Example-driven Search: a New Frontier for Exploratory Search

Matteo Lissandrini (Aalborg University), Davide Mottin (Aarhus University), Yannis Velegrakis (Utrecht University), Themis Palpanas (University of Paris)
Link for questions
https://j.mp/ExploreSIGIR

Tutorial Slides and Other Material
https://data-exploration.ml/
LET’S SOLVE THIS PROBLEM BY USING THE BIG DATA NONE OF US HAVE THE SLIGHTEST IDEA WHAT TO DO WITH
Traditional Data Management Systems

rdf:Type=GlutamateReceptor
ro:has_function=ex:GlutamateReceptorActivity001

Data
Modern Data Management Systems

Not clear what we are looking for

I would like to find acquisitions like the one of YouTube by Google
Exploration

We know where we start
we don’t know what we’ll find
Exploration

Traditional

On data
Data exploration

Cleaning and profiling

Visualization

Analysis

Interactions

Architectures

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

SIGIR 2019 tutorial
https://data-exploration.ml
Data exploration software

**Trifacta**: data preparation

**Tableau**: analysis and statistics

**OpenRefine**: data preparation and cleanup
Traditional data exploration methods

Efficiently extracting knowledge from data even if we do not know exactly what we are looking for

\[
\text{SELECT } \text{avg(system-stars)} \\
\text{FROM } \text{Universe} \\
\text{WHERE } \text{system-stars} > 10 \\
\text{GROUP BY } \text{galaxy}
\]

[Idreos et al., 2015]

Not easy for novices
Modern Data Management Systems

How do we describe what we are looking for?

I would like to find acquisitions like the one of YouTube by Google.
Declarative Exploratory methods

**Simple query (exploratory)**

```sql
SELECT galaxy_name
FROM Universe.Galaxy
```

**Over generic 100 billions results**

**Complex query (for data experts)**

```sql
SELECT g.galaxy_name, SUM(s.stars) as st_s
FROM Universe.Galaxy AS g
JOIN Universe.Systems AS s
ON g.galaxy_name = s.galaxy_name
WHERE
  g.st_s > 100B
  AND diameter > 100k AND diameter > 180k
  AND has_black_hole = TRUE
GROUP BY g.galaxy_name
```

**Specific Few results**

**SELECT**

```
g.galaxy_name, SUM(s.stars) as st_s
```

**FROM**

```
Universe.Galaxy AS g
```

**JOIN**

```
Universe.Systems AS s
```

**ON**

```
g.galaxy_name = s.galaxy_name
```

**WHERE**

```
g.st_s > 100B
AND diameter > 100k AND diameter > 180k
AND has_black_hole = TRUE
```

**GROUP BY**

```
g.galaxy_name
```
Examples as Exploratory Methods

Is there a galaxy like this?

Example is always more efficacious than precept
Samuel Johnson, Rasselas (1759), Chapter 29.

Answers
Tutorial’s goals

Techniques, Algorithms, Applications for using Examples to support Exploratory

- Exploratory methods using examples
- Algorithms for retrieving data without using query languages
- Interactive methods and user-in-the-loop feedback
- Machine learning for adaptive, online methods

But NOT

- Declarative query methods
- User interfaces and visualization
- Optimizations for fast data access
- Dynamic data
Our book on Example-based methods

Matteo Lissandrini
Aalborg University
Knowledge Graphs, Novel Query Paradigms, Graph Mining
http://people.cs.aau.dk/~matteo

Yannis Velegrakis
Utrecht University
Big Data Management & Analytics, Information Integration, Data Curation
https://yelias.github.io

Davide Mottin
Aarhus University
Graph Mining, Novel Query Paradigms, Interactive Methods
https://mott.in

Themis Palpanas
Paris Descartes University
Data Series Indexing & Mining, Data Analytics & Management
http://www.ml.parisdescartes.fr/~themisp

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

SIGIR 2019 tutorial
https://data-exploration.ml
Historical perspective: Query-by-example

Specify a query by example tables, or skeletons.

<table>
<thead>
<tr>
<th>Name</th>
<th>Stars</th>
<th>Diameter</th>
<th>Black_hole</th>
<th>Color</th>
<th>Life</th>
</tr>
</thead>
<tbody>
<tr>
<td>P._</td>
<td>&gt; 10B</td>
<td>&gt;100k</td>
<td>TRUE</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>*</td>
<td>*</td>
<td>&lt;180k</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

Incomplete values

Unspecified values

Intuitive interface for simple queries

Restricted to SQL syntax but not explicitly

SQL not required

Not example-based

[Zloof et al. 1975]
Similarities are the key …

If we knew how similar each item is with respect to any other for each user, we would know the answer to

Is there a galaxy like this?
Similarities are the key …

We define:
A universe $\mathcal{U}$ of items
A similarity among items $\sim$
A set of input examples $\mathcal{E}$
A set of output user desired answers $\mathcal{A}$
The example-based problem

Given

a set of examples $\mathcal{E}$ from a universe $\mathcal{U}$

Find

a similarity $\sim$ such that

1. $\mathcal{E}$ is part of the answers $\mathcal{A}$ partially or totally
2. The answers in $\mathcal{A}$ are the most similar to the examples in $\mathcal{E}$ according to $\sim$

How do we find $\sim$ for each user?
Do we need to know exactly $\sim$?
Example-based methods

Universe # → Desired Answers $ → Examples $ → Simplicity relation ~

Implicit (Unknown)
- Query Reverse Engineering
- Rule Discovery
- Relation Extraction

Explicit (Known)
- Structural Similarity
- Proximity Search
- Document Matching

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Example-based methods

**Relational**
- Reverse engineering queries
- Example-driven schema mapping
- Interactive data repairing

**Textual**
- Entity extraction by example text
- Web table completion using examples
- Search by example

**Graph**
- Community-based Node-retrieval
- Entity Search
- Path and SPARQL queries
- Graph structures as Examples
Tutorial structure

- Relational databases
- Textual data
- Graph and networks

Challenges and Remarks

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Where we are

- Relational databases
- Textual data
- Graphs and networks
- Challenges and Remarks

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Searching for …

**SEARCHING FOR**

**BY FOCUSING ON**

**APPLYING**

**PRODUCES**

**Tuples**

**Exact Queries**

- One-step

- Interactive
  - [Li et al., 2015a]

**Approximate Queries**

- Minimal
  - [Shen et al., 2014, Tran et al., 2014]

- Top-k
  - [Psallidas et al., 2015, Panev and Michel, '16]

**Matching rules**

**Schemas**

- Schema to examples
  - [Alexe et al., 2011a]

- Example-driven
  - [Alexe et al. 2011b, Cate et al. 2013, Gottlob Senellart, 2010, Bonifati et al., 2016]

**Data cleaning**

- Entity Mentions
  - [Singh et al. 2017]

- Data repairing
  - [He et al., 2016]

**Reverse Engineered SQL queries**

**Data integration**

**Duplicate matches**
Reverse engineering queries (REQ)

Given a set of examples, find the query that generated that set of tuples

Example tuples

How do you find such queries?

SELECT g.galaxy_name, SUM(s.stars) AS st_s
FROM Universe.Galaxy AS g
JOIN Universe.System AS s
ON g.galaxy_name = s.galaxy_name
WHERE
  g.st_s > 100B
  AND diameter > 100k AND diameter > 180k
  AND has_black_hole = TRUE
GROUP BY g.galaxy_name

SELECT galaxy_name
FROM Universe.Galaxy

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Query by Output – TALOS (classification-based) [Tran et al. 2013]

Main idea: Find the set of queries that exactly return a set of examples

Two queries Q and Q’ are instance equivalent on a database D, if the results of Q are the same of the results of Q’

Query Q → Query by Output → Reverse engineered Queries Q’
How many reverse engineered queries?

Master

<table>
<thead>
<tr>
<th>name</th>
<th>bat</th>
<th>throw</th>
<th>stint</th>
<th>weight</th>
<th>team</th>
</tr>
</thead>
<tbody>
<tr>
<td>t1</td>
<td>A</td>
<td>L</td>
<td>R</td>
<td>2</td>
<td>PIT</td>
</tr>
<tr>
<td>t2</td>
<td>A</td>
<td>L</td>
<td>R</td>
<td>2</td>
<td>MT1</td>
</tr>
<tr>
<td>t3</td>
<td>C</td>
<td>R</td>
<td>L</td>
<td>2</td>
<td>CHA</td>
</tr>
<tr>
<td>t4</td>
<td>D</td>
<td>L</td>
<td>R</td>
<td>3</td>
<td>PIT</td>
</tr>
<tr>
<td>t5</td>
<td>B</td>
<td>R</td>
<td>R</td>
<td>1</td>
<td>PIT</td>
</tr>
<tr>
<td>t6</td>
<td>B</td>
<td>R</td>
<td>R</td>
<td>1</td>
<td>PIT</td>
</tr>
<tr>
<td>t7</td>
<td>E</td>
<td>R</td>
<td>R</td>
<td>3</td>
<td>CHA</td>
</tr>
</tbody>
</table>

What queries generated Q(D)?

Q1 = SELECT name, team FROM Master WHERE bat = 'R' AND throw = 'R'
Q2 = SELECT name, team FROM Master WHERE bat = 'R' AND weight > 35
Q3 = SELECT name, team FROM Master WHERE bat = 'R' AND stint <> 2
...

Instance Equivalent Queries

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Idea: treat the problem as a binary classification

1. **Strict**: all tuples must be captured
2. **At-Least-one**: one tuple for example must be captured

\[
Gini(S_1, S_2) = \frac{|S_1|Gini(S_1) + |S_2|Gini(S_2))}{|S_1| + |S_2|}
\]

<table>
<thead>
<tr>
<th>name</th>
<th>bat</th>
<th>throw</th>
<th>stint</th>
<th>HR</th>
<th>team</th>
</tr>
</thead>
<tbody>
<tr>
<td>t1</td>
<td>A</td>
<td>L</td>
<td>R</td>
<td>2</td>
<td>40</td>
</tr>
<tr>
<td>t2</td>
<td>A</td>
<td>L</td>
<td>R</td>
<td>2</td>
<td>50</td>
</tr>
<tr>
<td>t3</td>
<td>C</td>
<td>R</td>
<td>L</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>t4</td>
<td>D</td>
<td>L</td>
<td>R</td>
<td>3</td>
<td>30</td>
</tr>
<tr>
<td>t5</td>
<td>B</td>
<td>R</td>
<td>R</td>
<td>1</td>
<td>73</td>
</tr>
<tr>
<td>t6</td>
<td>B</td>
<td>R</td>
<td>R</td>
<td>1</td>
<td>40</td>
</tr>
<tr>
<td>t7</td>
<td>E</td>
<td>R</td>
<td>R</td>
<td>3</td>
<td>60</td>
</tr>
</tbody>
</table>

Decision tree
How complex is exact REQ?

