Discovering Spatial and Temporal Links among RDF Graphs

Publishing and Interlinking Linked Geospatial Data
In Conjunction with the 12th Extended Semantic Web Conference

Portoroz, Slovenia, 1st June 2015

Presenter: Panayiotis Smeros
Outline

• Introduction to Entity Resolution and Link Discovery
  – Examples, Definitions, Common Problems

• Spatial Entity Resolution

• Spatial and Temporal Link Discovery
  – Background and Developed Methods
  – Extensions to the Silk Framework
  – Hands-on
Entities in Real-World

Most of our knowledge about the world is based on entities and their relations:
Entities in Data-World

Many names, descriptions or IDs (URIs) are used for the same real-world entity:

Portoroz Portorož پورتوروز Портороž Порторож پورتوروز Portorose Портороз Порторожу Portorožu Порторож

Portorož (Italian: Portorose, literally "Port of Roses"), is an Adriatic - Mediterranean coastal settlement in the Municipality of Piran in southwestern Slovenia. Its modern development began in the late 19th century with appearance of first health resorts.

http://www.geonames.org/3192682/portoroz.html
http://en.wikipedia.org/wiki/Portoroz
http://www.portoroz.si/en/
…
Many applications provide valuable information about each of these entities:

- Wiki pages about Portoroz
- News about Portoroz
- Reviews of hotels in Portoroz
- Pictures about Portoroz
- Social networks in Portoroz
- Videos for Portoroz
Content Providers

Many applications provide valuable information about each of these entities:

- Wiki pages about Portoroz
- News about Portoroz
- Reviews of hotels in Portoroz
- Social networks in Portoroz
- Pictures about Portoroz
- Videos for Portoroz

Solution?
Entity Resolution

Problem of understanding that two (or more) entities in data-world are references of the same real-world entity. [Christen, TKDE’11]
Entity Resolution (Example)

Problem of understanding that two (or more) entities in data-world are references of the same real-world entity. [Christen, TKDE’11]

DBpedia

Entity

name = PORTOROZ
population = 2,849

GeoNames

Entity

name = Portorose
population = 2,851
Entity Resolution (Example)

Problem of understanding that two (or more) entities in data-world are references of the same real-world entity. [Christen, TKDE’11]

Entity

name = PORTOROZ
population = 2,849

sameAs

Entity

name = Portorose
population = 2,851
Spatial Entity Resolution (Example)

Problem of understanding that two (or more) entities in **data-world** are references of the same **real-world** entity. [Christen, TKDE’11]

![Diagram showing Spatial Entity Resolution](image)

- **DBpedia**
  - Entity
    - name = PORTOROZ
    - population = 2,849
    - location = 45.51663, 13.57996

- **GeoNames**
  - Entity
    - name = Portorose
    - population = 2,851
    - location = 45.51661, 13.57998

sameAs
Entity Resolution (Definition)

Let $S$ and $T$ be two sets of entities. We define a distance (similarity) function $d_{\text{similarity}}$ and a distance (similarity) threshold $\theta_{d_{\text{similarity}}}$ as follows:

$$d_{\text{similarity}} : S \times T \rightarrow [0,1] , \quad \theta_{d_{\text{similarity}}} \in [0,1]$$

We define the set of discovered similarity links $DL_{\text{similarity}}$ as follows:

$$DL_{\text{similarity}} = \{(s, \text{sameAs}, t) \mid s \in S \land t \in T \land d_{\text{similarity}}(s, t) < \theta_{d_{\text{similarity}}})\}$$
Link Discovery is the fourth and the most important Linked Data Principle.

