</tr>
</tbody>
</table>

Query

“Woody Allen female actors”
Organizing the search results

<table>
<thead>
<tr>
<th>Movies</th>
<th>Plays</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>T_1</td>
<td>m_1</td>
<td>Annie Hall</td>
</tr>
<tr>
<td>T_2</td>
<td>m_2</td>
<td>Interiors</td>
</tr>
<tr>
<td>T_3</td>
<td>m_3</td>
<td>Manhattan</td>
</tr>
<tr>
<td>T_4</td>
<td>m_4</td>
<td>Broadway Danny Rose</td>
</tr>
<tr>
<td>T_5</td>
<td>m_5</td>
<td>The Purple Rose of Cairo</td>
</tr>
<tr>
<td>T_6</td>
<td>m_6</td>
<td>Hannah and her Sisters</td>
</tr>
<tr>
<td>T_7</td>
<td>m_7</td>
<td>Deconstructing Harry</td>
</tr>
<tr>
<td>T_8</td>
<td>m_7</td>
<td>Celebrity</td>
</tr>
</tbody>
</table>

Query

“Woody Allen female actors”
Organizing the search results

<table>
<thead>
<tr>
<th>Movies</th>
<th>Plays</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>T_1 : m_1 Annie Hall drama 1977 W.Allen</td>
<td>m_1 a_1</td>
<td>a_1 D. Keaton female 1946</td>
</tr>
<tr>
<td>T_2 : m_2 Interiors drama 1978 W.Allen</td>
<td>m_2 a_1</td>
<td>a_1 D. Keaton female 1946</td>
</tr>
<tr>
<td>T_3 : m_3 Manhattan drama 1979 W.Allen</td>
<td>m_3 a_1</td>
<td>a_1 D. Keaton female 1946</td>
</tr>
<tr>
<td>T_4 : m_4 Broadway Danny Rose comedy 1984 W.Allen</td>
<td>m_4 a_2</td>
<td>a_3 M. Farrow female 1945</td>
</tr>
<tr>
<td>T_5 : m_5 The Purple Rose of Cairo comedy 1985 W.Allen</td>
<td>m_5 a_2</td>
<td>a_3 M. Farrow female 1945</td>
</tr>
<tr>
<td>T_6 : m_6 Hannah and her Sisters comedy 1986 W.Allen</td>
<td>m_6 a_2</td>
<td>a_3 M. Farrow female 1945</td>
</tr>
<tr>
<td>T_7 : m_7 Deconstructing Harry comedy 1997 W.Allen</td>
<td>m_7 a_3</td>
<td>a_3 J. Davis female 1945</td>
</tr>
<tr>
<td>T_8 : m_7 Celebrity comedy 1998 W.Allen</td>
<td>m_8 a_3</td>
<td>a_3 J. Davis female 1945</td>
</tr>
</tbody>
</table>

Query

“Woody Allen female actors”
## Organizing the search results

<table>
<thead>
<tr>
<th>Movies</th>
<th>Plays</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>m₁ Annie Hall drama 1977 W.Allen</td>
<td>m₁ a₁</td>
<td>a₁ D. Keaton female 1946</td>
</tr>
<tr>
<td>m₂ Interiors drama 1978 W.Allen</td>
<td>m₂ a₁</td>
<td>a₁ D. Keaton female 1946</td>
</tr>
<tr>
<td>m₃ Manhattan drama 1979 W.Allen</td>
<td>m₃ a₁</td>
<td>a₁ D. Keaton female 1946</td>
</tr>
<tr>
<td>m₄ Broadway Danny Rose comedy 1984 W.Allen</td>
<td>m₄ a₂</td>
<td>a₃ M. Farrow female 1945</td>
</tr>
<tr>
<td>m₅ The Purple Rose of Cairo comedy 1985 W.Allen</td>
<td>m₅ a₂</td>
<td>a₃ M. Farrow female 1945</td>
</tr>
<tr>
<td>m₆ Hannah and her Sisters comedy 1986 W.Allen</td>
<td>m₆ a₂</td>
<td>a₃ M. Farrow female 1945</td>
</tr>
<tr>
<td>m₇ Deconstructing Harry comedy 1997 W.Allen</td>
<td>m₇ a₃</td>
<td>a₃ J. Davis female 1945</td>
</tr>
<tr>
<td>m₇ Celebrity comedy 1998 W.Allen</td>
<td>m₈ a₃</td>
<td>a₃ J. Davis female 1945</td>
</tr>
</tbody>
</table>