[Weiss et al., 2017]

Relational Operators:
- \( \sigma \) selection \( \{=, \neq, \geq, \leq\} \)
- \( \pi \) projection
- \( \bowtie \) natural join

Database \( D \)

\( E^+ \) Positive examples
\( E^- \) Negative examples

REQ

\( Q \) such that results contain
- All positive examples
- No negative example

How difficult is to find:
- A bounded size \( Q \)?
- An unbounded \( Q \)?
### Complexity - No parameters

[Weiss et al., 2017]

<table>
<thead>
<tr>
<th>Operator</th>
<th>Unbounded Queries</th>
<th>Bounded Queries</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\pi$</td>
<td>P</td>
<td>P</td>
</tr>
<tr>
<td>$\Join$</td>
<td>P</td>
<td>NPC</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>P</td>
<td>NPC</td>
</tr>
<tr>
<td>$\sigma, \Join$</td>
<td>P</td>
<td>NPC</td>
</tr>
<tr>
<td>$\pi, \sigma$</td>
<td>NPC</td>
<td>NPC</td>
</tr>
<tr>
<td>$\sigma, \Join$</td>
<td>DP</td>
<td>DP</td>
</tr>
<tr>
<td>$\pi, \sigma, \Join$</td>
<td>DP</td>
<td>DP</td>
</tr>
</tbody>
</table>

**Only projections:** Easy

**Unbounded selections:** Easy

**Bounded selections:** HARD

**Combination of operators:** HARD!!!

Reduction from SAT

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
### Unbounded Select

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>✗</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>✗</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>✓</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>✓</td>
<td>1</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

Possible queries:

- \( A = 1 \) AND
- \( B \geq 1 \) AND \( B \leq 5 \) AND
- \( C \geq 2 \) AND \( C \leq 4 \) AND
- \( D \geq 1 \) AND \( D \leq 4 \) AND \( D \neq 3 \)
- \( E \geq 3 \) AND \( E \leq 5 \) AND \( E \neq 4 \)

[Weiss et al., 2017]

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Bounded select

INPUT: Database D, Examples E, Query size k
OUTPUT: Does there exist a query satisfying D and E, of size at most k?

U = \{1,2,3,4,5\} \quad S = \{ \{1,2,3\}, \{2,4\}, \{3,4\}, \{4,5\} \}
Minimal Project Join REQ

Main idea: Find the set of queries that approximately return a set of examples

Partial query table

- **valid**: every tuple is present in query results
- **minimal**: any removal in query tree gets to an invalid query

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mike</td>
<td>Office</td>
</tr>
<tr>
<td>2</td>
<td>Mary</td>
<td>iPad</td>
</tr>
<tr>
<td>3</td>
<td>Bob</td>
<td>Dropbox</td>
</tr>
</tbody>
</table>
Candidate Query Generation

- Use candidate network generation algorithm (Hristidis 2002)

1. Generate join tree $J$
2. Generate mapping $\phi$
3. Check minimal:
   - Every leaf node contains a column that is mapped by an input column
Validity verification

[Shen et al., 2014]

- Naïve: check all candidate queries singularly if they return ALL examples
- Better: exploit substructures in candidate queries for pruning
- Best: adaptively select the substructures to have the min number of evaluations
  - NP-hard

Candidate query:

Sub 1

Sub 2

Sub 1 fails => Sub 2 fails

Sub 1 fails => \( CQ_2 \) invalid
Minimal Project Join REQ

Main idea: Allow missing rows/columns and rank the k best queries

**Partial query table**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>John</td>
<td>Smith</td>
<td>Xbox</td>
</tr>
<tr>
<td>2</td>
<td>Jill</td>
<td>Hans</td>
<td>Surface</td>
</tr>
</tbody>
</table>

Output: Top-k PJ Queries

[Psallidas et al., 2015]
Ranking score

Linear combination of row score and column score
(Overlapping with the example table)

\[
\alpha \cdot \text{score}_{\text{row}}(Q) + (1 - \alpha) \cdot \text{score}_{\text{col}}(Q) \over |Q|
\]

Row score

<table>
<thead>
<tr>
<th>Name</th>
<th>First Name</th>
<th>Last Name</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Smith</td>
<td>Xbox</td>
<td>3</td>
</tr>
<tr>
<td>Jill</td>
<td>Hans</td>
<td>Surface</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>5</td>
</tr>
</tbody>
</table>

Column score

<table>
<thead>
<tr>
<th>Name</th>
<th>First Name</th>
<th>Last Name</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Smith</td>
<td>Xbox</td>
<td>2</td>
</tr>
<tr>
<td>Jill</td>
<td>Hans</td>
<td>Surface</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

\[ \alpha = 1 \text{ penalizes missing rows} \]
\[ \alpha = 0 \text{ penalizes missing columns} \]
Interactive REQ – Query from Examples (cost model)

Main idea: Interactively remove candidate queries proposing a new set of query results from a modified database

[Li et al., 2015]
Database Refinement

REQs =
- $Q_1 = \sigma_{\text{gender}=M}(D)$
- $Q_2 = \sigma_{\text{salary}>3700}(D)$
- $Q_3 = \sigma_{\text{dept}=IT}(D)$

[Li et al., 2015]

Results

<table>
<thead>
<tr>
<th>name</th>
<th>gender</th>
<th>dept</th>
<th>salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>F</td>
<td>Sales</td>
<td>3700</td>
</tr>
<tr>
<td>Bob</td>
<td>M</td>
<td>IT</td>
<td>4200</td>
</tr>
<tr>
<td>Carol</td>
<td>F</td>
<td>Service</td>
<td>3000</td>
</tr>
<tr>
<td>Dave</td>
<td>M</td>
<td>IT</td>
<td>5000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>name</th>
<th>gender</th>
<th>dept</th>
<th>salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alice</td>
<td>F</td>
<td>Sales</td>
<td>3700</td>
</tr>
<tr>
<td>Bob</td>
<td>M</td>
<td>IT</td>
<td>4200</td>
</tr>
<tr>
<td>Carol</td>
<td>F</td>
<td>Service</td>
<td>3000</td>
</tr>
<tr>
<td>Dave</td>
<td>M</td>
<td>IT</td>
<td>5000</td>
</tr>
</tbody>
</table>

Result $R'_1 = Q_1(D') = Q_3(D')$
Result $R'_2 = Q_2(D')$
Cost model

\[
\text{cost}(D') = \text{edit}(D, D') + \beta \cdot n + \sum_{i=1}^{k} \text{edit}(R, R_i) + N \cdot \frac{\text{edit}(D, D')}{\mu} + \beta + \frac{2}{k} \sum_{i=1}^{k} \text{edit}(R, R_i)
\]

Main idea: Find a refined db D’ and results R₁, ... Rₖ with:
1. Minimum number of results k
2. Minimum differences i the database
3. The query are balanced (less interactions)
Examples for query suggestion: Blaeu (Clustering)

Main idea: Allow interactive navigation of the query space in a hierarchy

[Sellam et al., 2016]
Examples for query suggestion: Blaeu

[Sellam et al., 2016]

Given a result of an example query Q, explore the data through data maps = partitions

Output: Set of query refinements

Problem: User utility is unknown

- Cluster analysis for result exploration
- Zoom and projection operations
- User model

\[ u: DB \rightarrow \{-1,1\}, U(Q) = \sum_{t\in Q} u(t) \]
Examples for query suggestion: Blaeu

Find the partition $\mathcal{C} = \{C_1, \ldots, C_n\}$ of the results of $Q$ such that exists $C_j \in \mathcal{C}: U(C_j) > U(Q)$

Solution: interesting tuples are close to each other within a maximum separation threshold $\theta(\mathcal{C})$

Detect clusters (k-medoid)

Organize clusters

Inference

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Searching for ...

**SEARCHING FOR**

**BY FOCUSING ON**

**APPLYING**

**PRODUCES**

**Tuples**

Exact Queries

One-step


Interactive

[Li et al., 2015a]

Approximate Queries

Minimal

[Shen et al., 2014, Tran et al., 2014]

Top-k

[Psallidas et al., 2015, Panev and Michel, '16]

**Value Pairs**

Schemas

Schema to examples

[Alexe et al., 2011a]

Example-driven

[Alexe et al. 2011b, Cate et al. 2013, Gottlob Senellart, 2010, Bonifati et al., 2016]

**Values**

Data cleaning

Entity Mentions

[Singh et al. 2017]

Data repairing

[He et al., 2016]

Duplicate matches

**Reverse Engineered SQL queries**

**Data Integration**

**SIGIR 2019 tutorial**

https://data-exploration.ml

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Data Cleaning

- Often data have redundancy, wrong values, and missing values
- Different values can represent the same object (e.g., N.Y. and New York)
- Values can be simply wrong

Data cleaning refers to ways of making the data consistent and correct

<table>
<thead>
<tr>
<th>tid</th>
<th>Date</th>
<th>Molecule</th>
<th>Laboratory</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>t1</td>
<td>11 Nov</td>
<td>C_{16}H_{16}Cl</td>
<td>Austin</td>
<td>200</td>
</tr>
<tr>
<td>t2</td>
<td>12 Nov</td>
<td>statin</td>
<td>Austin</td>
<td>100</td>
</tr>
<tr>
<td>t3</td>
<td>12 Nov</td>
<td>C_{24}H_{75}S_{6}</td>
<td>N.Y.</td>
<td>100</td>
</tr>
<tr>
<td>t4</td>
<td>12 Nov</td>
<td>statin</td>
<td>Boston</td>
<td>200</td>
</tr>
<tr>
<td>t5</td>
<td>13 Nov</td>
<td>statin</td>
<td>Austin</td>
<td>200</td>
</tr>
<tr>
<td>t6</td>
<td>15 Nov</td>
<td>C_{17}H_{20}N</td>
<td>Dubai</td>
<td>1000</td>
</tr>
</tbody>
</table>
Data repairing: rules

A rule is a logical formula which determines how to change the value in a cell or a group of cells.

IF \( X_1 = C_1 \ldots X_n = C_n \) UPDATE \( X_i \) to some value

- The update \( t_3 \) [Laboratory] ← ”New York” can be obtained by the rule
- IF [Laboratory = ”N.Y.”] UPDATE Laboratory to ”New York”
- UPDATE Table
  SET Laboratory='New York'
  WHERE tid=t3

  **BUT it needs to be done for each cell!!**
Discovering rules

UPDATES:

\( \Delta_1 \): \( t3[\text{Laboratory}] \leftarrow \text{“New York”} \)
\( \Delta_2 \): \( t6[\text{Quantity}] \leftarrow 100 \)
\( \Delta_3 \): \( t2[\text{Molecule}] \leftarrow \text{“C}_{22}\text{H}_{28}\text{F”} \)

Some rules for \( \Delta_1 \):
1. Change all Laboratory values to “New York” (t1 – t6)
2. Reformatting all “N.Y” to “New York” (t3)

Some rules for \( \Delta_2 \):
1. Update the quantity to 100 if the molecule is \( \text{C}_{17}\text{H}_{20}\text{N} \) and the date is 15 Nov (t6)

Some rules for \( \Delta_3 \):
1. Update to \( \text{“C}_{22}\text{H}_{28}\text{F”} \) if molecule is statin (t2,t4,t5)
2. Update to \( \text{“C}_{22}\text{H}_{28}\text{F”} \) if molecule is statin and Laboratory Austin (t2,t5)
3. Update to \( \text{“C}_{22}\text{H}_{28}\text{F”} \) if molecule is statin and lab is Austin and date is 12 Nov and quantity is 100 (t2)
Interactive data cleaning: problem

User validates rules, but has no capacity to validate all rules for each update.

