# Object Property Matching utilizing the Overlap between Imported Ontologies

Benjamin Zapilko and Brigitte Mathiak

GESIS - Leibniz Institute for the Social Sciences
Unter Sachsenhausen 6-8, 50667 Cologne, Germany
{benjamin.zapilko,brigitte.mathiak}@gesis.org

**Abstract.** Large scale Linked Data is often based on relational databases and thereby tends to be modeled with rich object properties, specifying the exact relationship between two objects, rather than a generic is-a or part-of relationship. We study this phenomenon on government issued statistical data, where a vested interest exists in matching such object properties for data integration. We leverage the fact that while the labeling of the properties is often heterogeneous, e.g. ex1:geo and ex2:location, they link to individuals of semantically similar code lists, e.g. country lists. State-of-the-art ontology matching tools do not use this effect and therefore tend to miss the possible correspondences. We enhance the state-of-the-art matching process by aligning the individuals of such imported ontologies separately and computing the overlap between them to improve the matching of the object properties. The matchers themselves are used as black boxes and are thus interchangeable. The new correspondences found with this method lead to an increase of recall up to 2.5 times on real world data, with only a minor loss in precision.

**Keywords:** #eswc2014Zapilko

## 1 Introduction

The number of statistical data sets available as Linked Data has recently increased to a large degree. This is a welcome step towards governmental transparency, since professionals from many domains rely on the analysis of such data. Statistical data is periodically collected by administrative sources [30] as an attempt to describe the state of a nation in numbers, typically by collecting demographic and economic data, e.g. like population numbers and unemployment ratios but also subjective measurements like general wellbeing. One of the typical tasks for scientists using statistical data is the comparative analysis of more than one data set. Linked Open Data [13] is, in theory, a suitable source for this task as it allows the easy linking of data sets. In practice, only few links exist between these data sets and, as we will describe in Section 6, the correspondences created by matching tools have a rather low recall (0.4 on real world data). However, finding correspondences manually is much harder than dismissing wrong ones.

This systematic shortcoming is due to a high occurrence of heterogeneously labeled object properties, e.g. `ex1:geo` and `ex2:location`. The individuals linked to by object properties are not considered in full extent during ontology matching when they are part of external or separate ontologies like, e.g. code lists of country names maintained by a particular authority. Ontologies and instance data that are aligned in current benchmarks and alignment tasks of the Ontology Alignment Evaluation Initiative (OAEI) [29] do not yet address this problem. This is verified in Section 4 by comparing a large number of statistical data sets and the OAEI data sets. This critique on the current limitation on domains for ontology matching is not new. [34] suggests the consideration of new domains to reveal new challenges. Also, according to [12], domain-specific values, significant occurrences, patterns and constraints of values should also be considered.

Based on these ideas and our own findings on heterogeneous object properties, we develop a novel ontology matching method to improve the matching of object properties. The method utilizes an instance-based matcher as a core, but refines the results by matching the imported ontologies as well. The similarities between these imported ontologies are computed as an overlap score. This overlap score indicates whether a new correspondence between object properties is added to the generated correspondences between the input ontologies. This method allows us to detect additional correspondences between object properties like `ex1:geo` and `ex2:location` based on the individuals of imported ontologies. Thus, recall is increased. The approach is independent of the matching algorithm employed and may utilize any instance-based approaches or algorithms that consider extensional techniques and object similarity techniques [8].

We test different methods to calculate the overlap score: Jaccard Coefficient and three variants of it, finding that although some of the variants show clearer distinction between correct correspondences and false positive correspondences, the improvements are statistically not significant, especially not when comparing it to the influence of the matcher used.

We distinguish our method from current related work in Section 2. The problem statement of our approach is formulated in Section 3. It is supplemented by an use case and validated by a data analysis in Section 4. In Section 5, we describe our proposed algorithm in detail. Our method is evaluated in Section 6 in a benchmark scenario and a real world data scenario. In Section 7, we conclude and provide an outlook on future work.

## 2 Related Work

In the context of Linked Data, the matching of properties is not a trivial task as [33] argues, because the instances of two properties are typically described in ontologies that differ from those defining the properties. This observation can be adopted to ontologies when object properties are used to link to classes or individuals of another, imported ontology.