### Query

“Woody Allen female actors”
## Organizing the search results

<table>
<thead>
<tr>
<th>Movies</th>
<th>Plays</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_1$ : $m_1$ Annie Hall drama 1977 W.Allen</td>
<td>$m_1$ $a_1$</td>
<td>$a_1$ D. Keaton female 1946</td>
</tr>
<tr>
<td>$T_2$ : $m_2$ Interiors drama 1978 W.Allen</td>
<td>$m_2$ $a_1$</td>
<td>$a_1$ D. Keaton female 1946</td>
</tr>
<tr>
<td>$T_3$ : $m_3$ Manhattan drama 1979 W.Allen</td>
<td>$m_3$ $a_1$</td>
<td>$a_1$ D. Keaton female 1946</td>
</tr>
<tr>
<td>$T_4$ : $m_4$ Broadway Danny Rose comedy 1984 W.Allen</td>
<td>$m_4$ $a_2$ $a_3$ M. Farrow female 1945</td>
<td></td>
</tr>
<tr>
<td>$T_5$ : $m_5$ The Purple Rose of Cairo comedy 1985 W.Allen</td>
<td>$m_5$ $a_2$ $a_3$ M. Farrow female 1945</td>
<td></td>
</tr>
<tr>
<td>$T_6$ : $m_6$ Hannah and her Sisters comedy 1986 W.Allen</td>
<td>$m_6$ $a_2$ $a_3$ M. Farrow female 1945</td>
<td></td>
</tr>
<tr>
<td>$T_7$ : $m_7$ Deconstructing Harry comedy 1997 W.Allen</td>
<td>$m_7$ $a_3$ $a_3$ J. Davis female 1945</td>
<td></td>
</tr>
<tr>
<td>$T_8$ : $m_7$ Celebrity comedy 1998 W.Allen</td>
<td>$m_8$ $a_3$ $a_3$ J. Davis female 1945</td>
<td></td>
</tr>
</tbody>
</table>

### Query

“Woody Allen female actors”
## Group summaries

- **Summary for size=2**

### 1977-1979 Summary

- **genre:** drama
- **actor:** D. Keaton

<p>| | | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_1$:</td>
<td>$m_1$</td>
<td>Annie Hall</td>
<td>drama</td>
<td>1977</td>
<td>W. Allen</td>
<td>$m_1$</td>
<td>$a_1$</td>
<td>$a_1$</td>
<td>D. Keaton</td>
<td>female</td>
<td>1946</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$T_2$:</td>
<td>$m_2$</td>
<td>Interiors</td>
<td>drama</td>
<td>1978</td>
<td>W. Allen</td>
<td>$m_2$</td>
<td>$a_1$</td>
<td>$a_1$</td>
<td>D. Keaton</td>
<td>female</td>
<td>1946</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$T_3$:</td>
<td>$m_3$</td>
<td>Manhattan</td>
<td>drama</td>
<td>1979</td>
<td>W. Allen</td>
<td>$m_3$</td>
<td>$a_1$</td>
<td>$a_1$</td>
<td>D. Keaton</td>
<td>female</td>
<td>1946</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Group coherence evaluation for each query

<table>
<thead>
<tr>
<th>Query</th>
<th>Average group coherence</th>
<th>usefulness</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Daniel Craig” movies</td>
<td>1.50</td>
<td>0.75</td>
</tr>
<tr>
<td>“James Bond” movies</td>
<td>1.50</td>
<td>0.90</td>
</tr>
<tr>
<td>“James Bond” male actors</td>
<td>1.50</td>
<td>0.75</td>
</tr>
<tr>
<td>“Woody Allen” female actors</td>
<td>1.27</td>
<td>0.82</td>
</tr>
<tr>
<td>“Clint Eastwood” movies</td>
<td>1.50</td>
<td>0.86</td>
</tr>
<tr>
<td>“Peter Jackson” male actors</td>
<td>1.75</td>
<td>1.00</td>
</tr>
<tr>
<td>“Denzel Washington” Action</td>
<td>1.88</td>
<td>1.00</td>
</tr>
<tr>
<td>“Julia Roberts” movies</td>
<td>1.75</td>
<td>0.88</td>
</tr>
<tr>
<td>“Kevin Spacey” drama</td>
<td>1.27</td>
<td>0.82</td>
</tr>
<tr>
<td>“Jack Nicholson” female actors</td>
<td>1.44</td>
<td>0.89</td>
</tr>
<tr>
<td>“Al Pacino” male actors</td>
<td>1.50</td>
<td>0.88</td>
</tr>
<tr>
<td>“Al Pacino” directors</td>
<td>1.50</td>
<td>1.00</td>
</tr>
<tr>
<td>“Stanley Kubrick” actors</td>
<td>1.75</td>
<td>1.00</td>
</tr>
<tr>
<td>“Stanley Kubrick” movies</td>
<td>1.60</td>
<td>0.80</td>
</tr>
<tr>
<td>“Robert De Niro” directors</td>
<td>1.60</td>
<td>0.90</td>
</tr>
</tbody>
</table>

- Users evaluate if the periods detected are meaningful.
- The choices are: 0(no coherence), 1(poor coherence), 2(strong coherence)
- The values for usefulness are: 0(not useful), 1(useful)
Exploratory top-$k$ join queries

RQ 2
How can we efficiently explore a large collection of combinable objects?
Exploratory search in item collections

Desired features
- processor
- memory
- battery life
- storage
- weight
- low price
Top-$k$ queries

\[ w \left( o \right) = \sum_{i=1}^{d} \left[ w \left[ i \right] \right] o \left[ i \right] \]
Top-\(k\) queries

\[
f_w(o) = \sum_{i=1}^{d} w[i] o[i]
\]
Top-k queries

\[ f_w(o) = \sum_{i=1}^{d} w[i] o[i] \]
Top-$k$ queries

A \bullet B \bullet C \bullet D \bullet E \bullet F

\[ f_w(o) = \sum_{i=1}^{d} w[i] o[i] \]
Top-$k$ queries

\[ f_w(o) = \sum_{i=1}^{d} w[i] o[i] \]
Processing and organizing object combinations