- **Budget Repair Problem:** Given a set $Q$ of rules, a table $T$ and a budget $B$, find $B$ rules from $Q$ to maximize the number of repairs over $T$

- Budget repair problem is an *online problem*

Corresponding *offline problem* is: given as input $Q$ rules where validity of each rule is known, select $B$ rules from $Q$ to maximize the number of repairs over $T$. *(NP-Hard)*
Rule lattice

More specific

Number of tuples affected

More general

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Lattice pruning

1. If $Q$ is valid, $Q'$ is also valid if $Q' \preceq Q$
2. If $Q$ is invalid, $Q''$ is also invalid if $Q \preceq Q''$
3. If $Q$ is valid, all $Q'$ such that $Q' \preceq Q$ are valid.
4. If $Q$ is invalid, all $Q''$ such that $Q \preceq Q''$ are invalid.
Lattice pruning
Dive search

- Binary Search over the lattice, ordering with #affected tuples
- If $T \rightarrow \text{BinarySearch}(Q_\land)$
- If $F \rightarrow \text{BinarySearch}(Q_\lor)$

Algorithm complexity $\mathcal{O}(B|Q| \log|Q|)$
Searching for …

**SEARCHING FOR**

**BY FOCUSING ON**

**APPLYING**

**PROroduces**

---

**Tuples**

**Exact Queries**
- One-step
- Interactive
  - [Li et al., 2015a]

**Approximate Queries**
- Minimal
  - [Shen et al., 2014 Tran et al., 2014]
- Top-k
  - [Psallidas et al., 2015 Panev and Michel, '16]

**Value Pairs**

**Schemas**
- Schema to examples
  - [Alexe et al., 2011a]
- Example-driven
  - [Alexe et al. 2011b, Cate et al. 2013, Gottlob Senellart, 2010 Bonifati et al., 2016]

**Values**

**Data cleaning**
- Entity Mentions
  - [Singh et al. 2017]

**Data Integration**
- Reverse Engineered SQL queries

**Duplicate matches**
- Data repairing
  - [He et al., 2016]
Schema mapping

- **Schema mapping** finds a way to represent items on one database to items on another database.
- Finds a mapping $\Sigma$ between two schemas such that a query on one database can be converted to a query on the other database.
- Schema mappings in $\Sigma$ are rules in first-order logic that specifies the relationships between schema $S$ and $T$:

$$\forall x \forall y \ S(x, y) \land U(x, z) \rightarrow \exists v \ T(v, y) \land T'(v, z)$$

![Diagram showing schema mapping between Source S and Target T](https://data-exploration.ml)
A Data Exchange Example

[Popa et al. 2001]

Projects
- Code: E-services
  - Funds
    - Fid
    - FinId
    - G3
      - ???
  - Code: PIX
    - Funds
      - Fid
      - FinId
      - G1
        - ???
      - G2
        - ???

Finances
- FinId
- MPhone
- Company
  - ??? 3608679 ???
  - ??? 3608776 ???
  - ??? 3608600 ???

Companies
- Coid
- Name
  - ??? AT&T
  - ??? Lucent

project(NA, ST), grant(gid, na, re, ma, su), contact(ma, em, ph) →

project(NA, FUND), fund(gid, finId), finance(finId, ph, company),

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
### Mapping generation

#### [Bonifati et al. 2017]

<table>
<thead>
<tr>
<th>Company</th>
<th>Flight</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E_S ):</td>
<td>( E_T ):</td>
</tr>
<tr>
<td>( IdCompany )</td>
<td>( Departure )</td>
</tr>
<tr>
<td>'C1'</td>
<td>'Lyon'</td>
</tr>
<tr>
<td>'C2'</td>
<td>'Paris'</td>
</tr>
<tr>
<td>Name</td>
<td>'Ev'</td>
</tr>
<tr>
<td>Town</td>
<td>'Paris'</td>
</tr>
</tbody>
</table>

#### Travel Agency

<table>
<thead>
<tr>
<th>IdAgency</th>
<th>Name</th>
<th>Town</th>
</tr>
</thead>
<tbody>
<tr>
<td>'A1'</td>
<td>'TC'</td>
<td>'L.A.'</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Firm</th>
<th>Departure</th>
<th>Arrival</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Id )</td>
<td>( Town )</td>
<td>( IdFirm )</td>
</tr>
<tr>
<td>'Id1'</td>
<td>'Lyon'</td>
<td>'Id1'</td>
</tr>
<tr>
<td>'Id2'</td>
<td>'Paris'</td>
<td>'Id2'</td>
</tr>
<tr>
<td>'Id3'</td>
<td>'Ev'</td>
<td></td>
</tr>
<tr>
<td>Town</td>
<td>'Lyon'</td>
<td></td>
</tr>
</tbody>
</table>

\( m : Company(c_1, aa, paris) \land Company(c_2, ev, lyon) \land TravelAgency(a_1, tc, la) \land Flight(lyon, paris, c_1) \land Flight(paris, lyon, c_2) \land Firm(id_1, aa, paris) \land Departure(lyon, id_1) \land Arrival(paris, id_1) \land Firm(id_2, ev, lyon) \land Departure(paris, id_2) \land Arrival(lyon, id_2) \land Firm(id_3, tc, la) \)
Interactive Mapping

Input: set of examples \((E_S^1, E_T^1) \ldots (E_S^n, E_T^n)\)

1. **Normalization**
   - Question
   - Yes/No

2. **Atom refinement**
   - Question
   - Yes/No

3. **Join refinement**
   - Question
   - Yes/No

Output: refined mapping

[Bonifati et al. 2017]
Atom Refinement

Ask the user and refine the left part of the rule

Are the tuples Company(c1,aa,paris); Company (c2, ev, lyon) enough to produce Firm(id, aa, Paris); Departure (Lyon, id); Arrival(Paris, id)?
Atom Refinement

Ask the user and refine the left part of the rule

Are the tuples Company(c1,aa,paris); Flight (lyon, paris, c1) enough to produce Firm(id, aa, Paris); Departure (Lyon, id); Arrival(Paris, id)?
Where we are

- Relational databases
- Textual data
- Graphs and networks
- Challenges and Remarks

Machine learning

[Image: https://data-exploration.ml]
SIMILARITY for DOCUMENTS

Unstructured

Semi-Structured

Unstructured

Semi-Structured
SEARCHING FOR

Documents

BY LOOKING AT

Words
- Topic Models [Zhu and Wu. ’14]
- Segmentation [Papadimitriou et al. ’17]

Meta-Data
- Citation Graph Navigation [El-Arini et al. ’11, Jia and Saule ’17]
- Entity Linking [Bordino et al. ’13]

Semi-Structured information

Words
- Regular Expressions [Agichtein et al. ’00]
- Annotations [Hanafi et al. ’17]
- Entity Extraction [Ritter et al. ’15]

Web-Tables
- Entity Mentions [Wang et al. ’15]
- Schema Matching [Yakout et al. ’18]

APPLYING

Documents/Citations/Queries recommendations

PRODUCES

Relation Extraction
Document Matching

Entity Augmentation
Concept Expansion

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Document Search
Keyword Queries & Relevance

Keyword query: search text with text

“Action movie with magic”

Search documents containing those exact words

... a live action movie...

.... there is plenty of action...

... packed with action...

... Magic Mike is comedy movie ...

... in Harry Potter magic is everywhere..

Is this enough?

Identify “relevant words” and “relevant documents”

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Document Search

Relevant Keywords

**Relevance**: which keywords are more helpful in describing the content of the document?

**Relevance ≠ Frequency**

What keywords are more likely to be used to describe the document we want and not other documents?

1. Term-frequency: how many times the term appears in the document
2. Document-frequency: In how many documents the term appears

**TF-IDF**: Term Frequency Inverse Document Frequency

FEW SELECTED KEYWORDS IN THE USER QUERY

What keywords to choose?

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
TRADITIONAL SEARCH

EXPLORATORY SEARCH

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Documents as Examples

Exemplar documents
Set of exemplar documents rather than a set of keywords.

An entire document may contain more information!
It also contains more noise

Identify what makes them special, i.e., relevant

Example-based Document Search
Given a corpus of documents D, and a small set of relevant documents (D_{rel}), identify a set of answer documents D_A such that D_{rel} \subseteq D_A \subseteq D.

Model as a classification problem

Find me movies like these:

**PROBLEM:** MISSING NEGATIVE CLASS

Few positive examples and a large set of unknown.
What features can discriminate relevant and irrelevant?
Would be better to have some negative examples

Liu et al. [2003]
Text Classifiers
Using Positive and Unlabeled Examples

Positive Unlabeled learning
- a corpus of documents $D$,
- 2 Classes: relevant $\top$ & irrelevant $\bot$
- relevant documents ($D_{rel}$)
  $\forall d \in D_{rel}$. class$(d) = \top$
- Unlabeled documents $U = D - D_{rel}$

Goal:
- train a classifier $C : D \rightarrow \{\top, \bot\}$,
  to predict $\text{class}(u) \forall u \in U$.

Missing:
- To train $C$ we need examples for the negative class $\bot$

Algorithm 4.9 Document Classification with Positive and Unlabeled Data

<table>
<thead>
<tr>
<th>Input:</th>
<th>Relevant Documents $D_{rel} \subseteq D$, Unlabeled Documents $U \subseteq D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output:</td>
<td>Classifier $C$</td>
</tr>
<tr>
<td>1:</td>
<td>$D_{neg} \leftarrow \text{getNegativeSample}(U)$ \cite{Li and Liu 2003, Liu et al. 2002, Yu et al. 2002}</td>
</tr>
<tr>
<td>2:</td>
<td>$C \leftarrow \text{trainClassifier}(D_{rel}, D_{neg}, U \setminus D_{neg})$ \cite{E.g., Expectation Maximization, SVM, or Rocchio}</td>
</tr>
<tr>
<td>3:</td>
<td>return $C$</td>
</tr>
</tbody>
</table>
Inferring Negative Examples

Assign a label to Unlabeled data:
how to determine a negative sample set without asking the user

4 Alternative approaches

- **Naïve Bayes** (McCallum et al. [1998])
  - All unlabeled data are assumed negatives
  - NB-Classifier estimates \( P(c|d) \) based on on \( P(w|c) \)
    with \( c \in \{ \top, \bot \} \), \( d \in D \), and words \( w \in W \)

- The **Rocchio** technique (Raskutti et al. [2002])
  - For \( d \in D \), \( \vec{d}^* \) is the TF-IDF vector representation
  - Build prototype vectors \( \vec{c}_\top \) for documents in \( D_{rel} \)
  - and \( \vec{c}_\bot \) for documents in \( U \)
  - Compare each \( \forall d \in U \) with \( \vec{c}_\top \) and \( \vec{c}_\bot \)
  - assign the class of the most similar vector

\[
\vec{c}_\top = \alpha \frac{1}{|D_{rel}|} \sum_{d\in D_{rel}} \frac{\vec{d}}{||\vec{d}||} - \beta \frac{1}{|U|} \sum_{d\in U} \frac{\vec{d}}{||\vec{d}||}
\]

Goal:
Determine set of elements to be regarded as reliable negatives (RN)

Train a “simplistic” classifier

Liu et al. [2003]
Inferring Negative Examples (II)

Assign a label to Unlabeled data:
how to determine a negative sample set without asking the user

4 Alternative approaches

- The **Spy** technique (Liu et al. [2002])
  - Extract a sample $S$ from the positive example
  - Merge $S$ in $U$ (deploy the spies!)
  - Build NB classifier with EM
  - Determine threshold $t$ such that all spies are correctly classified
  - Document above the threshold are considered negative

- **1-DNF*** technique (Yu et al. [2002]).
  *Disjunctive Normal Form
  - Positive Example Based Learning
  - Get words $W_f \subset W$. $\text{freq}(w, D_{rel})/|D_{rel}| > \text{freq}(w, U)/|U|$
  - Remove from $U$ all documents containing any word in $W_f$

Goal:
Determine set of elements to be regarded as reliable negatives (RN)

Train a “simplistic” classifier

Liu et al. [2003]
Training the Expert Classifier

Exploit the partial-supervision

**Expert Classifier**

Builds on the result of the first step to train a much more sophisticated and precise classifier.