In this paper, we focus on instance-based matching of object properties. There are many established methods that perform instance-based matching and

apply extensional techniques like object similarities. Both, OLA [7] and Similarity Flooding [26], process input ontologies as graph structures and compute proximities between all elements of two graphs. These proximities are propagated throughout the graph structure. However, Similarity Flooding only detects correspondences between nodes of a graph, i.e. classes of an ontology, and does not perform property matching. COMA++ [6] contains two instance-based matchers which consider similarities and patterns of instance values. [28] presents an approach for matching RDF datatype properties based on the construction of a matrix of the property values. In [22], the domains and ranges of object properties, the property characteristics, and the cardinality restrictions are considered for computing similarities among properties. ASMOV [20] computes several similarities between properties like internal and extensional similarities. The instance values are part of an overall similarity measure consisting of four calculations. RiMOM [24] combines multiple strategies for ontology matching automatically and considers also instances for property matching. Detecting correspondences between attributes is also a traditional part in the domain of schema matching [18]. In the context of Linked Data, BLOOMS+ [19] uses contextual information from the input data for matching and a rich knowledge source. While BLOOMS+ focuses on linking classes only, ObjectCoref [15] and RAVEN [27] detect also similarities between property values. Additional prominent matching approaches are FALCON-AO [14], AgreementMaker [2], Semint [23], GLUE [4], and Dumas [1].

The above approaches have in common that only those individuals are considered for matching which are linked in the object properties of the input ontologies. In contrast, our approach identifies and considers additional individuals of an imported ontology that are not linked to in the object properties of the input ontologies. Another specific point of our approach is that we assume the imported ontologies to be sets of homogeneous entities like authority or code lists. This assumption will be verified in Section 4.

The computation of similarities between ontologies is discussed in several works. In [25], the ontology similarity is based on terminological similarity of concepts. Different similarities are combined in [5] where strings, concepts, and usage traces are considered. [35] presents a calculation between two A-Box ontologies, while also structural information out of their T-Box ontology is considered. Similar to our method is [3], where several measures are introduced for computing ontology similarity by considering available alignments. In [4], the Jaccard coefficient is introduced as a similarity measure for ontology matching. According to [17], simple similarity measures like the Jaccard coefficient perform best for instance-based matching which is why we chose it for our method.

The benchmark data set of the OAEI is an established source for evaluating ontology matching approaches. Based on an ontology describing bibliographic resources, it covers various kinds of transformations on structural and terminological levels and is used for different alignment tasks. The Islab Instance Matching Benchmark (IIMB) [11] has been created for evaluating instance matching systems. However, both benchmarks do not consider the underlying problem of our approach. Similar to the data in our use case is the RDF version [21] of the

Star Schema Benchmark [31] which comprises five single data sets. However, this distributed structure is not processible by most of the current ontology matching systems.

## 3 Problem Statement

The problem we address in this paper is illustrated in Figure 1. We assume two ontologies $O$ and $O'$ that hold classes $C$ and $C'$ with individuals $I_n$ and $I'_n$. We also assume that $R$ and $R'$ are ontologies with homogeneous entities $RC_n$ and $R'C'_n$ of the same type, e.g. authority or code lists. The individuals of the ontologies $O$ and $O'$ contain object properties $P$ and $P'$ that link to entities of the ontologies $R$ and $R'$. When matching ontologies $O$ and $O'$, correspondences between semantically similar object properties $P$ and $P'$ could be missed, when they are of different name and structure. This occurs although both object properties link to individuals of similar ontologies e.g. like `ex1: geo` and `ex2:location` linking to entities of country lists $R$ and $R'$.

