Master Thesis – MovieSearch
Building semantic search queries with suggestions

Tobias Sommer
Albert-Ludwigs-University Freiburg
Faculty of Engineering
Chair of Algorithms and Data Structures

1 MovieSearch – User Interface
   - Goals
   - Components
   - Architecture

2 Evaluation
   - User study – Quality of query building
   - Quality of results and their ranking

3 Conclusions
   - Discussion: Achieved goals
   - Reference
Goals

- Provide **user-friendly** Interface for Semantic Search in the domain movie
- Utilize plot and facts
- Support 3-ary relations
Use cases

Example tasks to fulfill:

- Find movies made by Jerry Bruckheimer.
  Explore data, e.g. relation names.

- Find movies where Frodo was played by Elijah Wood.
  Use and connect 3-ary relations.

- Find an action movie with Arnold Schwarzenegger where he fights with a sword.
  Query conditions: plot snippets and facts.
Use cases

Example tasks to fulfill:

- *Find movies made by Jerry Bruckheimer.*
  
  Explore data, e.g. relation names.

- *Find movies where Frodo was played by Elijah Wood.*
  
  Use and connect **3-ary relations**.

- *Find an action movie with Arnold Schwarzenegger where he fights with a sword.*
  
  Query conditions: **plot snippets and facts**.
Use cases

Example tasks to fulfill:

- Find movies made by Jerry Bruckheimer.  
  Explore data, e.g. relation names.

- Find movies where Frodo was played by Elijah Wood.  
  Use and connect 3-ary relations.

- Find an action movie with Arnold Schwarzenegger where he fights with a sword.  
  Query conditions: plot snippets and facts.
Splitting plot and facts

Consider use case:

Find an action movie
with Arnold Schwarzenegger
where he fights with a sword.

Plot
Text information → ... fights with a sword.

Facts
Structured information → ... with actor A. Schwarzenegger.
Searching for

Movie

Describe Plot

fights sword

Insert a Fact description

Action

Current Query consists of:

Looking for Movie with ...

Plot: fights sword

has genre
Action

Results for current Query:

60 total matches found in 4 ms

Equilibrium (2002)

When Preston fights the bodyguards with the swords, as he kills the last two men you can see that the man behind him (whom Preston just stuck a sword through) doesn’t hold the first part of the sword in a straight line, but accidentally holds it in an angle as he collapses.
User Interface – Example (2/3)

Searching for

Current Query consists of:

Looking for Movie with ...

Plot: fights sword

has genre Action

Results for current Query:

60 total matches found in 2 ms

Equilibrium (2002)

When Preston fights the bodyguards with the swords, as he kills the last two men you can see that the man behind him (whom Preston just stuck a sword through) doesn’t hold the first part of the sword in a straight line, but accidentally holds it in an angle as he collapses.
Current Query consists of:

Looking for Movie with...

- Plot: fight sword
- has genre: Action

in movie as actor: Schwarzenegger, Arnold

Results for current Query:

2 total matches found in 230 ms

Conan the Destroyer (1984)

During the fight at the orgy, Conan appears to strike a guard with the pommel (nilt) of his sword. The strike clearly misses, but the guard reacts as if it connected.

has rating: 5.8 with votes: 58,593

directed by: Fleischer, Richard
Facts

Store structured information as triples:

Example

(Conan, has-genre, Action)

In general

Fact := (Entity, relation, value)

Different kind of values

- word: (Inception, has-genre, Action)
- entity: (Inception, directed-by, Christopher Nolan)
- number: (Inception, has-budget, 160.000.000 $)
- date: (Inception, released, 29.07.2010)
Relations

Occurring relations:

- **Binary** relations

  Example
  as triple  \((\text{Conan}, \text{has-genre}, \text{Action})\)

- **3-ary** relations

  Example
  from text  \(\text{Mel Gibson plays William Wallace in Braveheart.}\)
  as triples  \((\text{cast-link}_1, \text{in-movie}, \text{Braveheart})\)
  \((\text{cast-link}_1, \text{actor}, \text{Mel Gibson})\)
  \((\text{cast-link}_1, \text{character}, \text{William Wallace})\)
Fact suggestions

Suggesting facts during input:
⇒ Discover names in unfamiliar data
⇒ Find connectable relations

How to find suggestions

**Names:** match description to relation and entity names
⇒ Inverted index of prefixes

**Triples:** find (relation, value) pairs
⇒ Facts graph from triples
Matching names – Inverted index

Example (Inverted index of prefixes)

<table>
<thead>
<tr>
<th>ID</th>
<th>Entity names</th>
<th>Prefix</th>
<th>ID lists</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>&quot;Barbara&quot;</td>
<td>[b]</td>
<td>0, 1</td>
</tr>
<tr>
<td>1</td>
<td>&quot;Bill Berg&quot;</td>
<td>[ba]</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[be]</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[bi]</td>
<td>1</td>
</tr>
</tbody>
</table>

Matching names:
- Get ID lists of matching prefixes
- Intersect all (sorted) ID lists
- Further filter for contains from start
Build a graph from triples.