- **1-shot approach**
  - Use $D_{rel}$ and RN and train a classifier (SVM or EM)

- **Iterative approach**
  - Use $D_{rel}$ and RN and train a classifier $C_i$
  - Use $C_i$ and extract new negative documents $Q$
  - Add $Q$ to RN, train a new classifier $C_{i+1}$
  - Continue until no more negative documents are retrieved

  *Optionally* evaluate the last trained classifier over $D_{rel}$ and discard it if it performs poorly

---

Liu et al. [2003]

Methods perform *poorly* when the initial set of documents is very small

The Rocchio approach + EM is best for this case

Advanced models with TF-IDF or Topic models

Zhu et al. [2013] - Zhu and Wu [2014]

Beware of Class Imbalance!

*SMOTE*: Synthetic Minority Over-sampling Technique
Document Segmentation
Intention-based relatedness

Model documents as Composite Objects
Do not perform matching across the posts as a whole but across fragments of them that are written for the same intention

Intuition:
Different parts of the document
Have different Purposes:
• Provide background information
• Describe Problem
• Ask question…

I have an HP system with a RAID 0 controller and 4 disks in form of a JBOD. I would like to install Hadoop with a replication 4 HDFS and only 320GB of disk space used from every disc. Do you know whether it would perform ok or whether the partial use of the disk would degrade performance. Friends have downloaded the Cloudera distribution but it didn’t work. It stopped since the web site was suggesting to have 1TB disks. I am asking because I do not want to install Linux and then realize that my hardware configuration is not the right one.

Extra RAID disk drives seem to be the solution to my problem but does adding RAID drives requires a reformat and rebuild of the system to improve performance?

My boss gave me yesterday an HP Pavilion computer with Intel Matrix Storage System, a 320GB drive and Linux pre-installed. I am thinking to add an extra disk drive using a RAID 0 or 1. Can I do it without having to rebuild the entire system? I have already looked at the HP official web site for how to use a JBOD. But I have not found anything related to it.

My HP Pavilion stops working after 15 min of activity. I called our technical department but no luck. Despite the many calls, I did not manage to find a person with adequate knowledge to find out what is wrong. All they said is bring it up and we will see, which frustrated me. At the end I had the brilliant idea to move it to a cooler place and voila. No more problems.

Papadimitriou et al. [2017]

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Segmentation
Boundaries

Use text characteristics and identify points in which a significant variation of these characteristics occurs, and place a segmentation border there.

Communication means & Text Features

<table>
<thead>
<tr>
<th></th>
<th>Present</th>
<th>Past</th>
<th>Future</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tense ($CM_{tense}$)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subject ($CM_{subj}$)</td>
<td>I/we</td>
<td>you</td>
<td>it/they/(s)he</td>
</tr>
<tr>
<td>Style ($CM_{neg}$)</td>
<td>interrog.</td>
<td>negative</td>
<td>affirmative</td>
</tr>
<tr>
<td>Status ($CM_{passat}$)</td>
<td>passive</td>
<td>active</td>
<td></td>
</tr>
<tr>
<td>Part of Speech ($CM_{pos}$)</td>
<td>verb</td>
<td>noun</td>
<td>adj./adverb</td>
</tr>
</tbody>
</table>

Not good border

Good border

segment 2

b₁ b₂ b₃

segment 1

Sᵢ₋₂ Sᵢ₋₁ Sᵢ Sᵢ₊₁

Bottom-up approach
1. Start with single words as segments
2. Compute a Diversity Index in each segment
3. Merge segments with low diversity
Intention Clustering & Matching

Clusters are based on intention

Given a document $d_i$,
1. the system will **segment** $d_i$,
2. identify for each segment the **segments in the same cluster**
3. **aggregate the similarity** of those segments into a score for each document.

C1

I have an HP system with a RAID 0 controller and 4 disks in form of a JBOD. Extra RAID disk drives seem to be the solution to my problem but does adding RAID drives requires a reformat and rebuild of the system to improve performance?

I have already looked at the HP official web site for how to use a JBOD. But I have not found anything related to it.

C2

My HP Pavilion stops working after 15 min of activity. I called our technical department but no luck.

Linux pre-installed.

C3

Friends have downloaded the Cloudera distribution but it didn’t work. It stopped since the web site was suggesting to have 1TB disks.

I am thinking to add an extra drive using a RAID 0 or 1. Can I do it without having to rebuild the entire system?

Do you know whether it would perform ok or whether the partial use of the disk would degrade performance?

Despite the many calls, I did not manage to find a person with adequate knowledge to find out what is wrong.

Explore based on related topics linked to common goals

All they said is bring it to up and we will see, which frustrated me. At the end I had the brilliant idea to move it to a cooler place and voila. No more problems.

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Influence in Citation Networks
Document relevance based on influence

Citation Network
- Nodes are Authors and Papers
- Edges are Authorship and Citations
- Influence is based on connecting Paths

Advance Models
- El-Arini and Guestrin [2011] :
  - Condition influence on topics
    Iterate for each topic T: Select topic T, keep only papers relevant for T, compute connecting Paths.
  - Weight edges with Influence-Probability
- Jia and Saule [2017]
  - Enrich graph with Keywords & Venues

Just looking at citations and co-citations is not sufficient.

Start from a known document
Explore new related topics, authors, venues...

El-Arini and Guestrin [2011]
Jia and Saule [2017]
Traverse (Document) Networks

Personalized Page Rank

- Start from seed nodes, i.e. the documents $D_{rel}$
- Navigate towards locally connected nodes

Global Page Rank

Starting from a random node, traversing randomly, random restart point anywhere in the graph

Example based Exploration implies locality

CHALLENGE:

Identify meaningful transition probabilities

E.g., El-Arini and Guestrin [2011]
M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

Sublinear Algorithms for Personalized PageRank, with Applications – Ashish Goel et al

El-Arini and Guestrin [2011]
Jia and Saule [2017]
Serendipitous Search
Enhance document links with Entities and Query-logs

Bordino et al. [2013]

Input: Query/Document
Output: Queries

Document

Serendipitous Search

Exploit “lateral connections” in User Search Behaviors

Connected entities

Francisco Pizarro
Rafting excursion
Amazon river
...

Machu Picchu
America
Peru

Query Logs

Related topics potentially come to mind after consulting the page.

raffing excursion down the urubamba river
el dorado temple of sun
indios quechus
map of peru
sapa inca

Explantions:
- Seriependitous Search enhances document links by exploiting entities and query-logs.
- The diagram illustrates how connected entities and query logs can be used to expand searches.
- Key concepts include serendipity, connected entities, and related topics.
Entity Query Graph

Entity-Query graph from queries to entities and back

EQGraph Weighted Edges

1. query to query:
   \[ w_Q(q_i \rightarrow q_j) = w_{QFG}(q_i \rightarrow q_j) \]

2. entity to query
   \[ w_{EQ}(e \rightarrow q) = \frac{f(q)}{\sum_{q_i \in X_E(q)} f(q_i)} \]
   Queries in the same session
   Frequency-based approach

3. entity to entity
   \[ w_E(e_u \rightarrow e_v) = 1 - \prod_{i=1, \ldots, r} (1 - p_{q_i \rightarrow q_i}(e_u \rightarrow e_v)) \]
   The more queries entities share the higher the probability

Personalized PageRank to score suggested queries

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

Bordino et al. [2013]
Entity Mentions & Web-Tables
Documents & semi-structured information

In fact, the Chinese market has the most influential names of the retail and tech space – Alibaba, Tencent (collectively touted as BAT), and is betting big in the global industry space. The giants which are claimed to have a cut-throat competition with the U.S. in terms of resources and capital are positioning themselves to become the future platforms. The trio is also expanding in other countries and investing heavily in AI-based startups to leverage the power of AI.

Backed by such powerful initiatives and presence of these conglomerates, the market in APAC AI is forecast to be the fastest-growing, with an anticipated CAGR of 45% over 2018-2024.

To further elaborate on the geographical trends, North America has procured more than 50% of the global share in 2017 and has been leading the regional landscape of AI in the retail market. The U.S. has a significant credit in the regional trends with over 65% of investments (including M&As, private equity, and venture capital) in artificial intelligence technology. Additionally, the region is a huge hub for startups in tandem with the presence of tech giants such as Google, IBM, and Microsoft.

<table>
<thead>
<tr>
<th>Company</th>
<th>Contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>PayTalk</td>
<td>Francisco Chang</td>
</tr>
<tr>
<td>Earn More</td>
<td>Roland Mendel</td>
</tr>
<tr>
<td>Island Trading</td>
<td>Helen Bennett</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>State</th>
<th>Capital</th>
<th>Population</th>
<th>Largest City</th>
<th>Population</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alaska</td>
<td>Juneau</td>
<td>31,275</td>
<td>Anchorage</td>
<td>291,626</td>
</tr>
<tr>
<td>Alabama</td>
<td>Montgomery</td>
<td>205,764</td>
<td>Birmingham</td>
<td>212,237</td>
</tr>
<tr>
<td>California</td>
<td>Sacramento</td>
<td>466,488</td>
<td>Los Angeles</td>
<td>3,922,612</td>
</tr>
<tr>
<td>Connecticut</td>
<td>Hartford</td>
<td>124,775</td>
<td>Bridgeport</td>
<td>144,229</td>
</tr>
<tr>
<td>Delaware</td>
<td>Dover</td>
<td>36,047</td>
<td>Wilmington</td>
<td>70,851</td>
</tr>
<tr>
<td>Florida</td>
<td>Tallahassee</td>
<td>181,376</td>
<td>Jacksonville</td>
<td>821,784</td>
</tr>
<tr>
<td>Illinois</td>
<td>Springfield</td>
<td>116,250</td>
<td>Chicago</td>
<td>2,695,986</td>
</tr>
<tr>
<td>Kansas</td>
<td>Topeka</td>
<td>127,473</td>
<td>Wichita</td>
<td>294,836</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>State</th>
<th>Capital</th>
<th>Population</th>
<th>Largest City</th>
<th>Population</th>
</tr>
</thead>
<tbody>
<tr>
<td>North Dakota</td>
<td>Bismarck</td>
<td>178,097</td>
<td>Dickinson</td>
<td>1,241</td>
</tr>
<tr>
<td>Ohio</td>
<td>Columbus</td>
<td>1,528,291</td>
<td>Cleveland</td>
<td>357,414</td>
</tr>
<tr>
<td>Oregon</td>
<td>Salem</td>
<td>417,948</td>
<td>Portland</td>
<td>628,502</td>
</tr>
<tr>
<td>Pennsylvania</td>
<td>Philadelphia</td>
<td>1,585,976</td>
<td>Philadelphia</td>
<td>1,585,976</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>State</th>
<th>Capital</th>
<th>Population</th>
<th>Largest City</th>
<th>Population</th>
</tr>
</thead>
<tbody>
<tr>
<td>Texas</td>
<td>Austin</td>
<td>760,380</td>
<td>Houston</td>
<td>2,092,492</td>
</tr>
<tr>
<td>Vermont</td>
<td>Montpelier</td>
<td>61,682</td>
<td>Burlington</td>
<td>45,440</td>
</tr>
<tr>
<td>Virginia</td>
<td>Richmond</td>
<td>194,214</td>
<td>Virginia Beach</td>
<td>122,250</td>
</tr>
<tr>
<td>Washington</td>
<td>Olympia</td>
<td>464,781</td>
<td>Seattle</td>
<td>608,660</td>
</tr>
<tr>
<td>Wisconsin</td>
<td>Madison</td>
<td>233,209</td>
<td>Milwaukee</td>
<td>594,833</td>
</tr>
</tbody>
</table>
**Entity-relation tuples**