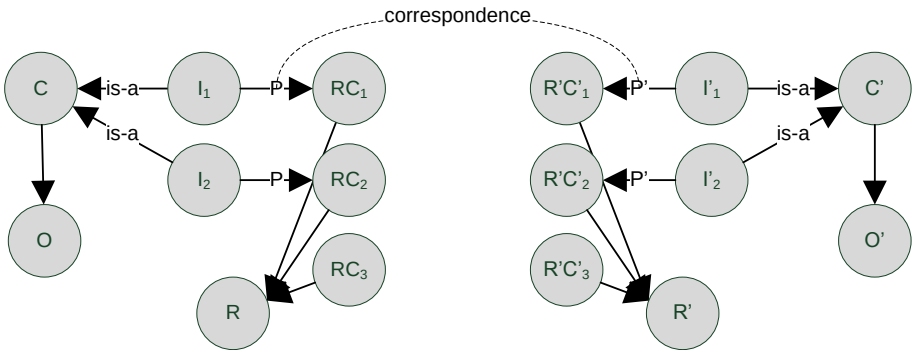


**Fig. 1.** Matching of object properties that link to individuals of imported ontologies.

As discussed in the related work, current approaches consider the individuals linked by object properties for ontology matching like ASMOV [20], RiMOM [24] and others. However, these referenced individuals play only a subsidiary role for the computation of a correspondence between the linking properties. Also, individuals of such imported ontologies which are not linked to are not considered. Thus, correspondences between object properties can be missed.

## 4 Use Case: Statistical Data

The use case, which supplements our problem statement, centers on statistical data. Scientists often integrate and merge two or more of these data sets in order to conduct comparative data analysis. In theory, ontology matching is the

ideal method for this task, however, in practice we show how matchers can be improved to give better results for this scenario.

Typically, statistical data is organized in a star or snowflake schema structure, which can be found in data warehouses [16] and is also reflected in the SDMX information model[1], a multidimensional standard model for describing statistical data. Represented as Linked Data, a statistical data set consists of several instances of data entries, each of which determines a particular data value, e.g. "548215". The data values are supplemented by additional objects which provide further information, e.g. in which country or at which time the data has been collected. This sets the data entries into context. Such objects are referenced in the data value instances by object properties. However, the objects themselves are classes or individuals of other external or separate data sets (typically classifications or code lists).

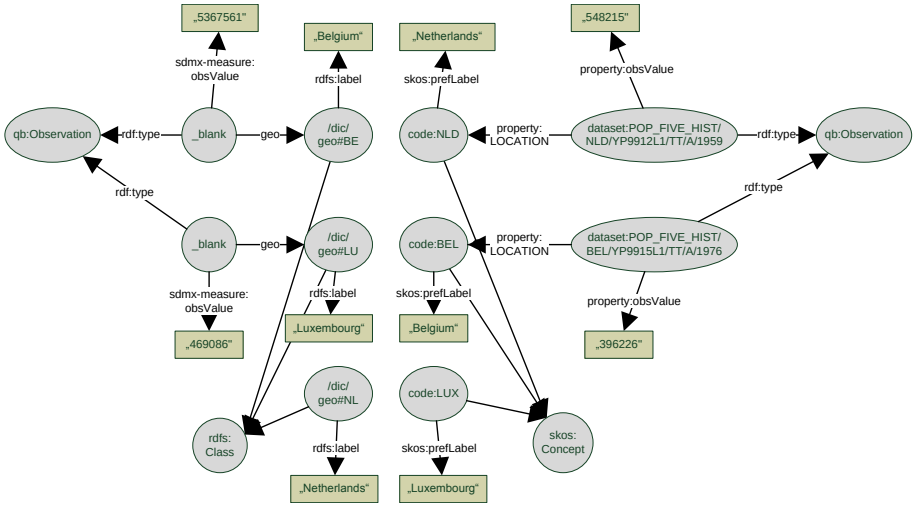


**Fig. 2.** Example of statistical data represented as Linked Data

In Figure 2, the problem of object property matching stated in the previous section is illustrated using real world statistical data. Excerpts of two data sets from Eurostat[2] and OECD[3] are shown. The instances of both data sets hold an object property indicating some geographical information (`geo` and `property:LOCATION`). Other object properties are omitted here. The object properties link to other individuals of code lists which is indicated by a different URI path and a different namespace. In Figure 2, the referenced data sets also contain the individuals `/dic/geo#NL` and `code:LUX`, which are not linked to

[1] http://sdmx.org/
[2] http://estatwrap.ontologycentral.com/
[3] http://oecd.270a.info/

by object properties. In our tests with matching systems, the object properties `geo` and `property:LOCATION` are not matched because they are labeled different and belong to different data sets. Even when matching the individuals of the referenced code lists, there is no inference on the referencing object properties.

In order to validate whether our problem statement is reasonable for the domain of statistical data by affecting a large amount of data sets, we verify our assumptions on the patterns of statistical data, which are:

1. Data entries are modeled as individuals and are accompanied by various named object properties linking to classes and individuals of external code lists or light-weight ontologies. Rather than forming a network or tree connected with homogeneous object properties, the data model is thus similar to a star schema [16].
2. Classifications and code lists[4] are often used in statistical data sets in the described way.
3. These code lists are referenced by object properties and are identifiable as additional ontologies or data sets by inspecting namespaces and URIs.

