Meta-blocking:
Taking Entity Resolution to the Next Level

George Papadakis, Georgia Koutrika,
Themis Palpanas, Wolfgang Nejdl

In IEEE TKDE, August 2014.
Entities: an invaluable asset

“Entities” is what a large part of our knowledge is about:
However ...

How many names, descriptions or IDs (URIs) are used for the same real-world “entity”?
However ...

*How many names, descriptions or IDs (URIs) are used for the same real-world “entity”?*
However ...

How many names, descriptions or IDs (URIs) are used for the same real-world “entity”?

London 런던 ლანდონ लंदन ळंडन लंडन ロンドン
Londain Londe Londen Londen Londen Londinium London Londona Londonas Londoni Londra Londres Londrez Londyn Lontoo Loundres Luân Dôn Lunden Lundúnir Lunnainn Lunnnon لندن لندن لندن لندن London Londen Londres Łódź Londres Londen Londres Лондон Лондон Лондон Лондон Лондон Londinium伦敦 ...

capital of UK, host city of the IV Olympic Games, host city of the XIV Olympic Games, future host of the XXX Olympic Games, city of the Westminster Abbey, city of the London Eye, the city described by Charles Dickens in his novels, ...
However ...

How many names, descriptions or IDs (URIs) are used for the same real-world “entity”?

capital of UK, host city of the IV Olympic Games, host city of the XIV Olympic Games, future host of the XXX Olympic Games, city of the Westminster Abbey, city of the London Eye, the city described by Charles Dickens in his novels, ...

http://sws.geonames.org/2643743/
...
... or ...

How many “entities” have the same name?

- London, KY
- London, Laurel, KY
- London, OH
- London, Madison, OH
- London, AR
- London, Pope, AR
- London, TX
- London, Kimble, TX
- London, MO
- London, MO
- London, London, MI
- London, London, Monroe, MI
- London, Uninc Conecuh County, AL
- London, Uninc Conecuh County, Conecuh, AL
- London, Uninc Shelby County, IN
- London, Uninc Shelby County, Shelby, IN
- London, Deerfield, WI
- London, Deerfield, Dane, WI
- London, Uninc Freeborn County, MN
- ...

... Or ...

How many “entities” have the same name?

- London, KY
- London, Laurel, KY
- London, OH
- London, Madison, OH
- London, AR
- London, Pope, AR
- London, TX
- London, Kimble, TX
- London, MO
- London, MO
- London, London, MI
- London, London, Monroe, MI
- London, Uninc Conecuh County, AL
- London, Uninc Conecuh County, Conecuh, AL
- London, Uninc Shelby County, IN
- London, Uninc Shelby County, Shelby, IN
- London, Deerfield, WI
- London, Deerfield, Dane, WI
- London, Uninc Freeborn County, MN
- ...

- London, Jack
  2612 Almes Dr
  Montgomery, AL
  (334) 272-7005

- London, Jack R
  2511 Winchester Rd
  Montgomery, AL 36106-3327
  (334) 272-7005

- London, Jack
  1222 Whitetail Trl
  Van Buren, AR 72956-7368
  (479) 474-4136

- London, Jack
  7400 Vista Del Mar Ave
  La Jolla, CA 92037-4954
  (858) 456-1850

- ...

Content Providers

*How many content types / applications provide valuable information about each of these “entities”?*

- News about London
- Reviews on hotels in London
- Wiki pages about the London
- Pictures and tags about London
- Social networks in London
- Videos and tags for London
Preliminaries

**Entity profile**

a uniquely identified set of name-value pairs that corresponds to a single real-world object.

**Entity Resolution**

identifies and aggregates the different entity profiles/records that actually describe the same real-world object.

**Application areas:**

Social Networks, census data, price comparison portals, Linked Data

**Useful because:**

- improves data quality and integrity
- fosters re-use of existing data sources.
Computational cost

ER is an inherently quadratic problem (i.e., $O(n^2)$): every entity has to be compared with all others.