Example-based extraction of Entity mentions and Relations

1. **Example**
   
   ⟨ Google ; Menlo Park ⟩

2. **Match**
   
   Google *founded in* Menlo Park...

3. **Extract Pattern**
   
   … [X] *founded in* [Y] …

4. **Extract New Mentions & Patterns**
   
   Apple *founded in* Coupertino …
   
   Apple *headquarters in* Coupertino

---

**Search for Information WITHIN Documents**

**Explore new Entities and new ways to express relations**

---

**Snowball**

**Find Occurrences of Exemplar Tuples**

**Generate Extaction Patterns**

**Generate New Exemplar Tuples**

**Tag Entities**

**Augment Table**

**Goal: Enrich a list of Entity-relationships data**

---

**Brin [1998]**

**Agichtein and Gravano [2000]**

---

**Entity-relation tuples**

**Example-based extraction of Entity mentions and Relations**

---

**Works bests with Binary relation**

Can work with multiple mentions:

Bob *born in U.S.A. in* 1978

---

**Brin [1998]**

**Agichtein and Gravano [2000]**
Entity-relation tuples
Example-based extraction of Entity mentions and Relations

How to validate the new rules extracted automatically?

1. **Compare extracted rules with known tuples:** confidence of R is based on how many known tuples extracts

2. **Compare extracted tuples with known rules:** confidence of T is based on how many known rules also extract T

New extracted Rules and Tuples should not create contradictions

This approach has no “human in the loop”

Find Occurrences of Exemplar Tuples

Generate New Exemplar Tuples

Generate Extraction Patterns

Augment Table

Snowball

Tag Entities

Brin [1998]
Agichtein and Gravano [2000]

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

83
IN MY DEFENSE
I WAS LEFT UNSUPERVISED
Entity-extraction by Example
Learn extraction rules from example

Hanafi et al., [2017]

Allow to match from text both Positive and Negative examples

Goal: Supervised Extraction

Output: Extraction rules

Hanafi et al., [2017] allow to match from text both Positive and Negative examples.

**Goal:** Supervised Extraction

**Output:** Extraction rules

```
definition) increased 9.6 percent, the number of murders increased 6.2 percent, aggravated assaults increased 2.3 percent, the number of rapes (revised definition) rose 1.1 percent, and robbery violations were up 0.3 percent.

Violent crime increased in all but two city groupings. In cities with populations from 50,000 to 99,999 inhabitants, violent crime was down 0.3 percent, and in cities with 500,000 to 999,999 in population, violent crime decreased 0.1 percent. The largest increase in violent crime, 5.3 percent, was noted in cities with 250,000
```

**SEER**

- **P: Percentage** = 1.0
- **D: [5, 6]** = 0.4
- **D: {percent, %}** = 0.4
- **R: [0-9]+** = 0.2
- **D: {percent, %}** = 0.4

```
= 1.0
= 0.4
= 0.3
```
Matching Rules
From string tokens to “semantics”

Example: 5 percent up in Dubai

Each token may have different candidate “matching rules”

Intuition: Exploit a vocabulary of simple specialized patterns with known semantics

Each rule has a “class” and a preference score

Hanafi et al., [2017]

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

86
SIGIR 2019 tutorial
https://data-exploration.ml
Merging Rules
Reconcile multiple interpretations

Example: 5 percent

Tokens: 5 percent
Tree:
Rule: R: [0-9]+ = 0.2
L: 'percent' = 0.4
R: [A-Za-z]+ = 0.2
L: '5' = 0.4

P: Percentage = 1.0

Example: 6 %

Tokens: 6 %
Tree:
Rule: R: [0-9]+ = 0.2
L: '%' = 0.4
R: symbols = 0.2
L: '6' = 0.4

P: Percentage = 1.0

Intersection: [5 percent, 6%]

Consider also Negative Examples to prune candidates

Hanafi et al., [2017]

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

SIGIR 2019 tutorial
https://data-exploration.ml
Web Tables
Semi-structured data on the web

https://en.wikipedia.org/wiki/Denmark#Regions

Regions

The governing bodies of the regions are the regional councils, each with forty-one councillors elected for four-year terms. The council is headed by regional district chairmen (regionsrådsformanden), who are elected by the council.[79] The areas of responsibility for the regional councils are the national health service, social services and regional development.[79][80] Unlike the counties they replaced, the regions are allowed to levy taxes and the health service is partly financed by a national health care contribution until 2018 (sundhedsbidrag), part of which is paid from both government and municipalities.[18] From 1 January 2019 this contribution will be abolished, as it is being replaced by higher VAT instead.

The area and populations of the regions vary widely; for example, the Capital Region, which encompasses the capital city of Copenhagen, has a population of about 1.8 million and is the most densely populated region. The largest city in the Capital Region is Copenhagen, with a population of 1.8 million. The North Denmark Region is the largest region in terms of area, with a total area of 2,568.29 km².

<table>
<thead>
<tr>
<th>Danish name</th>
<th>English name</th>
<th>Admin. centre</th>
<th>Largest city (population)</th>
<th>Population (January 2017)</th>
<th>Total area (km²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hovedstaden</td>
<td>Capital Region of Denmark</td>
<td>Hillerød</td>
<td>Copenhagen</td>
<td>1,807,404</td>
<td>2,568.29</td>
</tr>
<tr>
<td>Midtjylland</td>
<td>Central Denmark Region</td>
<td>Viborg</td>
<td>Aarhus</td>
<td>1,304,253</td>
<td>13,095.80</td>
</tr>
<tr>
<td>Nordjylland</td>
<td>North Denmark Region</td>
<td>Aalborg</td>
<td>Aalborg</td>
<td>587,335</td>
<td>7,907.09</td>
</tr>
<tr>
<td>Sjælland</td>
<td>Region Zealand</td>
<td>Søllerød</td>
<td>Roskilde</td>
<td>832,553</td>
<td>7,266.75</td>
</tr>
<tr>
<td>Syddanmark</td>
<td>Region of Southern Denmark</td>
<td>Vejle</td>
<td>Odense</td>
<td>1,217,224</td>
<td>12,132.21</td>
</tr>
</tbody>
</table>

Source: Regional and municipal key figures

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Entity List Expansion
Augmentation: identify entities to complete the list

1. Input: Incomplete list + Keyword query
2. Retrieve tables from pages based on the keyword query
3. Assign Score to tables based on relevance
4. Extract entity mentions from tables
5. Analyze Entity mention co-occurrence
6. Pick "co-occurring" Entities

Go to Bipartite-graph

**Problem:** entities may appear together for different reasons

<table>
<thead>
<tr>
<th>IT Company</th>
<th>Dell</th>
<th>IBM</th>
<th>Lenovo</th>
<th>....?</th>
</tr>
</thead>
</table>

**Problem:** Here PPR Causes concept drift

<table>
<thead>
<tr>
<th>IT Company</th>
<th>Dell</th>
<th>IBM</th>
<th>Lenovo</th>
<th>Apple</th>
<th>Samsung</th>
<th>HP</th>
<th>Acer</th>
</tr>
</thead>
</table>

**Goal:** Given some seed entity mentions, retrieve more entities of the same type

- Wang et al. [2015]
- M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Web-Table Completion
Identify relevant content, retrieve missing information

Web tables

Model    | Brand
---------|-------
S80      | Benq
A10      |       
GX-1S    |       
T1460    |       

Incomplete table

Extra Input: table header
target attribute name or example of completing attribute

Model    | Brand
---------|-------
S80      | Benq
A10      | Innostream
GX-1S    | Samsung
T1460    | Benq

Complete table

Yakout et al. [2012]

Intuition: If there is a structure, we can match it!

Goal: Retrieve missing attribute values

Problem: entities may appear together for different reasons
Table Correlation Graph

Schema matching for web-page and web-tables
Binary-relations only

Determine Table Match

Direct Match between Q(K,A) and T(K,B)
K=entity names in a column
A,B = attribute column name (header)

\[ S_{DMA}(T) = \begin{cases} \frac{|T \cap_K Q|}{\min(|Q|,|T|)} & \text{if } Q.A \approx T.B \\ 0 & \text{otherwise} \end{cases} \]

Problem: considers only direct links between Q and T

Can use approximate matching and thesaurus

Goal: Retrieve missing attribute values

Yakout et al. [2012]

Companies developing database software

Database software, 2011 revenue by vendor

OS support for top database software
Table Correlation Graph

Schema matching for web-page and web-tables
Binary-relations only

**Determine Table Match**

Holistic Match
1. Assign Direct Match Score from Query to Tables
2. Scores >0 are starting nodes
3. Use classifier to add weight to other table pairs
   - **Build Classifier** using
     - Context similarity
     - Table-to-content similarity
     - URL similarity
     - Tuples Similarity
     the model predicts the match between two tables with a probability
4. Use starting node and execute PPR
5. Use PPR scores to rank matching tables

**Goal:** Retrieve missing attribute values

- **T1**
  - **Product**
    - MySQL
    - PostgreSQL
    - MongoDB
    - Berkeley DB
  - **Vendor**
    - Oracle corp.
    - PostgreSQL Grp
    - Oracle corp.

- **T2**
  - **Name**
    - MySQL
    - Oracle
    - Firebird
    - Berkeley DB
  - **Max Row Size**
    - 64Kb
    - 8Kb
    - 64Kb
    - 8Kb

- **T3**
  - **Name**
    - MySQL
    - SQL Server
    - Office
    - Photoshop
  - **Developer**
    - Oracle
    - Microsoft
    - Microsoft
    - Adobe

- **T4**
  - **Name**
    - Oracle
    - MySQL
    - SQL Server
  - **Windows**
    - Yes
    - Yes
    - No
  - **Linux**
    - Yes
    - Yes
    - Yes

- **T5**
  - **Vendor**
    - Oracle corp.
    - IBM
    - Microsoft
    - Teradata
  - **Software**
    - Oracle DB
    - DB2
    - Teradata
    - Teradata Corp.