### Laptops

<table>
<thead>
<tr>
<th>id</th>
<th>CPU score</th>
<th>RAM (GB)</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>battery (hours)</th>
<th>weight (kg)</th>
<th>RAM type</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>c₁</td>
<td>3346</td>
<td>4</td>
<td>0</td>
<td>539</td>
<td>4</td>
<td>2.7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>c₂</td>
<td>3346</td>
<td>8</td>
<td>256</td>
<td>1119</td>
<td>6.5</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>c₃</td>
<td>3941</td>
<td>12</td>
<td>256</td>
<td>1199</td>
<td>8</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c₄</td>
<td>3997</td>
<td>8</td>
<td>0</td>
<td>1348</td>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

### Memory

<table>
<thead>
<tr>
<th>mid</th>
<th>RAM (GB)</th>
<th>price (USD)</th>
<th>RAM type</th>
</tr>
</thead>
<tbody>
<tr>
<td>m₁</td>
<td>4</td>
<td>71</td>
<td>2</td>
</tr>
<tr>
<td>m₂</td>
<td>4</td>
<td>45</td>
<td>1</td>
</tr>
<tr>
<td>m₃</td>
<td>8</td>
<td>81</td>
<td>1</td>
</tr>
</tbody>
</table>

### SSD

<table>
<thead>
<tr>
<th>sid</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>d₁</td>
<td>240</td>
<td>143</td>
<td>1</td>
</tr>
<tr>
<td>d₂</td>
<td>512</td>
<td>300</td>
<td>1</td>
</tr>
<tr>
<td>d₃</td>
<td>120</td>
<td>83</td>
<td>2</td>
</tr>
</tbody>
</table>

User query

\[ w = (0.1, 0.2, 0.3, 0.2, 0.1) \]
### Laptops

<table>
<thead>
<tr>
<th>id</th>
<th>CPU</th>
<th>RAM (GB)</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>battery (hours)</th>
<th>weight (kg)</th>
<th>RAM type</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>c_1</td>
<td>3346</td>
<td>4</td>
<td>0</td>
<td>539</td>
<td>4</td>
<td>2.7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>c_2</td>
<td>3346</td>
<td>8</td>
<td>256</td>
<td>1119</td>
<td>6.5</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>c_3</td>
<td>3941</td>
<td>12</td>
<td>256</td>
<td>1199</td>
<td>8</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c_4</td>
<td>3997</td>
<td>8</td>
<td>0</td>
<td>1348</td>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

### Memory

<table>
<thead>
<tr>
<th>mid</th>
<th>RAM (GB)</th>
<th>price (USD)</th>
<th>RAM type</th>
</tr>
</thead>
<tbody>
<tr>
<td>m_1</td>
<td>4</td>
<td>71</td>
<td>2</td>
</tr>
<tr>
<td>m_2</td>
<td>4</td>
<td>45</td>
<td>1</td>
</tr>
<tr>
<td>m_3</td>
<td>8</td>
<td>81</td>
<td>1</td>
</tr>
</tbody>
</table>

### SSD

<table>
<thead>
<tr>
<th>sid</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>d_1</td>
<td>240</td>
<td>143</td>
<td>1</td>
</tr>
<tr>
<td>d_2</td>
<td>512</td>
<td>300</td>
<td>1</td>
</tr>
<tr>
<td>d_3</td>
<td>120</td>
<td>83</td>
<td>2</td>
</tr>
</tbody>
</table>

### Query results:

**Top-3 query**

- c_3
- c_4
- c_2

**User query**

\[
 w = (0.1, 0.2, 0.3, 0.2, 0.1)
\]

---

**Processing and organizing object combinations**

- Laptops
- Memory
- SSD
Processing and organizing object combinations

### Laptops

<table>
<thead>
<tr>
<th>id</th>
<th>CPU score</th>
<th>RAM (GB)</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>battery (hours)</th>
<th>weight (kg)</th>
<th>RAM (GB)</th>
<th>SSD (GB)</th>
<th>type</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>c1</td>
<td>3346</td>
<td>4</td>
<td>0</td>
<td>539</td>
<td>4</td>
<td>2.7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>c2</td>
<td>3346</td>
<td>8</td>
<td>256</td>
<td>1119</td>
<td>6.5</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>c3</td>
<td>3941</td>
<td>12</td>
<td>256</td>
<td>1199</td>
<td>8</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c4</td>
<td>3997</td>
<td>8</td>
<td>0</td>
<td>1348</td>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

### Memory

<table>
<thead>
<tr>
<th>mid</th>
<th>RAM (GB)</th>
<th>price (USD)</th>
<th>RAM type</th>
</tr>
</thead>
<tbody>
<tr>
<td>m1</td>
<td>4</td>
<td>71</td>
<td>2</td>
</tr>
<tr>
<td>m2</td>
<td>4</td>
<td>45</td>
<td>1</td>
</tr>
<tr>
<td>m3</td>
<td>8</td>
<td>81</td>
<td>1</td>
</tr>
</tbody>
</table>