ER does not scale to large entity collections (i.e., Big Data).

Solution: Blocking

• group similar entities into blocks
• execute comparisons only inside blocks
Blocking

Metrics for assessing block quality:

• Pair Completeness: \[ PC = \frac{\text{detected\_matches}}{\text{existing\_matches}} \] (effectiveness)

• Reduction Ratio: \[ RR = 1 - \frac{\text{method\_comparisons}}{\text{baseline\_comparisons}} \] (efficiency)

Definition of Blocking:
Given an entity collection, cluster its entities into blocks and process them so that both PC and RR are maximized.

disclaimer:
Two matching profiles are detected as long as they co-occur in at least one block. Thus, the precision of entity matching is dependent on the entity similarity measures and is orthogonal to the above problem.
Blocking in Databases

Entity 1
- first name=Antony P.
- last name=Gray
- address=Los Angeles, California
- zip_code=91456

Entity 2
- first name=Bill
- last name=Green
- address=Los Angeles, California
- zip_code=94520

Entity 3
- first name=Antony
- last name=Gray
- address=L.A., California, USA
- zip_code=91456

Entity 4
- first name=William Nicholas
- last name=Green
- address=L.A., California, USA
- zip_code=94520

Blocks on zip_code:
- 91456: Entity 2, Entity 4
- 94520: Entity 1, Entity 3
Characteristics of Big Data

- They include Web 2.0 and Semantic Web data.
- Voluminous, (semi-)structured datasets.
  - DBPedia 3.4: 36.5 million triples and 2.1 million entities
  - BTC09: 1.15 billion triples, 182 million entities.

- Users are free to insert not only attribute values but also attribute names → high levels of heterogeneity.
  - DBPedia 3.4: 50,000 attribute names
  - Google Base:100,000 schemata for 10,000 entity types
  - BTC09: 136K attribute names

- Large portion of data originating from automatic information extraction techniques → noise, tag-style values.
Example of Big Data

DATASET 1

Entity 1
- name: United Nations Children’s Fund
- acronym: unicef
- headquarters: California
- address: Los Angeles, 91335

Entity 2
- name: Ann Veneman
- position: unicef
- address: California
- ZipCode: 90210

DATASET 2

Entity 3
- organization: unicef
- California
- status: active
- Los Angeles, 91335

Entity 4
- firstName: Ann
- lastName: Veneman
- residence: California
- zip_code: 90201

Loose Schema Binding

Split values

Attribute Heterogeneity

Noise
Token Blocking Example

DATASET 1

Entity 1
- name=United Nations Children’s Fund
- acronym=unicef
- headquarters=California

Entity 2
- name=Ann Veneman
- position=unicef
- address=California

DATASET 2

Entity 3
- organization=unicef
- hdq=California
- status=active

Entity 4
- firstName=Ann
- lastName=Veneman
- residence=California
Type of pair-wise comparisons

Every *comparison* between entity profiles $p_i$ and $p_j$ belongs to one of the following types:

1. **Matching** if $p_i \equiv p_j$.
2. **Redundant** if $p_i$ and $p_j$ co-occur and are compared in another block.
3. **Superfluous** if $p_i \neq p_j$ and the comparison is not redundant.
Meta-blocking

Goal:
restructure a given block collection into a new one that contains substantially lower number of redundant and superfluous comparisons ($RR \gg 0$), while maintaining the original number of matching ones ($\Delta PC \approx 0$).
Taxonomy of Blocking Methods

• **Redundancy-free:**
  disjoint blocks (e.g., Standard Blocking)

• **Redundancy-bearing:**
  overlapping blocks
  – **Redundancy-neutral** (e.g., Sorted Neighborhood)
    all entities share the same number of blocks
  – **Redundancy-negative** (e.g., Canopy Clustering)
    the most similar entities share few blocks (perhaps just one)
  – **Redundancy-positive** (e.g., Token Blocking)
    the more blocks two entities share, the more similar and the more likely they are to be matching
Outline of Meta-blocking