- **T6**
  - **Vendor**
    - Oracle corp.
    - IBM
    - Microsoft
    - Teradata
  - **Software**
    - Oracle
    - DB2
    - Teradata Corp.

**Companies developing database software**

**Database software, 2011**

**Vendor**
- Oracle corp.
- IBM
- Microsoft
- Teradata

**Revenue by vendor**
- Oracle: 11787M
- IBM: 4870M
- Microsoft: 4098M
- Teradata: 882M

**List of Open Source database software**

**Information about database size limits**

**Overcomes problems due to poor matching with the query**

**M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis**
SEARCHING FOR

BY LOOKING AT

APPLYING

PRODUCES

Documents/Citations/Queries recommendations

Relation Extraction
Document Matching

Entity Augmentation
Concept Expansion

Entity Mentions
[Hanafi et al.’17]

Regular Expressions
[Agichtein et al.’00]

Annotations
[Hanafi et al.’17]

Entity Extraction
[Ritter et al.’15]

Schema Matching
[Yakout et al.’18]

Web-Tables

Documents/Citations/Queries recommendations

Relation Extraction
Document Matching

Entity Augmentation
Concept Expansion

Entity Mentions
[Hanafi et al.’17]

Regular Expressions
[Agichtein et al.’00]

Annotations
[Hanafi et al.’17]

Entity Extraction
[Ritter et al.’15]

Schema Matching
[Yakout et al.’18]

Web-Tables

Entity Extraction
[El-Arini et al.’11, Jia and Saule’17]

Entity Linking
[Bordino et al. ’13]

Segmentation
[Papadimitriou et al. ’17]

Topic Models
[Zhu and Wu. ’14]

Text Classifier
[Liu et al. ’03, Zhang and Lee’09, Zhu et al.’13]

Citation Graph Navigation
[El-Arini et al.’11, Jia and Saule’17]

Meta-Data

Words

Citation Graph Navigation
[El-Arini et al.’11, Jia and Saule’17]

Entity Linking
[Bordino et al. ’13]

Segmentation
[Papadimitriou et al. ’17]

Topic Models
[Zhu and Wu. ’14]

Text Classifier
[Liu et al. ’03, Zhang and Lee’09, Zhu et al.’13]

Citation Graph Navigation
[El-Arini et al.’11, Jia and Saule’17]

Entity Linking
[Bordino et al. ’13]

Segmentation
[Papadimitriou et al. ’17]

Topic Models
[Zhu and Wu. ’14]

Text Classifier
[Liu et al. ’03, Zhang and Lee’09, Zhu et al.’13]
Where we are

- Relational databases
- Textual data
- Graphs and networks
- Challenges and Remarks

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Graphs are Everywhere
Graphs

Connected Data

A graph is a graph is a graph

Arnold Schwarzenegger
actedIN

Terminator
Release 1984

Fact Graph

Ontology Tree

is A

is A

subClassOf

The Structure of the Graph is as important as the Data-values

Edge-labelled Multigraphs

G: \{V, E, L, ℓ\}

Attributes:
V/E: <key, value>

RDF (subject, predicate, object)

(Arnold_Schwarzenegger, isA, Person)
(Actor, subClassOf, Person)
(Arnold_Schwarzenegger, actedIn, Terminator)

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

SIGIR 2019 tutorial
https://data-exploration.ml
Exemplar Queries
Example-driven graph search

Input: \( Q_e \), an example element of interest

Output: set of elements in the desired result set

Exemplar Query Evaluation

• evaluate \( Q_e \) in a database \( D \), finding a sample \( S \)
• find the set of elements \( A \) similar to \( S \) given a similarity relation
• [OPTIONAL] return only the subset \( A^R \) that are relevant

Nodes/Entities
Edges/Facts
Structures

Mottin et al. [2014,2016]

Usually requires an intermediate step:
User input (keywords) → Element in the graph

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
SIMILARITY for GRAPHS

**Nodes**
- Connectivity

**Structures**
- Properties
- Queries
- (Edge-)Labels

**CHALLENGE: DISCOVER USER PREFERENCE**

**CHALLENGE: EFFICIENT SEARCH**
SEARCHING FOR

BY LOOKING AT

PRODUCES

Nodes

Connectivity
Mediator Nodes [Gionis et al. ‘15, Ruchansky et al. ‘15]
Clustering [Perozzi et al. ‘14, Kloumann et al. 14]

Properties
Similar Entities [Metzger et al. ‘13, Sobczak et al. ‘15]

Queries
Path Queries [Bonifati et al. ‘15]
SPARQL [Arenas et al. ‘16]

Structures
Entity Tuples [Jayaram et al. ‘15]
Similar Structures [Mottin et al. ‘14, Xie et al. ‘17, Lissandrini et al. ‘18]

SIGIR 2019 tutorial
https://data-exploration.ml
M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Seed Set Expansion
Nodes connected by a community

Given a graph $G$, and a set of query nodes $V_Q \subseteq V_G$, retrieve all other nodes $V_C \subseteq V_G$, where $C$ is a community in $G$, and $V_Q \subseteq V_C$.

Solution: PPR

$$v^{t+1} = (1 - \alpha)M \cdot v^t + \alpha v^0$$

Communities can be extremely large
Identify “central nodes” or “the core subgraph”

Kloumann and Kleinberg [2014]

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
The Minimum Wiener Connector Problem

Model: Unlabeled Undirected Graph
Query: A set of Nodes Q
Similarity: Shortest-Path distance
Output: A Set of Connector Nodes H

“explains” connections in Q

Connectors:
Nodes with HIGH closeness to ALL the inputs

Similar to a Steiner-Tree but overall pairwise distances are optimized

Case: Infected Patients
→ Culprit/Other Infected

Case: Target Audience
→ Influencers

Ruchansky et al. [2015]
The Minimum Wiener Connector Problem

Model: Unlabeled Undirected Graph
Query: A set of Nodes Q
Similarity: Shortest-Path distance
Output: A Set of Connector Nodes H

“explains” connections in Q

minimize the sum of pairwise shortest-path-distances between nodes in the connector H

Called: Wiener Index.

tradeoff between size and average distance

\[
\min \sum_{(u,v) \in H} d(u, v)
\]

\(d(u, v)\) is the shortest-path distance

Sometimes The Best Solution is NOT A Tree

NP-Hard

[1] Ruchansky et al. [2015]
Approximate minimum Wiener Index Connector

**Ruchansky et al. [2015]**

**Approximated with Edge-Weighted Steiner Tree**

Enumerate Candidate Solutions for $r \in Q \& \lambda$ and keep best tree

CHOOSE $r \in Q \& \lambda \in \left[ 1, \log_{(1+\beta)} |V| \right]$

All Pairwise Distances

- Distances from a root $r$

Measure distance in $H$ (i.e., subgraph-induced)

- Precomputed distance in $G$

Edge Weights

$$w(u, v) = \lambda + \frac{\max\{d_G(r, u), d_G(r, v)\}}{\lambda}$$
Focused Clustering and Outlier Detection

Similarity based on attributes

**Model:** Unlabeled Undirected Graph with Node Attributes

**Query:** A set of Nodes \( Q \)

**Similarity:** To Be Inferred

*based on Attribute Values & Connectivity*

**Output:** Clusters of Nodes: Dense & Coherent

+ Outliers

Case: Target Users → Community with same interests

Case: Products → Co-purchased products with similar features
Focused Clustering

Infer User Focus

TASK: Infer “FOCUS”, important attributes

attribute weights $\beta$

\[
\begin{pmatrix}
\text{PhD} \\
\text{NYC} \\
\text{English} \\
\text{Google}
\end{pmatrix}
\begin{pmatrix}
\text{PhD} \\
\text{NYC} \\
\text{French} \\
\text{SAP}
\end{pmatrix}
\rightarrow
\begin{pmatrix}
0.5 \\
0.5 \\
0 \\
0
\end{pmatrix}
\]

1. Set of similar pairs, PS (from Q)

2. Set of dissimilar pairs, PD (random sample)

3. Learn a distance metric between PS and PD

$$
\min_{A} \sum_{(u,v) \in P_S} (f_i - f_j)^T A (f_i - f_j) - \gamma \log \left( \sum_{(u,v) \in P_D} \sqrt{(f_i - f_j)^T A (f_i - f_j)} \right)
$$

( Distance Metric Learning, inverse Mahalanobis distance: Xing, et al 2002)
Focused Clustering
Prune the Graph and keep dense communities

**TASK: Extract Clusters on Focused Graph**

attribute weights $\beta \rightarrow$ Edge Weight

1. **Find Starting Set** of Small Candidate Clusters
   1.a Drop low-weight edges
   1.b Extract Strongly Connected Component $C_1, C_2, ...$

2. **Grow Clusters** around Candidates
   2.a Compute conductance of $C$: $\phi^{(w)}(C, G)$
   2.b Select node to add to $C'$: best improvement to $\Delta \phi^{(w)}(C, C')$ (greedy)
   2.c Prune Underperforming nodes

3. **Detect Outliers**: High unweighted conductance
   w.r.t. low weighted conductance

---

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
iQBEES: Entity Search by Example

Knowledge Graph Search

Model: Knowledge Graph (Edge-labels)

Query: A set of Entities Q

Similarity: Shared semantic properties

Output: A Set of Similar Entities (ranked)

Case: Products → Products with similar aspects

Case: Social Media → User recommendation
Maximal Aspects
Selecting Features of Entity Similarity

1. Prune generic aspects

2. Rank Set of aspects

Is not maximal if Adding any aspect → E(A)={Arnold}

Include Typical Types

Use most Specific Type

REPEATABLE Update Q

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis

https://data-exploration.ml
SIMILARITY for GRAPHS

Queries can retrieve both Nodes and Structures
Nodes

Connectivity
Mediator Nodes
Clusters
[Perozzi et al. ’14, Kloumann et al. 14]

Properties
Similar Entities
[Metzger et al.’13, Sobczak et al.’15]

Structures
Queries
Path Queries
[Bonifati et al.’15]
SPARQL
[Arenas et al.’16]

(Edge-)Labels
Entity Tuples
[Jayaram et al.’15]
Similar Structures
[Mottin et al.’14, Xie et al.’17, Lissandrini et al’18]

SEARCHING FOR
BY LOOKING AT
PRODUCES
**Learning Path Queries on Graphs**

Queries from Examples

**Model:** Edge Labeled Graph

**Query:** 2 sets of Entities $Q^+$, $Q^-$

**Similarity:** Common Path Query (RegExp)

$q := \epsilon | a (a \in \Sigma) | q_1 + q_2 | q_1 \cdot q_2 | q^*$

(bus|tram)*+ Cinema

**Output:** Set of Nodes satisfying paths for $Q^+$

but not paths for $Q^-$

Case: Proteins $\rightarrow$ Similar interactions/co-expression

Case: Tasks Initiator $\rightarrow$ Similar Processes/Behaviours

---

**MONADIC:** only starting nodes

extensible to

BINARY/ N-ARY : path from X to Y
Learnability of Path Queries
When is possible and How

Query: 2 sets of Entities $Q^+$, $Q^-$

Consistency:
1. Select Smallest Consistent Path
   $\forall v \in Q^+, paths_G(v) \subseteq paths_G(Q^-)$

2. Loops cause infinite paths? Fix Maximal Length $K$
   When to use Kleene star $\ast$?

   $C | (A \cdot B \cdot C) \rightarrow (A \cdot B)^* \cdot C$

3. Generalize SCP
   a) Construct Prefix Tree Acceptor
   b) Generalize into DFA with Merge

Can be INTERACTIVE! The system presents to the user nodes to label as Positive/Negative

For paths of Length $N$
$K = 2 \times N + 1$

Sometimes Positive & Negative Examples Cannot be reconciled!
Reverse engineering SPARQL queries

Knowledge Graph Search

Model: Knowledge Graph (Edge-labels)

Query: Set of Answers → Not Graphs but Tuples (of Nodes?)

