Data Linking: Capturing and Utilising Implicit Schema-Level Relations

Andriy Nikolov
Victoria Uren
Enrico Motta
• Automatic instance matching algorithms
  – SILK, ODDLinker, KnoFuss, ...

• Pairwise matching of datasets
  – Requires significant configuration effort

• Transitive closure of links
  – Use of “reference” datasets
Problems

- Transitive closures often incomplete
  - Reference “hub” dataset is incomplete
  - Missing intermediate links
  - Direct comparison of relevant datasets is desirable
- Schema heterogeneity
  - Which instances to compare?
  - Which properties are relevant?
• KnoFuss architecture
Inferring schema mappings from pre-existing instance mappings

Utilizing schema mappings to produce new instance mappings

Background knowledge:
- Data-level (intermediate repositories)
- Schema-level (datasets with more fine-grained schemas)
• Step 1:
  – Obtaining transitive closure of existing mappings

LinkedMDB

movie:music_contributor/2490

MusicBrainz

music:artist/a16…9fdf

DBPedia

dbpedia:Ennio_Morricone

Algorithm
Step 2: Inferring class and property mappings

- **ClassOverlap** and **PropertyOverlap** mappings
  - Confidence (classes A, B) = \(|c(A) \cap c(B)| / \min(c(|A|), c(|B|))\)
    (overlap coefficient)
  - Confidence (properties r1, r2) = |c(X)|/|c(Y)|
    - X – identity clusters with equivalent values of r1 and r2
    - Y – all identity clusters which have values for both r1 and r2

**Diagram:**

```
LinkedMDB
  ^
  |    movie:is_a
  |      movie:is_a
  |                    movie:is_a
--- = =
MusicBrainz
  |
DBPedia
```

```
movie:music_contributor/2490 = music:artist/a16…9fd
```

```
dbpedia:Artist = dbpedia:Ennio_Morricone
```
Step 3: Inferring data patterns

Functionality restrictions

IF 2 equivalent movies do not have overlapping actors AND have different release dates THEN break the equivalence link

Note:
- Only usable if not taken into account at the initial instance matching stage
Algorithm

- Step 4: utilizing mappings and patterns
  - Run instance-level matching for individuals of strongly overlapping classes
  - Use patterns to filter out existing mappings

- **DBLP**
  SELECT ?uri
  WHERE {
  }

- **DBPedia**
  SELECT ?uri
  WHERE {
  }
• Class mappings:
  – Improvement in recall
    • Previously omitted mappings were discovered after direct comparison of instances

• Data patterns
  – Improved precision
    • Filtered out spurious mappings
    • Identified 140 mappings between movies as “potentially spurious”
    • 132 identified correctly
Limitations & future work

• Large-scale tests
  – Billion Triple Challenge 2009, other repositories

• Initial mappings
  – What to do if a repository is not connected to any other one?
  – Utilizing low-cost instance-matching techniques
Questions?

Thanks for your attention