Establish **semantic relations** between entities in order to enrich the information that is known about them. [Bizer et al., IJSWIS’06]
Let $S$ and $T$ be two sets of entities and $R$ the set of relations that can be discovered between entities. For a relation $r \in R$, w.l.o.g., we define a distance function $d_r$ and a distance threshold $\theta_{d_r}$ as follows:

\[
d_r : S \times T \rightarrow [0, 1] , \quad \theta_{d_r} \in [0, 1]
\]

We define the set of discovered links for relation $r$ ($DL_r$) as follows:

\[
DL_r = \{(s, r, t) \mid s \in S \land t \in T \land d_r(s, t) < \theta_{d_r}\}
\]
Link Discovery (Example)

Natura (2000) - Fields

Fields - OSM Water Bodies
Link Discovery (Example)

Natura (2000) - Fields

contains

intersects

Fields - OSM Water Bodies
Main Problem: Heterogeneity

- Different Data Providers create **Heterogeneous Datasets**
  - Example: Literal Heterogeneity (case, language, etc).

- We focus on:
  - Heterogeneity in the Representation of Geospatial Information in RDF
  - Heterogeneity in the Representation of Temporal Information in RDF
Heterogeneity in the Representation of Geospatial Information in RDF

_:1 rdf:type geo:Geometry .
_:1 geo:hasGeometry "<http://www.opengis.net/def/crs/EPSG/0/4326> POINT(10 20)"^^geo:wktLiteral .

_:1 rdf:type strdf:Geometry .

_:1 rdf:type wgs84Geo:Point .
_:1 wgs84Geo:lat "10"^^xsd:double .
_:1 wgs84Geo:long "20"^^xsd:double .
Heterogeneity in the Representation of Geospatial Information in RDF

_:1 rdf:type geo:Geometry .
_:1 geo:hasGeometry

_:1 rdf:type strdf:Geometry .
_:1 strdf:hasGeometry

- **Different Vocabularies**
Heterogeneity in the Representation of Geospatial Information in RDF

`:1 rdf:type geo:Geometry .
`:1 geo:hasGeometry

`:1 rdf:type strdf:Geometry .
`:1 strdf:hasGeometry

- **Different Vocabularies**
- **Different Serializations of Geometries**
Heterogeneity in the Representation of Geospatial Information in RDF

```sparql
_:1 rdf:type geo:Geometry .
_:1 geo:hasGeometry
"<http://www.opengis.net/def/crs/EPSG/0/4326>
POINT(10 20)"^^geo:wktLiteral .

_:1 rdf:type strdf:Geometry .
_:1 strdf:hasGeometry
"<gml:Point crsName="EPSG:2100">\n\t<gml:coordinates>10,20
\t</gml:coordinates></gml:Point>"^^strdf:GML .
```

- **Different Vocabularies**
- **Different Serializations of Geometries**
- **Geometries expressed in Different Coordinate Reference Systems (CRS)**
Discovering Spatial and Temporal Links among RDF Graphs
Heterogeneity in the Representation of Geospatial Information in RDF

- Different Sampling Values
- Different Granularity
- Different Rounding Effects
Heterogeneity in the Representation of Temporal Information in RDF

_:1 ex:hasBirthday "1989-09-24T11:05:00+01:00"xsd:dateTime .

_:1 ex:hasAffiliation ex:UoA
"[2007-10-15T00:00:00+03:00, 2013-10-15T00:00:00+04:00)"^^strdf:Period .
Different Vocabularies
Heterogeneity in the Representation of Temporal Information in RDF

_:1 ex:hasBirthday "1989-09-24T11:05:00+01:00" xsd:dateTime .

_:1 ex:hasAffiliation ex:UoA "[2007-10-15T00:00:00+03:00,
                        2013-10-15T00:00:00+04:00]"^^strdf:Period .

- **Different Vocabularies**
- **Different Time Zones**
Heterogeneity in the Representation of Temporal Information in RDF

_:1 ex:hasBirthday "1989-09-24T11:05:00+01:00" xsd:dateTime .