### SSD

<table>
<thead>
<tr>
<th>sid</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>240</td>
<td>143</td>
<td>1</td>
</tr>
<tr>
<td>d2</td>
<td>512</td>
<td>300</td>
<td>1</td>
</tr>
<tr>
<td>d3</td>
<td>120</td>
<td>83</td>
<td>2</td>
</tr>
</tbody>
</table>

**User query**

\[ w = (0.1, 0.2, 0.3, 0.2, 0.1) \]

**Query results:**

**All combinations**

<table>
<thead>
<tr>
<th>rank</th>
<th>combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( {c2, m3} )</td>
</tr>
<tr>
<td>2</td>
<td>( {c2, m3, d3} )</td>
</tr>
<tr>
<td>3</td>
<td>( {c4, m1, d2} )</td>
</tr>
<tr>
<td>4</td>
<td>( {c4, m1, d1} )</td>
</tr>
<tr>
<td>5</td>
<td>( {c4, m1} )</td>
</tr>
<tr>
<td>6</td>
<td>( {c3} )</td>
</tr>
<tr>
<td>7</td>
<td>( {c2, m2} )</td>
</tr>
<tr>
<td>8</td>
<td>( {c2, m2, d3} )</td>
</tr>
<tr>
<td>9</td>
<td>( {c1, m3, d2} )</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>( {c4} )</td>
</tr>
<tr>
<td>15</td>
<td>( {c2} )</td>
</tr>
</tbody>
</table>


### Laptops

<table>
<thead>
<tr>
<th>id</th>
<th>CPU</th>
<th>RAM (GB)</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>battery (hours)</th>
<th>weight (kg)</th>
<th>RAM type</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>c1</td>
<td>3346</td>
<td>4</td>
<td>0</td>
<td>539</td>
<td>4</td>
<td>2.7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>c2</td>
<td>3346</td>
<td>8</td>
<td>256</td>
<td>1119</td>
<td>6.5</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>c3</td>
<td>3941</td>
<td>12</td>
<td>256</td>
<td>1199</td>
<td>8</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c4</td>
<td>3997</td>
<td>8</td>
<td>0</td>
<td>1348</td>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

### Memory

<table>
<thead>
<tr>
<th>mid</th>
<th>RAM (GB)</th>
<th>price (USD)</th>
<th>RAM type</th>
</tr>
</thead>
<tbody>
<tr>
<td>m1</td>
<td>4</td>
<td>71</td>
<td>2</td>
</tr>
<tr>
<td>m2</td>
<td>4</td>
<td>45</td>
<td>1</td>
</tr>
<tr>
<td>m3</td>
<td>8</td>
<td>81</td>
<td>1</td>
</tr>
</tbody>
</table>

### SSD

<table>
<thead>
<tr>
<th>sid</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>240</td>
<td>143</td>
<td>1</td>
</tr>
<tr>
<td>d2</td>
<td>512</td>
<td>300</td>
<td>1</td>
</tr>
<tr>
<td>d3</td>
<td>120</td>
<td>83</td>
<td>2</td>
</tr>
</tbody>
</table>

#### Query results:

**All combinations**

<table>
<thead>
<tr>
<th>rank</th>
<th>combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>{c2, m3}</td>
</tr>
<tr>
<td>2</td>
<td>{c2, m3, d3}</td>
</tr>
<tr>
<td>3</td>
<td>{c4, m1, d2}</td>
</tr>
<tr>
<td>4</td>
<td>{c4, m1, d1}</td>
</tr>
<tr>
<td>5</td>
<td>{c4, m1}</td>
</tr>
<tr>
<td>6</td>
<td>{c3}</td>
</tr>
<tr>
<td>7</td>
<td>{c2, m2}</td>
</tr>
<tr>
<td>8</td>
<td>{c2, m2, d3}</td>
</tr>
<tr>
<td>9</td>
<td>{c1, m3, d2}</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>{c4}</td>
</tr>
<tr>
<td>15</td>
<td>{c2}</td>
</tr>
</tbody>
</table>

**User query**

\[ w = (0.1, 0.2, 0.3, 0.2, 0.1) \]
### Laptops

<table>
<thead>
<tr>
<th>id</th>
<th>CPU</th>
<th>RAM (GB)</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>battery (hours)</th>
<th>weight (kg)</th>
<th>RAM type</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>c₁</td>
<td>3346</td>
<td>4</td>
<td>0</td>
<td>539</td>
<td>4</td>
<td>2.7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>c₂</td>
<td>3346</td>
<td>8</td>
<td>256</td>
<td>1119</td>
<td>6.5</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>c₃</td>
<td>3941</td>
<td>12</td>
<td>256</td>
<td>1199</td>
<td>8</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c₄</td>
<td>3997</td>
<td>8</td>
<td>0</td>
<td>1348</td>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

### Memory

<table>
<thead>
<tr>
<th>mid</th>
<th>RAM (GB)</th>
<th>price (USD)</th>
<th>RAM type</th>
</tr>
</thead>
<tbody>
<tr>
<td>m₁</td>
<td>4</td>
<td>71</td>
<td>2</td>
</tr>
<tr>
<td>m₂</td>
<td>4</td>
<td>45</td>
<td>1</td>
</tr>
<tr>
<td>m₃</td>
<td>8</td>
<td>81</td>
<td>1</td>
</tr>
</tbody>
</table>