Similarity: common AND/OPT/FILTER query

Output: a SPARQL query / query results

Arenas et al. [2016]

Case: Open Data → Query Unknown Schema

Case: Novice User → Avoid SPARQL
Complex SPARQL queries
A quick-peek to the complex pattern queries

```
SELECT * WHERE {
  ?deal a :Deal ;
  :employee ?employee ;
  :customer ?customer .
  ?employee :name ?employeeName ;
  :involvedAsEmployee ?matter .
  ?customer :name ?customerName ;
  :involvedAsCustomer ?matter .
}
```

Variables start with ?
Reverse engineering SPARQL queries

Challenges and Complexity

<table>
<thead>
<tr>
<th>?X</th>
<th>?Y</th>
<th>?Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>John</td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>Mary <a href="mailto:mary@email.eu">mary@email.eu</a></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td>Lucy Roses Street</td>
<td></td>
</tr>
</tbody>
</table>

Query: Set of Variable Mappings

- Enumerate all possible SPARQL queries satisfied by the mappings
- Build tree-shaped SPARQL queries IMPLIED by the mappings

Incomplete Mappings are treated as OPTIONAL

Typical of RDF queries

Arenas et al. [2016]

\[
\Sigma^p_2 - \text{complete}
\]
Reverse engineering SPARQL queries

Challenges and Complexity

Query: Set of Variable Mappings $\Omega$

<table>
<thead>
<tr>
<th>$\Omega$</th>
<th>$?X$</th>
<th>$?Y$</th>
<th>$?Z$</th>
<th>$?W$</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>a1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>a2</td>
<td>b2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M3</td>
<td>a3</td>
<td>c3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>M4</td>
<td>a4</td>
<td>b4</td>
<td>c4</td>
<td>d4</td>
</tr>
</tbody>
</table>

Greedy: keep just enough to cover all variables

3 Instantiations:
1. Only Positive Examples
2. Positive & Negative
3. Exact Result only
SEARCHING FOR

BY LOOKING AT

PRODUCES

Nodes

Connectivity
Mediator Nodes
[Gionis et al. ‘15
Ruchansky et al.’15]

Clusters
[Perozzi et al.’14,
Kloumann et al. 14]

Properties
Similar Entities
[Metzger et al.’13,
Sobczak et al.’15]

Structures

Queries
Path Queries
[Bonifati et al.’15]

SPARQL
[Arenas et al.’16]

(Edge-)Labels
Entity Tuples
[Jayaram et al.’15]

Similar Structures
[Mottin et al.’14,
Xie et al.’17,
Lissandrini et al’18]
Graph Exemplar Queries
Search for Structures

Model: Knowledge Graph
Query: Example Structure
Similarity: Isomorphism/Simulation
Output: A set of Sub-Graphs

Case: Rich Schema → Find complex structures
Graph Isomorphism vs. Simulation

Variants

Structural Congruence/Similarity

Isomorphism requires an **bijective function**
Simulation requires only a **surjective relation**
Preserves only Parent $\rightarrow$ Child relationships

Example of **Simulating** (G1 $\sim$ {G2,G3,G4}) and **Strong-simulating** Graphs (G1 $\approx$ G2)

Strong Simulation preserves close connectivity

Strong simulation: Capturing topology in graph pattern matching
– Shuai Ma et al., 2014

– M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Computing Exemplar Queries (i)
Fast Structure Matching

### Exact Pruning technique:
- Compute the neighbor labels of each node
  \[ W_{n,a,i} = \{n_1 | l(n_1, n_2) = a \: \forall \: n \in N_{i-1}(n) \} \]
- Prune nodes not matching query nodes neighborhood labels
- Apply iteratively on the query nodes

\[ \text{neighborhood} (v) = \{(B,1)\} \]
\[ \text{neighborhood} (u) = \{(A,1)\} \]

**Reduce Search Space:**
Removes nodes that cannot be part of a solution

NP-complete (subgraph isomorphism)
\( \mathcal{O}(|V|^4) \) (simulation)
**Computing Exemplar Queries (ii)**

**Prune Irrelevant Answers**

**Approximation:**
- Nodes closed to the sample are more important
- Use **Personalized PageRank** with a weighted matrix

\[ v = (1 - c)A v + c p \]

- Weight edges: **frequency of the edge-label**

\[ I(e^f_{ij}) = I(\ell) = \log \frac{1}{P(\ell)} = -\log P(\ell) \]

\[ P(\ell) = \frac{|E^f|}{|E|} \]

**Reduce Search Space:**
Removes nodes that are likely to be less relevant
Ranking Results
Score Relevance of Answers

Combination of two factors

1. Structural: similarity of two nodes in terms of neighbor relationships
2. Distance-based: the PageRank already computed
Search with Multiple Examples
Combining partial answers

- Multiple Simple Examples
- Each Example describes an Aspect
- Results are Combinations of aspects
- Results have possibly Multiple Structures

Case: Unknown Structures → Find Complex Connections with Simpler Components
Search Framework
Pruning and Partial matching

Multi-exemplar Answering

Input: Database $G : (V, E, \ell)$
Input: Samples $S : \{s_1, \ldots, s_m\}$
Output: Answers $A$

1: $\bar{G} \leftarrow \text{PARTIAL}(G, S)$
2: $A \leftarrow \text{SEARCH}(\bar{G}, S)$
3: return $A$

Exploit Localized Search

Lissandrini et al. [2018]

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Fast Candidate Region Search
Reducing the search space

Identify SEED:
With cardinality Estimation

Select SINGLE NODE
With neighborhood-mapping

EXPAND around each seed:
Retrieve candidate Regions

DISCARD incomplete regions
With neighborhood-mapping & before graph-search
https://www.youtube.com/watch?v=A1_dKvX5ZRk
Graph Query by Example (GQBE)

Search for example Tuples

Model: Knowledge Graph
Query: Entity Tuples
Similarity: ~Isomorphism
Output: A set of Tuples

In GQBE Input is a set of (disconnected) entity mention tuples

Case: Known Entities + Unknown Connections → Find Complex Connections

Jayaram et al. [2015]

Q = (Google, S. Mateo)
Results = (Yahoo, S. Clara) (CBS, New York)
**GQBE: Maximum Query Graph**

Understand the connections implied by the tuples

1. Find the maximum query graph
   - Graph with $M$ edges having the maximum weight

2. Answers subgraph-isomorphic to the query graph  
   - NP-hard

3. Return top-k

Answer score:
- Sum of query graph weights
- Similarity match between edges in the answer and the query (shared nodes take extra credit)

$$\text{match}(e, e') = \begin{cases} 
\frac{w(e)}{|E(u)|} & \text{if } u = f(u) \\
\frac{w(e)}{|E(v)|} & \text{if } v = f(v) \\
\frac{w(e)}{\min(|E(u)|, |E(v)|)} & \text{if } u = f(u), v = f(v) \\
0 & \text{otherwise}
\end{cases}$$
GQBE: Multiple Query Tuples
Understand the connections implied by the tuples

Find answers using a lattice obtained removing edges from the union graph

GQBE finds answers for multiple query tuples
Compute a re-weighted union graph of the individual query graphs

Jayaram et al. [2015]
SEARCHING FOR

BY LOOKING AT

PRODUCES

Nodes

Connectivity
Mediator Nodes
[Gionis et al. ‘15
Ruchansky et al. ’15]
Clusters
[Perozzi et al. ’14,
Kloumann et al. ’14]

Properties
Similar Entities
[Metzger et al.’13,
Sobczak et al.’15]

Queries
Path Queries
[Bonifati et al.’15]
SPARQL
[Arenas et al.’16]

Structures
Entity Tuples
[Jayaram et al.’15]
Similar Structures
[Mottin et al.’14,
Xie et al.’17,
Lissandrini et al’18]

Few Approaches accept User Feedback

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Where we are

- Relational databases
- Textual data
- Graphs and networks

Challenges and Remarks

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
How ML fits the Big Picture

- Interface
- Example-based approaches
- Middleware
- Machine learning
- Data structures and hardware
  - Relational
  - Graph
  - Text

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Interactive exploration of datasets

**Main idea:** Learn the items to show online as more points are acquired

Two ways of learning: passive and active

RESPONDS
To USER INPUT

Learn

PASSIVE

REQUESTS
USER INPUT

Is v ✓ or x ?

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
**Main idea:** learn an implicit query from user examples and optional scores

Searching “mildly overweighted” patients

- The doctor selects examples by **browsing** patient database
- The examples have “**oblique**” correlation
- We can “**guess**” the implied query
Learning an ellipsoid distance

**Weighted distance matrix**

\[ D(x, q) = (x - q)^T M (x - q) \]

**Implicit query**

\[ D(x, q) = \sum_{j}^{n} \sum_{k}^{n} m_{jk} (x_j - q_j)(x_k - q_k) \]

Learn the query minimizing the penalty = weighted sum of distances between query point and sample vectors

\[ \text{minimize} \sum_{i} (x_i - q)^T M (x_i - q) \]

subject to \( \det(M) = 1 \)

[Ishikawa et al., 1999]
Learning the distance

Query point is moved towards “good” examples — Rocchio formula in IR

[Ishikawa et al., 1999]

$Q_0$: query point

$Q_1$: new query point

$\bullet$: retrieved data

✓: relevance judgments

Learning can be done online!!!
Explore-by-Example: AIDE

[Dimitriadou et al., 2014, 2016]
The AIDE algorithm

1. Divide the space into d-dimensional cubes
2. Find the sample points in the cubes (medoids)
3. Train the classifier
4. Refine the training sampling from neighbors of misclassified points
5. Boundary refinement

[Dimitriadou et al., 2014, 2016]
Classification & Query Formulation

[Dimitriadou et al., 2014, 2016]

<table>
<thead>
<tr>
<th>Sample</th>
<th>Red</th>
<th>Green</th>
<th>Relevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object A</td>
<td>13.67</td>
<td>12.34</td>
<td>Yes</td>
</tr>
<tr>
<td>Object B</td>
<td>15.32</td>
<td>14.50</td>
<td>No</td>
</tr>
<tr>
<td>..</td>
<td>..</td>
<td>..</td>
<td>..</td>
</tr>
<tr>
<td>Object X</td>
<td>14.21</td>
<td>13.57</td>
<td>Yes</td>
</tr>
</tbody>
</table>

```
SELECT * FROM galaxy WHERE red <= 14.82 AND red >= 13.5 AND green <= 13.74
```

Decision Tree Classifier

SELECT * FROM galaxy WHERE red <= 14.82 AND red >= 13.5 AND green <= 13.74
Misclassified Sample Exploitation

[Dimitriadou et al., 2014, 2016]
Clustering-based Sampling

Idea: Use a k-medoid approach to find sampling areas

Iterative approach: How many samples does it take to reach the desired result?