_:1 ex:hasAffiliation ex:UoA "[2007-10-15T00:00:00+03:00, 2013-10-15T00:00:00+04:00)"^^strdf:Period

- **Different Vocabularies**
- **Different Time Zones**
- **Time Instants and Periods**
Outline

• Introduction to Entity Resolution and Link Discovery
  – Examples, Definitions, Common Problems

• Spatial Entity Resolution

• Spatial and Temporal Link Discovery
  – Background and Developed Methods
  – Extensions to the Silk Framework
  – Hands-on
Problem of understanding that two (or more) entities in data-world are references of the same real-world entity. [Christen, TKDE’11]
Spatial Entity Resolution (1/4)

• Location Name Similarity
  – Edit, Jaccard distance

• Location Similarity
  – Euclidean distance

• Location Type Similarity
  – (e.g. type “river” is similar to type “stream”)

Combines the above similarities to compute the overall similarity between entities
Spatial Entity Resolution (2/4)

• Similarity measure: Hausdorff Distance
  – Intuitively, Hausdorff Distance is defined as the largest distance between the closest points of two geometric shapes

\[ d_H(A, B) = \max \left( \left\{ \arg \max_{a \in A} \arg \min_{b \in B} d(a, b), \arg \max_{b \in B} \arg \min_{a \in A} d(a, b) \right\} \right) \]

• Handling Geospatial Heterogeneity
  – Converts geometries to a common vocabulary (NeoGeo)
  – Assumes WGS-84 CRS

• Optimization
  – Simplifies Geometries with Ramer-Douglas-Peucker algorithm
Spatial Entity Resolution (3/4)

- Heuristic Combination of:
  - URI Similarity
  - Label Similarity
    - Considering the language of the labels
  - Location Similarity
    - Assuming the W3C Geo vocabulary
  - Geometric Similarity
    - Minimum Distance between two Geometries
Spatial Entity Resolution (4/4)

• Non-Spatial Criteria
  – Implemented within the LIMES framework

• Geometric Similarity
  – Hausdorff Distance
  – Optimizations
    • Bounding Circle: Avoids useless comparisons
      \[ \mu(s, t) = \delta(\zeta(s), \zeta(t)) - r(s) - r(t) > \theta \Rightarrow \delta(s, t) > \theta \]
    • Space tiling: Reduces the quadratic number of comparisons

[Ngonga Ngomo, ISWC'13]
Spatial Entity Resolution

- [Sehgal et al. GIS’06]
  - Spatial and non-Spatial Criteria
  - Only Location Similarity

- [Salas et al., TerraCognita’11]
  - Only Spatial Criteria
  - Complex Geometric Similarity Methods

- [Vilches-Blázquez et al., AGILE’12]
  - Spatial and non-Spatial Criteria
  - Simple Geometric Similarity Methods

- [Ngonga Ngomo, ISWC’13]
  - Spatial and non-Spatial Criteria
  - Complex Geometric Similarity Methods
  - Reduced number of comparisons
Outline

• Introduction to Entity Resolution and Link Discovery
  – Examples, Definitions, Common Problems

• Spatial Entity Resolution

• Spatial and Temporal Link Discovery
  – Background and Developed Methods
  – Extensions to the Silk Framework
  – Hands-on
Link Discovery is the fourth and the most important **Linked Data Principle**.

Establish **semantic relations** between entities in order to **enrich the information** that is known about them. [Bizer et al., IJSWIS’06]
• Dimensionally Extended 9-Intersection Model
  [Clementini et al., SSD’93]
  – Captures topological relations in $\mathbb{R}^2$, by considering the dimension (dim) of the intersections involving the interior (I), the boundary (B) and the exterior (E) of the two geometries.

$$DE-9IM(a,b) = \begin{bmatrix}
  \text{dim}(I(a) \cap I(b)) & \text{dim}(I(a) \cap B(b)) & \text{dim}(I(a) \cap E(b)) \\
  \text{dim}(B(a) \cap I(b)) & \text{dim}(B(a) \cap B(b)) & \text{dim}(B(a) \cap E(b)) \\
  \text{dim}(E(a) \cap I(b)) & \text{dim}(E(a) \cap B(b)) & \text{dim}(E(a) \cap E(b))
\end{bmatrix}$$