### SSD

<table>
<thead>
<tr>
<th>sid</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>d₁</td>
<td>240</td>
<td>143</td>
<td>1</td>
</tr>
<tr>
<td>d₂</td>
<td>512</td>
<td>300</td>
<td>1</td>
</tr>
<tr>
<td>d₃</td>
<td>120</td>
<td>83</td>
<td>2</td>
</tr>
</tbody>
</table>

### User query

\[ w = (0.1, 0.2, 0.3, 0.2, 0.1) \]
Processing and organizing object combinations

### Laptops

<table>
<thead>
<tr>
<th>id</th>
<th>CPU score (GB)</th>
<th>RAM (GB)</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>battery (hours)</th>
<th>weight (kg)</th>
<th>RAM type</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>c1</td>
<td>3346</td>
<td>4</td>
<td>0</td>
<td>539</td>
<td>4</td>
<td>2.7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>c2</td>
<td>3346</td>
<td>8</td>
<td>256</td>
<td>1119</td>
<td>6.5</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>c3</td>
<td>3941</td>
<td>12</td>
<td>256</td>
<td>1199</td>
<td>8</td>
<td>2.5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c4</td>
<td>3997</td>
<td>8</td>
<td>0</td>
<td>1348</td>
<td>13</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

### Memory

<table>
<thead>
<tr>
<th>mid</th>
<th>RAM (GB)</th>
<th>price (USD)</th>
<th>RAM type</th>
</tr>
</thead>
<tbody>
<tr>
<td>m1</td>
<td>4</td>
<td>71</td>
<td>2</td>
</tr>
<tr>
<td>m2</td>
<td>4</td>
<td>45</td>
<td>1</td>
</tr>
<tr>
<td>m3</td>
<td>8</td>
<td>81</td>
<td>1</td>
</tr>
</tbody>
</table>

### SSD

<table>
<thead>
<tr>
<th>sid</th>
<th>SSD (GB)</th>
<th>price (USD)</th>
<th>SSD type</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>240</td>
<td>143</td>
<td>1</td>
</tr>
<tr>
<td>d2</td>
<td>512</td>
<td>300</td>
<td>1</td>
</tr>
<tr>
<td>d3</td>
<td>120</td>
<td>83</td>
<td>2</td>
</tr>
</tbody>
</table>

**User query**

\[
\mathbf{w} = (0.1, 0.2, 0.3, 0.2, 0.1)
\]
We introduce the Exploratory Top-k Join ($XTJ_k$) queries that consider all possible combinations between main and accessory items. $XTJ_k$ queries return the best combination for each main item. A state-of-the-art approach (HRJN) is adapted to process $XTJ_k$ queries. A more efficient algorithm ($XRJN$) is proposed. $XRJN$ exploits the “star”-like structure of the joins.
Exploratory rank-join (XRJN) algorithm

<table>
<thead>
<tr>
<th></th>
<th>Laptops</th>
<th>RAM</th>
<th>SSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>7+</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>10*</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>6*</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>7+</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>5</td>
<td>3*+</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Searching for the top-1 combination

Best seen score: 16
Exploratory rank-join (XRJN) algorithm

<table>
<thead>
<tr>
<th>Laptops</th>
<th>RAM</th>
<th>SSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>7+</td>
</tr>
<tr>
<td>2</td>
<td>10*</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>6*</td>
</tr>
<tr>
<td>4</td>
<td>7+</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Searching for the **top-1** combination

Best seen score: 16

<table>
<thead>
<tr>
<th>Combinations</th>
<th>Score</th>
<th>Potential Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptops[1]</td>
<td>10</td>
<td>18</td>
</tr>
<tr>
<td>Laptops[2]</td>
<td>16</td>
<td>19</td>
</tr>
<tr>
<td>RAM[3]</td>
<td>16</td>
<td>19</td>
</tr>
<tr>
<td>Laptops[3]</td>
<td>8</td>
<td>13</td>
</tr>
<tr>
<td>Laptops[4]</td>
<td>14</td>
<td>17</td>
</tr>
<tr>
<td>RAM[1]</td>
<td>14</td>
<td>17</td>
</tr>
</tbody>
</table>
Exploratory rank-join (XRJN) algorithm

<table>
<thead>
<tr>
<th>Laptops</th>
<th>RAM</th>
<th>SSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>7+</td>
</tr>
<tr>
<td>2</td>
<td>10*</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>6*</td>
</tr>
<tr>
<td>4</td>
<td>7+</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>3*+</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Searching for the **top-1** combination

Best seen score: 16

<table>
<thead>
<tr>
<th>Combinations</th>
<th>Score</th>
<th>Potential Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptops[1]</td>
<td>10</td>
<td>18</td>
</tr>
<tr>
<td>RAM[3]</td>
<td>16</td>
<td><strong>19</strong></td>
</tr>
<tr>
<td>Laptops[3]</td>
<td>8</td>
<td>13</td>
</tr>
<tr>
<td>Laptops[4]</td>
<td>14</td>
<td>17</td>
</tr>
</tbody>
</table>
Performance evaluation (Uniform dataset)

![Graph showing performance evaluation results for different number of add. relations.](image-url)
Most continuous influential objects

**RQ3**

How can we efficiently identify objects that are constantly highly ranked by a large number of users over a specific time-period?
Top-$k$ queries
Top-$k$ queries

$TOP_2(w_1) = \{ G, A \}$
Top-$k$ queries

\[ \text{TOP}_2(w_1) = \{G, A\} \]

\[ \text{TOP}_2(w_2) = \{G, B\} \]
Reverse Top-$k$ queries

A reverse top-$k$ query returns the preference vectors $w$ for which an object $q$ is in the $TOP_k$ set.