We verify our assumptions by analyzing and comparing data from three sources. Real world data sets are considered from two of the main repositories for Open Data: Data Hub[5] (DH)[6] and the wiki of Planet Data[7] (PD). They are compared to data sets used in previous campaigns of the OAEI [29] to show that this is in fact a novel problem, not one investigated before. Within this third set, we examine data sets of the instance matching (IM) tracks separately due to major differences between ontologies and data sets containing mostly instance data. Due to the diversity of the data sources, the data analysis was done manually with the help of standardized SPARQL queries and scripts.

**Table 1.** Comparison of statistical data and OAEI data (as of December 2013)

| **Criteria** | **DH** | **PD** | **OAEI** | **IM** |
|---|---|---|---|---|
| Number of all examined data sets | 49 | 22 | 54 | 15 |
| Data structure | 93,8 % | 95,4 % | 0 % | 13,3 % |
| Presence of thesauri references | 91,8 % | 95,4 % | 3,7 % | 13,3 % |
| OWL/RDF data set | 0 % | 0 % | 90,7 % | 40 % |
| Other RDF-based data set | 100 % | 100 % | 24,1 % | 73,3 % |

We investigate our data structure hypothesis by examining whether the data set is organized similar to our assumed pattern. The structure is detected by

[4] With regard to their similar function for statistical data classifications and code lists are summarized as code lists for the entire paper.

[5] http://thedatahub.io/

[6] Due to the amount of data sets, Data Hub has been analyzed by sampling. Data sets have been examined that are tagged with “format-rdf”, “format-qb”, “format-scovo” as well as “statistics”, “government”, “census”, or “lod” and similar spellings.

[7] http://wiki.planet-data.eu/web/data sets

analyzing and counting the links inside a data set and out to other data sets. The results in Table 1 show that most of the examined data sets from Planet Data and Data Hub reflect this typical structure of statistical data, but almost none from the OAEI and IM challenges. Based on the identified schema structure, we then investigate whether linkes to ontologies similar to code lists can be identified. References to code lists entries could be observed in most cases of the Data Hub and the Planet Data data sets (see Table 1). The detected code lists have a list-type character, like country lists, age groups, or entries of a scale. Only in a few cases, there are hierarchies inside these code lists, e.g. in a geographical classification with different administrative levels. In the OAEI and IM challenges, only in two cases references to code lists, i.e. object properties that link to individuals of imported ontologies, could be detected.

**Table 2.** Analysis of the structure of statistical data (as of December 2013)

| **Criteria** | **Percentage** |
|---|---|
| Number of all examined data sets (sample from DH and PD) | 40 |
| Different NS for input and referenced ontologies | 67,5 % |
| URI path of linked individuals equal for particular object properties | 100 % |
| Individuals of a referenced ontology of the same class | 100 % |

Finally, we examined whether the different ontologies of the detected structure can be distinguished by different namespaces and URIs. The individuals of an ontology are considered to be defined in one namespace. Moreover, the classes and individuals of an imported ontology have to be addressed by the same object property of the input ontology. The results in Table 2 show that this is indeed the case. For all studied data sets, looking at the URI path was sufficient to identify and distinguish the ontologies.

## 5 Computing Overlaps for Object Property Matching

Knowing the structural differences between current benchmarks and statistical data according to our problem statement, leads us to the following algorithm to improve the matching of object properties. Revisiting our use case, we complement the matching process of the two data sets by identifying those code lists that contain the referenced individuals like `/dic/geo#BE` and `code:NLD`. Then, the overlap between these code lists is computed which we conjecture to represent a semantic similarity between the object properties `geo` and `property:LOCATION`. This is used as correspondence for the overall matching between the data sets.

The algorithm is formalized as follows. Given as input are two ontologies $O$ and $O'$ with classes $C$ and $C'$, properties $P$ and $P'$, and individuals $I$ and $I'$. The objects $RC$ and $R'C'$ of the object property instances are classes or individuals of imported ontologies $R$ and $R'$. These are ontologies with homogeneous entities

of the same type, e.g. code lists. The imported ontologies are either T-Boxes or A-Boxes of their own with different namespaces. Thus, based on the data analysis conducted in Section 4 we formulate the following additional definitions.