– Examples: Intersects, Equals, Touches, Disjoint, Contains, Crosses, Covers, CoveredBy and Within
• Region Connection Calculus [Randell et al. KR’92]
  – **RCC-8**: a well-known subset of RCC, which is based on eight topological relations

  ![Diagram of eight RCC-8 relations](image)

  - DC stands for DisConnected, EC for Externally Connected, TPP for Tangential Proper Part, NTPP, for Non Tangential Proper Part, and TPPi and NTPPi are the inverse relations of TPP and NTPP
Background on Temporal Relations

- Allen’s Interval Calculus [Allen, Commun. ACM’83]
  - thirteen jointly exclusive and pairwise disjoint qualitative relations

<table>
<thead>
<tr>
<th>Relation</th>
<th>Illustration</th>
</tr>
</thead>
<tbody>
<tr>
<td>X before Y</td>
<td>X ______</td>
</tr>
<tr>
<td>Y after X</td>
<td>______ Y ______</td>
</tr>
<tr>
<td>X meets Y</td>
<td>X _____</td>
</tr>
<tr>
<td>Y isMetBy X</td>
<td>Y _____</td>
</tr>
<tr>
<td>X overlaps Y</td>
<td>X _____</td>
</tr>
<tr>
<td>Y isOverlappedBy X</td>
<td>Y _____</td>
</tr>
<tr>
<td>X starts Y</td>
<td>X _____</td>
</tr>
<tr>
<td>Y isStartedBy X</td>
<td>Y __________</td>
</tr>
<tr>
<td>X during Y</td>
<td>X ___</td>
</tr>
<tr>
<td>Y contains X</td>
<td>Y __________</td>
</tr>
<tr>
<td>X finishes Y</td>
<td>X _____</td>
</tr>
<tr>
<td>Y isFinishedBy X</td>
<td>Y __________</td>
</tr>
<tr>
<td>X equals Y</td>
<td>X _____</td>
</tr>
<tr>
<td></td>
<td>Y _____</td>
</tr>
</tbody>
</table>
Spatial and Temporal Relations

• We consider the previous Spatial ($R_s$) and Temporal ($R_t$) relations as Boolean relations ($R_B$) i.e., either they hold or they do not:

$$R_s, R_t \subset R_B$$

• $R_B$ constitutes a special subset of $R$. The distance function $d_r$ and the distance threshold $\theta_{d_r}$ for a relation $r \in R_B$ are defined as follows:

$$d_r(s,t) = \begin{cases} 0 & \text{if } r \text{ holds} \\ 1 & \text{elsewhere} \end{cases}, \quad \theta_{d_r} = 1$$
Spatial and Temporal Transformations (1/2)

- **CRS Transformation.** The geometries of a dataset can be expressed in a Coordinate Reference System that is more precise for the geographic area that they describe (e.g., the GGRS87 for Greece). This transformation converts the CRS of a geometry to the World Geodetic System (WGS 84).

- **Vocabulary Transformation.** This transformation converts geometry literals from GeoSPARQL, stRDF or W3C GEO to a common vocabulary (GeoSPARQL).

- **Serialization Transformation.** This transformation converts the geometries of a dataset to a common serialization (WKT).

- **Time-Zone Transformation.** This transformation converts the time zone of a given time interval to Coordinated Universal Time (UTC).

- **Period Transformation.** This transformation converts a time instant to a period with the same starting and ending point.
Spatial and Temporal Transformations

(2/2)

- **Simplification Transformation.** Some datasets have very complex geometries, which makes the computation of spatial relations inefficient. This transformation simplifies a geometry according to a given distance tolerance, ensuring that the result is a valid geometry having the same dimension and number of components as the input.

- **Envelope Transformation.** This transformation computes the envelope (i.e., the minimum bounding rectangle) of a geometry and it is useful in cases that we want to compute approximate spatial relations between two datasets.