B → Graph Building → G_B → Edge Weighting → G_B^w → Graph Pruning → G_B^p → Block Collecting → B'
Graph Building

for every block
  for every entity → add a node
    for every pair of entities → add an undirected edge

Blocking graph:
• It eliminates all redundant comparisons.
• Low materialization cost → implicit materialization through inverted indices or bit arrays.
Edge Weighting

We propose five attribute-agnostic weighting schemes that rely on the following evidence:

• the number of blocks shared by two entities
• the size of the common blocks
• the number of blocks or comparisons involving each entity.

Computational Cost:

• In theory, equal to executing all pair-wise comparisons in the given block collection.
• In practice, significantly lower because it does not employ string similarity metrics.
Weighting Schemes

We propose five attribute-agnostic weighting Schemes:

1. Aggregate Reciprocal Comparisons Scheme (ARCS)
   \[ e_{i,j}.\text{weight} = \sum_{b_k \in \mathcal{B}_{i,j}} \frac{1}{||b_k||} \]

2. Common Blocks Scheme (CBS)
   \[ e_{i,j}.\text{weight} = |\mathcal{B}_{i,j}| \]

3. Enhanced Common Blocks Scheme (ECBS)
   \[ e_{i,j}.\text{weight} = |\mathcal{B}_{i,j}| \cdot \log \frac{|\mathcal{B}|}{|\mathcal{B}_i|} \cdot \log \frac{|\mathcal{B}|}{|\mathcal{B}_j|} \]

4. Jaccard Scheme (JS)
   \[ e_{i,j}.\text{weight} = \frac{|\mathcal{B}_{i,j}|}{|\mathcal{B}_i| + |\mathcal{B}_j| - |\mathcal{B}_{i,j}|} \]

5. Enhanced Jaccard Scheme (EJS)
   \[ e_{i,j}.\text{weight} = \frac{|\mathcal{B}_{i,j}|}{|\mathcal{B}_i| + |\mathcal{B}_j| - |\mathcal{B}_{i,j}|} \cdot \log \frac{|E_B|}{|v_i|} \cdot \log \frac{|E_B|}{|v_j|} \]
Graph Pruning – Part A

Pruning algorithms
1. Edge-centric
2. Node-centric
   they produce directed blocking graphs

Pruning criteria
• Scope:
  1. Global
  2. Local
• Functionality:
  1. Weight thresholds
  2. Cardinality thresholds
Graph Pruning – Part B

Every family of pruning algorithms requires setting a threshold. Experimentally verified robust behavior of the following configurations:

1. **Weight Edge Pruning (WEP)**
   average weight across all edges

2. **Cardinality Edge Pruning (CEP)**
   \[ K = BC^* \cdot |E| / 2 \]

3. **Weight Node Pruning (WNP)**
   for each node, the average weight of the adjacent edges

4. **Cardinality Node Pruning (CNP)**
   for each node, \( k=BC-1 \)

* **Blocking Cardinality (BC):** average blocks per entity
Block Collecting

Transform the pruned blocking graph into a new block collection.

For **undirected** blocking graphs:
- every retained edge creates a block of minimum size

For **directed** blocking graphs:
- for every node (with retained *outgoing* edges), we create a new block containing the corresponding entities
Experimental Settings

• Metrics
  – Pair Completeness: \[ PC = \frac{\text{detected_matches}}{\text{existing_matches}} \]
  – Reduction Ratio: \[ RR = 1 - \frac{\text{method_comparisons}}{\text{baseline_comparisons}} \]

• Datasets

<table>
<thead>
<tr>
<th></th>
<th>Dbpedia (Clean-Clean ER)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>30RC</td>
</tr>
<tr>
<td>Entities</td>
<td>1,190,734</td>
</tr>
<tr>
<td>Name-Value Pairs</td>
<td>17,453,516</td>
</tr>
<tr>
<td>Duplicates</td>
<td></td>
</tr>
</tbody>
</table>