**Definition 1 (Object Property Instance and Property Object).** *An instance $OPI$ of an object property $P$ is a tuple of the form (I,P,RC), where $I$ is an individual of ontology $O$ and $P$ is the particular object property of $O$. A property object $RC$ of $OPI$ is a class or individual of a referenced ontology $R$.*

**Definition 2 (Imported Ontology).** *An imported ontology $R$ is either a T-Box or A-Box ontology with classes or individuals $RC$ which are objects in the object property instances $OPI$ of the ontology $O$. An imported ontology $R$ and its entities $RC$ are held in a namespace different from the namespace of $O$ and all its entities.*

The objective of our algorithm is to detect an alignment $A$ as output with correspondences between all entities of $O$ and $O'$. Additionally, overlaps between all $R_n$ are used in order to generate additional correspondences between object properties $P$ and $P'$. In the algorithm, we apply any given ontology matching system that generates correspondences between two input ontologies. As mentioned before, the matcher is used as a black box in our algorithm. The algorithm goes through five phases for matching two input ontologies $O$ and $O'$.

1. All $RC$ inside each ontology are grouped in order to identify the imported ontologies $R_n$ and $R'_m$ per each ontology $O$ and $O'$.
2. The input ontologies $O$ and $O'$ are matched by an ontology matching tool. The resulting correspondences are included to the alignment $A$.
3. All pairs of $R_n$ and $R'_m$ are matched with each other by the same matcher. The resulting correspondences are the basis for calculating the overlap scores in the next phase.
4. Overlap scores are computed pairwise for each $R_n$ and $R'_m$. Different similarity measures can be applied. We utilize the Jaccard coefficient [32, 4]. However, the Jaccard coefficient is known for its unbalancy [17], especially when two sets are highly different in their size. This may complicate the choice of a suitable threshold. Hence, we introduce three additional similarity measures for addressing this problem in Definition 3. The overlap between two ontologies is computed by assuming that a correspondence between two individuals of the ontologies indicates that they are part of the intersection set of $R_n$ and $R'_m$. This way, we can determine $|R_n \bigcap R'_m|$. If the overlap is higher than a specific threshold $t$, we assume that there is a correspondence between the object properties $P$ and $P'$ that hold $R_n$ and $R'_m$ as objects in $OPI$ and $OPI'$.
5. We add the detected correspondence with the calculated overlap score between their imported ontologies $R_n$ and $R'_m$ as confidence value to the alignment $A$. If a correspondence between two object properties already exists in $A$, the correspondence with the higher confidence value is kept and the other one is discarded.

**Definition 3 (Overlap utilizing Jaccard coefficient and variations).** *The overlap between two imported ontologies $R_n$ and $R'_m$ is computed as*

$$JC = \frac{|R_n \bigcap R'_m|}{|R_n \bigcup R'_m|}$$

$$JC_{min} = \frac{|R_n \bigcap R'_m|}{|min(|R_n|, |R'_m|)|}$$

$$JC_{res} = \frac{|R_n \bigcap R'_m|}{|R_{n-Linked} \bigcup R'_{m-Linked}|}$$

$$JC_{min+res} = \frac{|R_n \bigcap R'_m|}{|min(|R_{n-Linked}|, |R'_{m-Linked}|)|}$$

*where*

- $|R_n \bigcap R'_m|$ *is the number of all correspondences between* $R_n$ *and* $R'_m$*,*
- $|R_n \bigcup R'_m|$ *is the number of all entities in* $R_n$ *and* $R'_m$ *and*
- $R_{n-Linked}$ *and* $R'_{m-Linked}$ *are only those classes of* $R_n$ *and* $R'_m$ *that are referenced in the ontologies* $O$ *and* $O'$*.*

**Definition 4 (Correspondence between two Object Properties).** *A correspondence between two object properties* $P$ *and* $P'$ *is described by the following 5-tuple adopted from [8].*

$$\langle id, e_1, e_2, r, n \rangle$$

*where*

- *id is an identifier for the particular correspondence;*
- $e_1$ *and* $e_2$ *are the object properties* $P$ *and* $P'$*;*
- $r$ *determines the type of the relation between* $P$ *and* $P'$*, in our case an equivalence relationship;*
- $n$ *represents the confidence values, which is in our case the* $overlap(R_n, R'_m)$*.*

This method is simple to implement with any instance-based matcher and enables us to match object properties like `geo` and `property:LOCATION` in our example. The runtime is comparable to matching the whole ontologies. The split between the different ontologies decreases the time needed for matching the particular ontologies, offsetting the need to run additional matching processes.