[Dimitriadou et al., 2014, 2016]
Active learning for online query systems [Vanchinathan et al., 2015]

Main idea: the system “queries” the user to understand their preferences

Learn unknown preferences and minimize the number of questions to the user
Learning unknown preferences

Problem: Find a set $S$ that maximize the unknown user preference within a budget (e.g., number of interactions)

$$\text{arg max} \sum_{v \in S} \text{pref}(v)$$

subject to $\text{Cost}(S) \leq \text{budget}$

User preferences

Cost for the set $S$
A step back ... Learning from an unknown environment ...
Multi-armed bandits

- Maximize the **reward** by successively playing gamble machines (the ‘arms’ of the bandits)
- Invented in **early 1950s** by Robbins for decision making under uncertainty when the environment is unknown
- The reward is unknown ahead of time
Multi-armed bandits

- Reward = random variable $X_{i,n} ; 1 \leq i \leq K, n \geq 1$
- $i =$ index of the gambling machine
- $n =$ number of plays
- $\mu_i =$ expected reward of machine $i$.

A policy, or allocation strategy $A$ is an algorithm that chooses the next machine to play based on the sequence of past plays and obtained rewards.
Exploration vs Exploitation


M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Greedy: A pure exploitation algorithm

Choose the machine with current best expected reward

- **Exploitation vs exploration dilemma**: Should you exploit the information you’ve learned or explore new options in the hope of greater payoff?

- In the greedy case, the balance is completely towards exploitation
- Yet, **only exploitation will not lead to a good solution**
Quality measure - Regret

Total expected regret (after T plays):

\[ R_T = \mu^* \cdot T - \sum_{i=1}^{K} \mu_j \cdot \mathbb{E}[N_{i,T}] \]

\( \mu^* \): highest expected reward

\( \mathbb{E}[N_{i,T}] \): expected number of times machine \( i \) is played
An optimistic view
Upper confidence bound (UCB) algorithm

Optimistic estimate of the mean of arm = ‘largest value it could plausibly be’

1. Pull at each time $t$ the arm with the maximum probability of being the best

   $$\frac{1}{n_j} \sum_{s=1}^{n_j} X_{j,s} + \sqrt{\frac{2 \log(1/t)}{n_j}}$$

2. Repeat until the budget (number of steps $T$) is depleted

$n_j$: number of times the arm $j$ has been pulled

Balance exploration and exploitation: The uncertainty diminishes as the time passes
Back to our problem
Modeling the same problem as a Multi-Armed Bandit

Red wavelength

Sampling Areas as Arms

Green Wavelength

Discrete Search Space Vs Continuous Space?

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Background: Gaussian processes

Idea: Model the user preferences as a Gaussian Process

A Gaussian Process (GP) is an infinite set of variables, any subset of this is Gaussian

$$P(f|\Sigma, \mu) = |2\pi\Sigma|^\frac{1}{2} \exp(-\frac{1}{2}(f - \mu)^T\Sigma^{-1}(f - \mu))$$

Gaussian prior

Specified only by mean and covariance

Given observations \(\{x, y\}_{i=1}^n\) over an unknown function \(f\) drawn from a Gaussian prior, the posterior is Gaussian

$$P(f|y) \propto \int dx \ P(f, x, y)$$
GP-Select

Algorithm 1 GP-Select

Input: Ground Set \( V \), kernel \( \kappa \) and budget \( B \)
Initialize selection set \( S \)
for \( t = 1, 2, \ldots, B \) do
  Model Update:
  \[ [\mu_{t-1}(), \sigma^2_{t-1}()] \leftarrow \text{GP-Inference}(\kappa,(S,y_{1:t-1})) \]
  Item Selection:
  Set \( v_t \leftarrow \operatorname{argmax}_{v \in V \setminus \{v_1:t-1\}} \mu_{t-1}(v) + \beta_t^{1/2} \sigma_{t-1}(v) \)
  \( S \leftarrow S \cup \{v_t\} \)
  Receive feedback \( y_t = f(v_t) + \epsilon_t \)
end for

Learn posterior

Trades off exploration exploitation

Ask user feedback

- Exploration: select items with high-variance
- Exploitation: select items with high-value

[Vanchinathan et al., 2015]
Active learning on graphs – which prior?

Idea: Use the graph structure to infer the node classes

Use graph Laplacian as prior

\[ L = D - A, \ A \text{ is the adjacency matrix} \]

\[ p(f) \sim \mathcal{N}(0, L^{-1}) \]

Laplacian: higher probability of having the same class if two nodes are connected

[Ma et al., 2015]
Where could Active learning help?

Reverse engineering queries and rules
- Interactive Refinement of example tuples
- Learning the most probable queries from their results

Graph exploration
- Summarization of knowledge graphs with preferences
- Seed set expansion
- Recommendation of relevant nodes

Text processing
- Fast entity matching
- Advertising based on documents search
Example-based methods

Similarity relation ∼

Implicit (Unknown)
- Query Reverse Engineering
- Rule Discovery
- Relation Extraction

Explicit (Known)
- Structural Similarity
- Proximity Search
- Document Matching

NEW GOAL: Learn ∼ Interactively!
MAB: good resources

Books and surveys

Tutorials
- Lattimore - AAAI 2018: part 1 - part 2
- Tutorial on bayesian optimization of expensive cost functions
- Blog on bandits: http://banditalgs.com/
Where we are

- Relational databases
- Textual data
- Graphs and networks

Challenges and Remarks
Big data – Easy value?

Let’s solve this problem by using the big data. None of us have the slightest idea what to do with.

Answers

Simple but wrong

Complex but right

This way

@marketoonist.com
Exploration

We know where we start
we don’t know what we’ll find
Traditional Search Methods are not Enough
We need Specialized Methods for Data Exploration

From broad views
From exploration as
select count(*)

To Detailed view
To find what is interesting

From Exact Search
Based on explicit conditions

To Exploratory Search
Based on Implicit needs
Similarities are the key …

Is there a galaxy like this?

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Example-based methods: All You Need is …
Example-based methods

**Relational**
- Reverse engineering queries
- Example-driven schema mapping
- Interactive data repairing

**Textual**
- Search documents by example
- Entity extraction by example text
- Web table completion using examples

**Graph**
- Community-based Node-retrieval
- Entity Search
- Path and SPARQL queries
- Graph structures as Examples
### Example-based methods: takeaways

**Relational**
- Complex search space
- Exact and approximate
- Interactivity can improve the quality
- Limited to query inference

**Textual**
- Allows serendipitous search
- Easier document finding
- Speed up entity matching
- Extract semi-structure data

**Graph**
- Heterogenous Structures
- Exploit locality
- Entity attributes are expressive
- Large result-sets require ranking

---

*Example-based methods: takeaways*

- M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
The use of examples

Examples can ease data exploration
• … reduce need for complex queries / simplify user input
• … require no schema knowledge
• … allow uncertainty in search conditions
• … require little data analytics expertise
Acknowledgments

We would like to thank the authors of the papers who kindly provided us the slides

Angela Bonifati, Radu Ciucianu, Marcelo Arenas, Gonzalo Diaz, Egor Kostylev, Yaacov Weiss, Sarah Cohen, Fotis Psallidas, Li Hao, Chan Chee Yong, Ilaria Bordino, Mohamed Yakout, Kris Ganjam, Kaushik Chakrabati, Thibault Sellam, Rohit Singh, Maeda Hanafi, Dmitri Kalashnikov, Marcin Sydow, Mingzhu Zhu, Yoshiharu Ishikawa, Daniel Deutch, Nandish Jayaram, Paolo Papotti, Bryan Perozzi, Kiriaki Dimitriadou, Yifei Ma, Natali Ruchansky, Quoc Trung Tran, Hastagiri Prakash Vanchinathan

... and many others (see references)
Where should we invest time?

- Machine learning
- Approximate Methods
- User models
- Scalability
Where should we invest time?

Learn from Examples

- … Similarity Measures: are often “fuzzy” and “implicit”
- … New representations of the search space
- Challenge: Scale! Exploration of large search spaces
Where should we invest time?

Learn from Examples

• … Similarity Measures to represent User Interests
• … User-centric, dynamic, Exploration-strategies: learn as you go
• Challenge: Distinct User have Different Goals! Explore in different ways

We need more data!
Where should we invest time?

Scale Example-based search

- ... Huge search space, dynamic data, variety of data models
- ... Exploration is Interactive, requires Interactive response time
- Adaptive Data-structures, localized access, flexible schema, incremental index
Where should we invest time?

Scale Example-based search

- ... An approximate answer now is better than a precise answer in 1 hour
- ... Approximate answers can provide insights without being accurate

Exploratory queries retrieve large resultsets: the user needs only a glimpse to figure out if they are moving in the right direction!
Features of Exploratory Search Systems [White and Roth, 2009]

Support querying and rapid query refinement:
- Offer facets and metadata-based result filtering
- Leverage search context

Example-driven
- visualizations, summarizations, and explanations
- paired with methods to suggest further example-based explorations.

Support learning and understanding
Interactive Example Based Exploration System?

Requires:

**Fast Query Processing**
Avoid the full recomputation of a query
Limit the computation to only a sample
Adaptive query executions
Adaptive data-structures and indexes,

**Automatic Result Analysis**
Automatically identify peculiar characteristics,
Data-summarization techniques
Learn user interests automatically
ADOPT HETEROGENEITY

Need for solutions that operate across different models
operate on heterogeneous datastores
dataset search

*Data Lakes??*
DEMOCRATIZATION

easy access to data

Tools that work on commodity hardware, mobile devices

Data-exploration for everyday use-cases

Users want back the control on their data

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
NATURAL LANGUAGE INTERFACE

flexible, vague, imprecise input

Exploration through conversation

M. Lissandrini, D. Mottin, T. Palpanas, Y. Velegrakis
Example is always more efficacious than precept

Samuel Johnson, Rasselas (1759), Chapter 29.


Slides: https://data-exploration.ml/
References


A. Bonifati, R. Ciucanu, and A. Lemay. Learning path queries on graph databases. EDBT, 2015.


I. Bordino, G. De Francisci Morales, I. Weber, and F. Bonchi. From machu picchu to rafting the urubamba river: anticipating information needs via the entity-query graph. WSDM, 2013.

References


References

N. Jayaram, A. Khan, C. Li, X. Yan, and R. Elmasri. Querying knowledge graphs by example entity tuples. TKDE, 2015.


References


References


M. M. Zloof. Query by example. AFIPS NCC, 1975.