**Example (Triples)**

- (Braveheart, written-by, Randall Wallace)
- (cast-link₁, in-movie, Braveheart)
- (cast-link₁, actor, Mel Gibson)
- (cast-link₁, character, William Wallace)
Connectable relations – Facts graph (2/2)

- Braveheart
- cast-link\(_1\)
- William Wallace
- character
- in-movie
- written-by
- Randall Wallace
- actor
- Mel Gibson

**MovieSearch**
Usability evaluation

1. MovieSearch – User Interface
   - Goals
   - Components
   - Architecture

2. Evaluation
   - User study – Quality of query building
   - Quality of results and their ranking

3. Conclusions
   - Discussion: Achieved goals
   - Reference
User study

8 participants
21 tasks $Q_i$
Build a query for each task

⇒ Evaluating usability
User study – Quality of query building

Evaluating the building process

Count number of text inputs needed to build a query.

<table>
<thead>
<tr>
<th></th>
<th>user study text inputs</th>
<th>minimum text inputs</th>
<th>relative user extra input</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. $Q_i$</td>
<td>3.2</td>
<td>2</td>
<td>68%</td>
</tr>
</tbody>
</table>

$\approx$ one extra input
User study – Input comparison

\[ Q_{11} \] "In which movies directed by Garry Marshall was Julia Roberts starring?".

Compare minimum text inputs needed for \[ Q_{11} \] :

<table>
<thead>
<tr>
<th>Graph-based Systems</th>
<th>Text inputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>GoRelations</td>
<td>9</td>
</tr>
<tr>
<td>NotAnotherGoogleAnswer</td>
<td>6</td>
</tr>
<tr>
<td>SFC (Semantic Focused Crawler)</td>
<td>5</td>
</tr>
<tr>
<td>MovieSearch [minimum]</td>
<td>2</td>
</tr>
<tr>
<td>MovieSearch [study avg.]</td>
<td>2.8</td>
</tr>
</tbody>
</table>

## User study – Quality of the built queries

Results with the built queries in the user study:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total query answers</strong></td>
<td>168</td>
<td></td>
</tr>
<tr>
<td><strong>with expected results</strong></td>
<td>159</td>
<td><strong>94.64%</strong></td>
</tr>
<tr>
<td><strong>expected Result with expected Query</strong></td>
<td>130</td>
<td><strong>77.38%</strong></td>
</tr>
<tr>
<td><strong>expected Result with other Query</strong></td>
<td>29</td>
<td><strong>17.26%</strong></td>
</tr>
</tbody>
</table>
Results of the queries: Quality and ranking (1/2)

MovieSearch vs. natural-language-based UI (Valossa)

- Usability → natural-language-based is main competitor
- Compare results for the 21 tasks $Q_i$:
  MovieSearch expected queries vs. Valossa task text input

- Regard Top 10 results
- Ranking quality via Discounted Cumulative Gain, for $w_i \in \{0, 1\}$:

\[
DCG_{10} := w_1 + \sum_{i=2}^{10} \frac{w_i}{\log_2 i}.
\]
Results of the queries: Quality and ranking (2/2)

<table>
<thead>
<tr>
<th></th>
<th>avg. Recall</th>
<th>avg. Precision</th>
<th>avg. $NDCG_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MovieSearch</td>
<td>66.60%</td>
<td>94.96%</td>
<td>95.81%</td>
</tr>
<tr>
<td>Valossa</td>
<td>47.62%</td>
<td>35.00%</td>
<td>62.19%</td>
</tr>
</tbody>
</table>

- Recall $\rightarrow$ tasks more hits than 10
  - $Q_2$ Movies with songs from Hans Zimmer.

- Valossa: answers without any hit
  - $Q_{20}$ Movie with Angelina Jolie and Brad Pitt where they have secrets.
  - $Q_7$ Movie that is 111 minutes long and released at 11.11.2011.