## 6 Evaluation

We evaluate our method on both artificial and real world data to show the impact of our method on object property matching. The results show a significant improvement in both scenarios, especially the sought-after improvement of recall.

### 6.1 Setup

The evaluation consists of two scenarios. The first scenario "Benchmark" is conducted on an artificially created benchmark for statistical data which is introduced in Section 6.2. In the second evaluation scenario "Real World Data", we apply our method on the two real world data sets from Eurostat and OECD from our use case. In each scenario, the matching systems are executed with the input ontologies at first ("State-of-the-Art"). In a second run, our method ("Object Property Matching") is applied by matching the imported ontologies additionally.

In both scenarios, the resulting correspondences are validated with their particular reference alignments. We compute precision, recall and F-measure for each alignment task, since they are standard evaluation measures for ontology matching evaluation [8]. For computing the overlap value, we utilize a threshold of 0.3 in the benchmark scenario. This has turned out to be a suitable value during pretests. Because the Jaccard coefficient can get unbalanced [17], we compare the different similarity measures defined in Section 5 in the second scenario.

We chose FALCON-AO [14] and AgreementMaker [2] as black box matcher from which we assume representative results. FALCON-AO has been chosen because of applying extensional matching techniques like object similarity, while AgreementMaker contains an instance-based matching algorithm. Our instance-based object property matching approach is compared best to those techniques. Both systems have been successful regarding their performances in previous OAEI campaigns [9, 10] and are executed without any manipulation in their standard configurations.

### 6.2 Benchmark

It was not possible to evaluate our method on a gold standard, because unfortunately no such standard exists yet. The OAEI data sets and other data sets used for evaluations lack important characteristics we are looking for (see Sections 2 and 4). Hence, we decided to design a benchmark specific to the problem based on the principles of established benchmarks [29, 11].

The benchmark reflects the assumptions made in Section 4 concerning heterogeneous object properties and their linking to classes of code lists, located in other namespaces and URI paths. The T-Box is a simplified version of a data model for statistical data: one named class representing a data entry and several object properties linking to classes of imported ontologies. These imported ontologies are included with different URI paths. We populate this seed ontology with 50 randomly generated individuals as A-Box. An example is given in Figure 3: `:Entry11` represents an observation on the satisfaction level of German young adults. The object properties link to classes of code lists from different namespaces[8].

[8] In the actual benchmark, they are differentiated by URI path not by namespace as this has the greater coverage on statistical data. This example uses namespaces for clarification. For the algorithm, there is no practical difference.

Individual of the Seed Ontology

```
:Entry11  a  STATTBOX:DataEntry ,
             owl:NamedIndividual ;
          STATTBOX:date  "1981/08/02"^^xsd:integer ;
          STATTBOX:obsValue  "886"^^xsd:integer ;
          STATTBOX:agegroup  ages:20-29 ;
          STATTBOX:gender  sex:sex-M ;
          STATTBOX:geo  countriesISO:DE ;
          STATTBOX:maritalStatus  concepts:cl_mar_total ;
          STATTBOX:occupation  indic_1:occup_value3 ;
          STATTBOX:satisfaction  indic_2:sat_value4 .
```

Classes of Referenced Ontologies

```
ages:20-29  a  owl:Class ;
rdfs:label  "From 20 to 29 years" .

sex:sex-M  a  owl:Class ;
rdfs:label  "Male" .

countriesISO:DE  a  owl:Class ;
rdfs:label  "Germany" .

concepts:cl_mar_total  a  owl:Class ;
rdfs:label  "Total" .

indic_1:occup_value3  a  owl:Class ;
rdfs:label  "Unemployed" .

indic_2:sat_value4  a  owl:Class ;
rdfs:label  "Very dissatisfied" .
```

**Fig. 3.** Example individual of the seed ontology

The seed ontology is used to produce variations. The namespaces of all involved code lists, i.e. imported ontologies, were changed. Additionally, specific properties were changed in accordance to what we have observed about statistical data. In the variations 010 - 011, the names of the object properties are changed on a random basis. In 020 - 024, the code lists that are referenced are changed in label name, class name, URI path, etc. This notably lowers the overlap. In 030 - 031, we test the matching without any overlap, to test how our system works on standard ontologies. Each variation forms together with the seed ontology an alignment task. The complete benchmark, the variations and the single tests are available at http://code.google.com/p/matching-statistics/.