Reverse top-$k$ queries:
- $RTOP_2(A) = \{w_1\}$
- $RTOP_2(G) = \{w_1, w_2\}$
Reverse Top-k queries

A reverse top-k query returns the preference vectors \( w \) for which an object \( q \) is in the \( \text{TOP}_k \) set.

- The influence score of an object is the size of the respective \( \text{RTOP}_k \) set.
  - It indicates how many people are going to see a product.

\[
\begin{align*}
\text{RTOP}_2(A) &= \{w_1\} \\
\text{RTOP}_2(G) &= \{w_1, w_2\}
\end{align*}
\]
Reverse Top-k queries

A reverse top-
k query returns the preference vectors \( w \) for which an object \( q \) is in the \( TOP_k \) set.

The influence score of an object is the size of the respective \( RTOP_k \) set.

- It indicates how many people are going to see a product.

The influence score of \( G \), \( f_k^I(G) = 2 \)

\[ RTOP_2(A) = \{ w_1 \} \]
\[ RTOP_2(G) = \{ w_1, w_2 \} \]
Reverse Top-$k$ queries

A reverse top-$k$ query returns the preference vectors $w$ for which an object $q$ is in the $TOP_k$ set.

The influence score of an object is the size of the respective $RTOP_k$ set.

- $RTOP_2(A) = \{w_1\}$
- $RTOP_2(G) = \{w_1, w_2\}$

The influence score of $G$, $f_k^I(G) = 2$.

The most influential object is $G$.

$ITOP_2^1 = G$.
Finding objects that are continuously influential

- The influence score does not consider the temporal aspect of user preferences
  - Instantly influential objects cannot be distinguished from continuously influential objects
- How can we find objects that are continuously influential for long time periods?
  - Associate each preference vector a time-stamp and a validity period
  - Partition the time domain of the preference vectors in a set of time-intervals
  - Find the objects that are influential in the **longest continuous sequence** of time intervals
Finding objects that are continuously influential

- The influence score does not consider the temporal aspect of user preferences
  - Instantly influential objects cannot be distinguished from continuously influential objects
- How can we find objects that are continuously influential for long time periods?
  - Associate each preference vector a time-stamp and a validity period
  - Partition the time domain of the preference vectors in a set of time-intervals
  - Find the objects that are influential in the **longest continuous sequence** of time intervals

![Time domain with continuity scores]

<table>
<thead>
<tr>
<th>Time Intervals</th>
<th>Continuity Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>cis(A) = 2</td>
</tr>
<tr>
<td>C</td>
<td>cis(B) = 4</td>
</tr>
<tr>
<td>A</td>
<td>cis(C) = 1</td>
</tr>
<tr>
<td>A</td>
<td>cis(D) = 2</td>
</tr>
</tbody>
</table>

**continuity score (cis) = 4**
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

$p_1(1, 10)$
$p_2(1, 10)$
$p_3(1, 10)$

$UB(p_1)$
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

- $p_1(1, 10)$
- $p_2(1, 9)$
- $p_3(1, 9)$
- $p_5(1, 9)$
- $p_6(1, 9)$

UB($p_1$)
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

\[ p_1(1,5) \]
\[ p_2(1,9) \]
\[ p_3(1,9) \]
\[ p_4(1,9) \]
\[ p_5(1,4) \]
\[ p_6(1,4) \]

Accessed objects

\[ UB(p_1) \]
\[ UB(p_2) \]
\[ UB(p_3) \]

Accessed objects

\[ UB(p_4) \]
\[ UB(p_5) \]
\[ UB(p_6) \]
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

\( p_1(1,5) \)
\( p_2(1,9) \)
\( p_3(1,9) \)
\( p_4(1,9) \)
\( p_5(1,4) \)
\( p_6(1,4) \)

UB\((p_2)\)

1 2 3 4 5 6 7 8 9 10
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

\[ p_1(1, 4) \]
\[ p_2(1, 4) \]
\[ p_3(1, 4) \]
\[ p_4(1, 4) \]
\[ p_5(1, 4) \]
\[ p_6(1, 4) \]
\[ p_7(1, 4) \]
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

- $p_1(1, 4)$
- $p_2(1, 4)$
- $p_3(1, 4)$
- $p_4(1, 4)$
- $p_5(1, 4)$
- $p_6(1, 4)$
- $p_7(1, 4)$

$UB(p_1)$
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

\( p_1(1, 4) \)
\( p_2(1, 4) \)
\( p_3(1, 4) \)
\( p_4(1, 4) \)
\( p_5(1, 4) \)
\( p_6(1, 4) \)
\( p_7(1, 2) \)
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