Note: out of 43,75 million distinct triples of \( D_{\text{DBPedia}} \), only 10,36 million (<25%) are common.
## Meta-blocking Performance

<table>
<thead>
<tr>
<th>WEP</th>
<th>Comparisons</th>
<th>RR</th>
<th>PC</th>
<th>ΔPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>TB+BP</td>
<td>$3.98 \cdot 10^{10}$</td>
<td>-</td>
<td>99.91%</td>
<td>-</td>
</tr>
<tr>
<td>ARCS</td>
<td>$2.85 \cdot 10^{8}$</td>
<td>99.28%</td>
<td>92.45%</td>
<td>-7.45%</td>
</tr>
<tr>
<td>CBS</td>
<td>$3.40 \cdot 10^{9}$</td>
<td>91.46%</td>
<td>95.47%</td>
<td>-4.42%</td>
</tr>
<tr>
<td>ECBS</td>
<td>$5.77 \cdot 10^{9}$</td>
<td>85.50%</td>
<td>99.66%</td>
<td>-0.23%</td>
</tr>
<tr>
<td>JS</td>
<td>$1.11 \cdot 10^{10}$</td>
<td>71.80%</td>
<td>99.73%</td>
<td>-0.16%</td>
</tr>
<tr>
<td>EJS</td>
<td>$1.10 \cdot 10^{10}$</td>
<td>72.32%</td>
<td>99.77%</td>
<td>-0.11%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>WNP</th>
<th>Comparisons</th>
<th>RR</th>
<th>PC</th>
<th>ΔPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCS</td>
<td>$1.85 \cdot 10^{9}$</td>
<td>95.34%</td>
<td>99.41%</td>
<td>-0.48%</td>
</tr>
<tr>
<td>CBS</td>
<td>$3.57 \cdot 10^{9}$</td>
<td>91.04%</td>
<td>99.35%</td>
<td>-0.54%</td>
</tr>
<tr>
<td>ECBS</td>
<td>$9.94 \cdot 10^{9}$</td>
<td>75.02%</td>
<td>99.75%</td>
<td>-0.14%</td>
</tr>
<tr>
<td>JS</td>
<td>$1.96 \cdot 10^{10}$</td>
<td>50.76%</td>
<td>99.87%</td>
<td>-0.02%</td>
</tr>
<tr>
<td>EJS</td>
<td>$1.99 \cdot 10^{10}$</td>
<td>49.74%</td>
<td>99.88%</td>
<td>-0.01%</td>
</tr>
</tbody>
</table>

### Comp. | RR | ARCS | CBS | ECBS | JS | EJS
---|-----|------|-----|------|----|-----
CEP | $0.26 \cdot 10^{8}$ | 99.94% | 79.46% | 51.71% | 61.14% | 82.09% | 79.61%
CNP | $0.50 \cdot 10^{8}$ | 99.88% | 93.43% | 92.35% | 94.05% | 95.57% | 95.99%
Meta-blocking Time Requirements

Materialization Time ($MT$): time for Graph Building and Edge Weighting.

Restructure Time ($RT$): time for Graph Pruning and Block Collecting.

Comparisons Time ($CT$): time for executing the retained comparisons.

Performance over DBPedia in hours, using Intel Xeon E5472 3.0 GHz and 16GB of RAM. Profile comparison was done with Jaccard similarity.
Conclusions

Contributions:
1. We formalized the problem of meta-blocking.
2. We presented a taxonomy of solutions based on the blocking graph.
3. We coined five schema-agnostic weighting schemes.
4. We introduced two categories of pruning algorithms along with two orthogonal categories of pruning criteria.

Several challenges ahead:
   – Parallelization according to the MapReduce paradigm.
   – Incremental methods are necessary.
Thank You!