- **Area Transformation.** In some cases it is enough to compare just the areas of two geometries to infer whether they are the same or not. This transformation computes the area of a given geometry in square metres.

- **Points-To-Centroid Transformation.** In crowdsourcing datasets like OpenStreetMap, multiple users can define the position of the same placemark. As a better approximation of the real position of this placemark we can compute the centroid of these positions. This transformation computes the centroid of a cluster of points.
Techniques for Checking the Relations

• **Cartesian Product** Technique (Naive)
  – Performs exhaustive checks between the pairs of the entities of datasets
  – Complete
  – Complexity: $O(|S||T|)$ checks

• **Blocking** Technique [Isele et al., WebDB’11, Papadakis et al, TKDE’13]
  – Divides the entities into blocks
  – Decreases the number of checks
  – Complete
  – Complexity: $O(|S||T|)$ checks (worst case), $O(|L|)$ checks (best case)

* $|S|$, $|T|$: number of entities in datasets $S$ and $T$; $|L|$: number of links between datasets $S$ and $T
Blocking Technique for Spatial Relations

- Divide the surface of the earth into *curved rectangles* (blocks)

- Adjust the area of the blocks with a *blocking factor* (bf)

\[
\text{blockArea: } \frac{1}{bf^2}
\]

- If the MBB of a geometry spatially intersects with a block, then insert it in this block

- Check for a spatial relation only within each block (independently)

- Construct the set of discovered links \((DL_r)\) by *aggregating* the respective links that have been discovered within each block
Blocking Technique for Temporal Relations

- Divide the time into **intervals** (blocks)

- Adjust the length of the blocks with a **blocking factor** (bf) (blockLength: $\frac{1}{bf}$ *time units*)

- If a time period or instant temporally intersects with a block, then insert it in this block

- Check for a temporal relation **only within** each block (independently)

- Construct the set of discovered links ($DL_r$) by **aggregating** the respective links that have been discovered within each block
Blocking Technique

- Fully parallelizable with respect to the blocks
- Proven sound and complete
- 100% accurate links
- 100% precision, recall, F-measure
Extensions to the Silk Framework: Spatial and Temporal Relations

Data Source → Blocking → Link Generation → Filtering → Output

Silk

Discovering Spatial and Temporal Links among RDF Graphs
Extensions to the Silk Framework: Spatial and Temporal Transformations

Data Source → Blocking → Link Generation → Filtering → Output

Buffer, Serialization, Area, Geometry, Literal, Points-To-Centroid, Envelope, Simplification, TimeZone, CRS
Extensions to the Silk Framework

• Spatial and Temporal Extensions for Silk implemented as Plugins

• Transparent to all the applications of Silk
  – Single Machine
  – MapReduce
  – Workbench
Hands-on Silk

• Download: [https://github.com/silk-framework/silk](https://github.com/silk-framework/silk)
• Workbench application pre-installed in the VM

• Discover the following links:

<table>
<thead>
<tr>
<th>Source Dataset</th>
<th>Relation</th>
<th>Target Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Boundaries</td>
<td>Contains</td>
<td>Raster Cells</td>
</tr>
<tr>
<td>OSM Water Bodies</td>
<td>Intersects</td>
<td>Natura (2000)</td>
</tr>
<tr>
<td>Natura (2000)</td>
<td>Within</td>
<td>Federal States of Germany</td>
</tr>
</tbody>
</table>

All the datasets will be first converted to RDF with GeoTriples!
References (1/3)

- [Bizer et al., IJSWIS’06]

- [Christen, TKDE’11]

- [Auer, RW’13]

- [Salas et al., TerraCognita’11]

- [Sehgal et al. GIS’06]
References (2/3)

• [Vilches-Blázquez et al., AGILE’12]

• [Ngonga Ngomo, ISWC’13]

• [Clementini et al., SSD'93]

• [Randell et al. KR’92]

• [Allen, Commun. ACM’83]
• [Isele et al., WebDB’11]
• [Papadakis et al, TKDE’13]
Thanks for your attention!
Questions?