- $p_1(1, 4)$
- $p_2(1, 4)$
- $p_3(1, 4)$
- $p_4(1, 4)$
- $p_5(1, 4)$
- $p_6(1, 4)$
- $p_7(1, 2)$

$UB(p_1)$
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

\[ p_1(3, 4) \]
\[ p_2(2, 4) \]
\[ p_3(1, 4) \]
\[ p_4(1, 4) \]
\[ p_5(1, 4) \]
\[ p_6(1, 4) \]
\[ p_7(1, 2) \]
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

- \( p_1(4, 4) \)
- \( p_2(2, 4) \)
- \( p_3(2, 4) \)
- \( p_4(2, 4) \)
- \( p_5(1, 4) \)
- \( p_6(1, 4) \)
- \( p_7(1, 2) \)
Algorithm: Early Termination Best-first Interval (TBI)

Accessed objects

\[ p_1(4, 4) \]
\[ p_2(2, 4) \]
\[ p_3(2, 4) \]
\[ p_4(2, 4) \]
\[ p_5(1, 4) \]
\[ p_6(1, 4) \]
\[ p_7(1, 2) \]
Selecting diverse items

**RQ4**

How can we efficiently identify groups of products that attract customers with diverse preferences?
Selecting $r$ diverse objects

- Each object $p$ corresponds to an $RTOP_k(p)$ set.
- For two objects $p, q$, their distance $d(p, q)$ is determined based on their $RTOP_k$ sets.
  \[ d(p, q) = f_d(RTOP_k(p), RTOP_k(q)) \]

**r-Diversity problem**: Find a set of $r$ objects such that:

\[
D^* = \arg\max_{|D|=r} \sum_{p,q \in D \atop p \neq q} d(p, q)
\]
Finding the most diverse objects

- Two-step approach (All Top-k):
  - Calculate the centroids of all $RTOP_k$ sets
  - Find the most diverse centroids

- The processing cost of the two-step approach can be quite high

- One step approach (Selective Top-k):
  - Searches for diverse preference vectors instead of centroids
  - Updates the centroid vectors progressively
  - It does not require the exact calculation of the centroids
  - Performs a small number of top-$k$ queries
Selective Top-k Example

Finding the 2 most diverse items

<table>
<thead>
<tr>
<th>Initialization step:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TOP_k(w_1) = p_1, p_2, p_3$, $TOP_k(w_2) = p_2, p_4, p_5$</td>
</tr>
<tr>
<td>$c_{p_1} = w_1$, $c_{p_2} = \frac{1}{2}(w_1 + w_2)$,</td>
</tr>
<tr>
<td>$c_{p_3} = w_1$, $c_{p_4} = w_2$, $c_{p_5} = w_2$</td>
</tr>
</tbody>
</table>

$Dr = \{c_{p_1}, c_{p_4}\}$
Selective Top-k-example

- Finding the 2 most diverse items

<table>
<thead>
<tr>
<th>Initialization step:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TOP_k(w_1) = p_1, p_2, p_3$, $TOP_k(w_2) = p_2, p_4, p_5$</td>
</tr>
<tr>
<td>$c_{p_1} = w_1$, $c_{p_2} = \frac{1}{2}(w_1 + w_2)$,</td>
</tr>
<tr>
<td>$c_{p_3} = w_1$, $c_{p_4} = w_2$, $c_{p_5} = w_2$</td>
</tr>
<tr>
<td>$D_r = {c_{p_1}, c_{p_4}}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>First step:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TOP_k(w_3) = p_3, p_4, p_5$</td>
</tr>
<tr>
<td>$c_{p_1} = w_1$, $c_{p_2} = \frac{1}{2}(w_1 + w_2)$, $c_{p_3} = \frac{1}{2}(w_1 + w_3)$</td>
</tr>
<tr>
<td>$c_{p_4} = \frac{1}{2}(w_2 + w_3)$, $c_{p_5} = \frac{1}{2}(w_2 + w_3)$</td>
</tr>
<tr>
<td>$D_r = {c_{p_1}, c_{p_4}}$</td>
</tr>
</tbody>
</table>
Selective Top-$k$-example

- Finding the 2 most diverse items

<table>
<thead>
<tr>
<th>Initialization step:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TOP_k(w_1) = p_1, p_2, p_3,$ $TOP_k(w_2) = p_2, p_4, p_5$</td>
</tr>
<tr>
<td>$c_{p_1} = w_1,$ $c_{p_2} = \frac{1}{2}(w_1 + w_2),$</td>
</tr>
<tr>
<td>$c_{p_3} = w_1,$ $c_{p_4} = w_2,$ $c_{p_5} = w_2$</td>
</tr>
</tbody>
</table>

| $D_r = \{c_{p_1}, c_{p_4}\}$ |

<table>
<thead>
<tr>
<th>First step:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TOP_k(w_3) = p_3, p_4, p_5$</td>
</tr>
<tr>
<td>$c_{p_1} = w_1,$ $c_{p_2} = \frac{1}{2}(w_1 + w_2),$ $c_{p_3} = \frac{1}{2}(w_1 + w_3)$</td>
</tr>
<tr>
<td>$c_{p_4} = \frac{1}{2}(w_2 + w_3),$ $c_{p_5} = \frac{1}{2}(w_2 + w_3)$</td>
</tr>
</tbody>
</table>