- MovieSearch: hard criteria with facts
  - Tradeoff: (high Precision) $\triangleright$ (potential for almost hits)
## Results of the queries: Quality and ranking (2/2)

<table>
<thead>
<tr>
<th></th>
<th>avg. Recall</th>
<th>avg. Precision</th>
<th>avg. $NDCG_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MovieSearch</td>
<td>66.60%</td>
<td>94.96%</td>
<td>95.81%</td>
</tr>
<tr>
<td>Valossa</td>
<td>47.62%</td>
<td>35.00%</td>
<td>62.19%</td>
</tr>
</tbody>
</table>

- Recall → tasks more hits than 10
  
  $Q_2$ Movies with songs from Hans Zimmer.

- Valossa: answers without any hit
  
  $Q_{20}$ Movie with Angelina Jolie and Brad Pitt where they have secrets.

- $Q_7$ Movie that is 111 minutes long and released at 11.11.2011.

- MovieSearch: hard criteria with facts
  
  Tradeoff: (high Precision) (potential for almost hits)
### Results of the queries: Quality and ranking (2/2)

<table>
<thead>
<tr>
<th>Method</th>
<th>avg. Recall</th>
<th>avg. Precision</th>
<th>avg. $NDCG_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MovieSearch</td>
<td>66.60%</td>
<td>94.96%</td>
<td>95.81%</td>
</tr>
<tr>
<td>Valossa</td>
<td>47.62%</td>
<td>35.00%</td>
<td>62.19%</td>
</tr>
</tbody>
</table>

- **Recall** → tasks more hits than 10
  
  $Q_2$: *Movies with songs from Hans Zimmer.*

- **Valossa:** answers without any hit
  
  $Q_{20}$: *Movie with Angelina Jolie and Brad Pitt where they have secrets.*
  
  $Q_7$: *Movie that is 111 minutes long and released at 11.11.2011.*

- **MovieSearch:** hard criteria with facts
  
  Tradeoff: (high Precision) $\triangleright$ (potential for almost hits)
Results of the queries: Quality and ranking (2/2)

<table>
<thead>
<tr>
<th></th>
<th>avg. Recall</th>
<th>avg. Precision</th>
<th>avg. $NDCG_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MovieSearch</td>
<td>66.60%</td>
<td>94.96%</td>
<td>95.81%</td>
</tr>
<tr>
<td>Valossa</td>
<td>47.62%</td>
<td>35.00%</td>
<td>62.19%</td>
</tr>
</tbody>
</table>

- **Recall → tasks more hits than 10**
  
  $Q_2$: Movies with songs from Hans Zimmer.

- **Valossa: answers without any hit**
  
  $Q_{20}$: Movie with Angelina Jolie and Brad Pitt where they have secrets.
  
  $Q_7$: Movie that is 111 minutes long and released at 11.11.2011.

- **MovieSearch: hard criteria with facts**

  Tradeoff: (high Precision) $\triangleright$ (potential for almost hits)
1. MovieSearch – User Interface
   - Goals
   - Components
   - Architecture

2. Evaluation
   - User study – Quality of query building
   - Quality of results and their ranking

3. Conclusions
   - Discussion: Achieved goals
   - Reference
Goals achieved?

Query-building worked in a lot of the cases from the user study.

- **Provide user-friendly Interface**
  94% successful answered tasks in user study

- **Utilize plot and facts**
  Splitting tasks – only problem affecting results
  \(\Rightarrow\) More help from UI would be good

- **Support 3-ary relations**  \(\Rightarrow\) Better awareness

- Partial value matching
Goals achieved?

Query-building worked in a lot of the cases from the user study.

- **Provide user-friendly Interface**
  94% successful answered tasks in user study

- **Utilize plot and facts**
  Splitting tasks – only problem affecting results

  ⇒ More help from UI would be good

- **Support 3-ary relations**

  ⇒ Better awareness

- Partial value matching
Goals achieved?

Query-building worked in a lot of the cases from the user study.

- **Provide user-friendly Interface**
  94% successful answered tasks in user study

- **Utilize plot and facts**
  Splitting tasks – only problem affecting results
  ⇒ More help from UI would be good

- **Support 3-ary relations**
  ⇒ Better awareness

⊕ Partial value matching
Goals achieved?

Query-building worked in a lot of the cases from the user study.

• *Provide user-friendly Interface*
  94% successful answered tasks in user study

• *Utilize plot and facts*
  Splitting tasks – only problem affecting results
  \[ \Rightarrow \text{More help from UI would be good} \]

• *Support 3-ary relations* \[ \Rightarrow \text{Better awareness} \]

⊕ Partial value matching
**SPARQL Backend** at
https://github.com/Buchhold/SparqlEngineDraft

[Styperek:2015]
STYPEREK, Adam ; CIESIELCZYK, Michal ; SZWABE, Andrzej ; MISIOREK, Pawel:
**Evaluation of SPARQL-compliant semantic search user interfaces.**

Thank you for your attention.