| $D_r = \{c_{p_1}, c_{p_4}\}$ |

<table>
<thead>
<tr>
<th>Second step:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$TOP_k(w_4) = p_1, p_2, p_6$</td>
</tr>
<tr>
<td>$c_{p_1} = \frac{1}{2}(w_1 + w_4),$ $c_{p_2} = \frac{1}{3}(w_1 + w_2 + w_4),$</td>
</tr>
<tr>
<td>$c_{p_3} = \frac{1}{2}(w_1 + w_3),$ $c_{p_4} = \frac{1}{2}(w_2 + w_3),$</td>
</tr>
<tr>
<td>$c_{p_5} = \frac{1}{2}(w_2 + w_3),$ $c_{p_6} = w_4$</td>
</tr>
</tbody>
</table>

| $D_r = \{c_{p_1}, c_{p_5}\}$ |
Experimental evaluation

![Graph showing total time (sec) vs. data cardinality for Stopk and Atopk](image-url)

- X-axis: Data Cardinality (100K, 200K, 500K)
- Y-axis: total time (sec)

- Bars represent the total time for Stopk and Atopk for different data cardinalities.
Maximizing influence of spatio-textual objects

RQ 5
How can a description of a service or product be enriched to improve its visibility?
How can we improve the visibility of hotel A?
How can we improve the visibility of hotel A?

- Improve the influence score
Example

- How can we improve the visibility of hotel A?
- Improve the influence score
- Enhance its textual description
Problem description

Best-terms query
Given a spatio-textual object $q$ and a number of $b$, find $b$ terms that will maximize the influence score of $q$.

- Best-terms query is NP-hard
  - Reduction to the Maximum-Coverage (MC) problem
- We employed two greedy algorithms
  - Best Term First (BTF): Adaptation of the greedy algorithm for the MC problem
  - Graph-Based Term Selection (GBTS)
Graph construction algorithm

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

User terms

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

$b=3$
Graph construction algorithm

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

$$b = 3$$

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

User terms

1. $t_1$
2. $t_3$
3. $t_2$
Graph construction algorithm

User terms

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

query

<table>
<thead>
<tr>
<th>object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

$b=3$

1

$1 \rightarrow t_1$

1

$1 \rightarrow t_3$

1

$1 \rightarrow t_2$

$1 \rightarrow t_5$

$1 \rightarrow t_4$
Graph construction algorithm

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

b = 3

User terms

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>
Graph construction algorithm

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>t₀</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td>t₁, t₂, t₃</td>
<td>1</td>
</tr>
<tr>
<td>u₂</td>
<td>t₂, t₄, t₅</td>
<td>2</td>
</tr>
<tr>
<td>u₃</td>
<td>t₂, t₃, t₅, t₆</td>
<td>3</td>
</tr>
<tr>
<td>u₄</td>
<td>t₁, t₃, t₆</td>
<td>3</td>
</tr>
</tbody>
</table>

b = 3
Graph construction algorithm

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

User terms

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

$b=3$
Graph construction algorithm

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

**User terms**

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

$b=3$
Best subgraph selection

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

User terms

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

b=3
Best subgraph selection

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

User terms

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

$b=3$
Best subgraph selection

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

User terms

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

$b=3$
Best subgraph selection

<table>
<thead>
<tr>
<th>query object</th>
<th>existing terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>q</td>
<td>$t_0$</td>
</tr>
</tbody>
</table>

**User terms**

<table>
<thead>
<tr>
<th>user</th>
<th>terms</th>
<th>min terms to be added</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$t_1, t_2, t_3$</td>
<td>1</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$t_2, t_4, t_5$</td>
<td>2</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$t_2, t_3, t_5, t_6$</td>
<td>3</td>
</tr>
<tr>
<td>$u_4$</td>
<td>$t_1, t_3, t_6$</td>
<td>3</td>
</tr>
</tbody>
</table>

$b=3$
Experimental evaluation

![Graph showing time (sec) vs. data cardinality for GBTS and BTF methods. The data cardinality is given in thousands (10K, 20K, 30K, 50K). The graph indicates that GBTS generally has lower time (sec) compared to BTF across different data cardinalities.](image-url)
Conclusion

- **Main topic**
  - Visibility enhancement of database content
    - Provide an overview of the available content through summarized results
    - Reveal objects that are potentially attractive to the user basis

- **Future directions**
  - Exploration in spatio-textual data
  - Exploratory search in dynamic data and streams
  - Big Data exploration


Orestis Gkorgkas, Akrivi Vlachou, Christos Doulkeridis, and Kjetil Nørvåg. Exploratory product search using top-k join queries. *(Under submission).*
Thank you!

Thank you for your attention!
Keyword search running times

clustering time
JTT generation time

queries

msec

10.0M
1.0M
100.0k
10.0k
1.0k
100.0
10.0
1.0
q2 q4 q5 q9 q10 q11 q12 q13 q15 q16 q17 q18 q26 q27 q28 q30 q31 q32 q33 q34