### 6.3 Real World Data Sets

For the real world data scenario, we revisit the data sets from our use case. They hold many different properties that semantically overlap and are representative for statistical data. The idea is to examine many different cases in just one pair of data sets, as the preparation is quite labor-intensive. The EUROSTAT data set covers “Labour input in industry”. This data set has 16783 instances and 7 object properties. The OECD data set covers “Outward activity of multinationals - Share in national total (manufacturing)”. It has 5343 instances and 8 object properties. In both data sets, the object properties link to classes of particular code lists. Also, both data sets have some object properties that are not linked inside the actual instances. We manually identified five properties that match semantically.

In order to use the code lists with the matching systems, they had to be preprocessed. The changes include generic transformations of the referenced code lists from SKOS to T-Box ontologies. Similar preprocessing has been previously done in Library Tracks of OAEI, where SKOS thesauri have been transformed to OWL. The data is available at http://code.google.com/p/matching-statistics/.

### 6.4 Results and Discussion

The results in Table 3 indicate major improvements on matching object properties in all scenarios. The results of the tests 010 - 011, which hold differently labeled object properties, expose the strengths of our method compared to the state-of-the-art. The matchers could not find correspondences between the heterogeneous object properties, even if their referenced individuals are equal or similar like `concepts:geo#geo_DE` and `vocab:country#DE`. The information given in the labels of these classes is not considered for detecting correspondences between the referring object properties. The recall of our method is much higher for these tests. The results of the tests 020 - 024 show that the distance between our method and the state-of-the-art is decreasing depending on the matching between the imported ontologies and the resulting different overlaps. However, the results of these tests are always better or at least equal to the state-of-the-art approach when utilizing our method. This is also demonstrated with the counter check 030 - 031 (no overlap), which shows at least no worsening.

**Table 3.** Results for both evaluation scenarios. The best result in row is bold. For the single tasks of the variations the means have been computed. P = Precision, R = Recall, F = F-measure

| **Approach** | **State-of-the-Art (SotA)** | | | | | | **Object Property Matching** | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| System | AgreementMaker | | | FALCON-AO | | | AgreementMaker | | | FALCON-AO | | |
| | P | R | F | P | R | F | P | R | F | P | R | F |
| Test 001 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Tests 010-011 | 1.00 | 0.45 | 0.61 | 1.00 | 0.34 | 0.46 | 1.00 | 0.89 | **0.94** | 1.00 | 0.78 | 0.87 |
| Tests 020-024 | 1.00 | 0.42 | 0.59 | 1.00 | 0.29 | 0.41 | 1.00 | 0.85 | **0.92** | 1.00 | 0.67 | 0.79 |
| Tests 030-031 | 1.00 | 0.45 | **0.61** | 1.00 | 0.34 | 0.46 | 1.00 | 0.45 | **0.61** | 1.00 | 0.34 | 0.46 |
| Real World Data | 1.00 | 0.40 | 0.70 | 1.00 | 0.40 | 0.70 | 0.83 | 1.00 | **0.92** | 0.45 | 1.00 | 0.73 |

The results using real world data are similar to the benchmark tests 020 - 024, because there are not necessarily overlaps between the code lists. The object properties in both data sets are named differently, the number of classes in all code lists is unbalanced, and there may not be necessarily correspondences between all object properties. While recall improves, there is some loss of precision (see Table 3). False positives occur when the matchers find correspondences between unlike code lists, e.g. `geo` (containing country names) of Eurostat with `property:ISIC3` (containing branches of industry) of OECD. Nevertheless, the higher recall shows that by our method new correspondences have been detected that have not been identified by the state-of-the-art approach.

In order to cut off these false positives, we choose a threshold value. Since the unbalance of the simple Jaccard coefficient makes it difficult to set a suitable threshold, we have compared the similarity measures defined in Definition 3 regarding their impact on the real world data scenario.

**Table 4.** Similarity Measures for detected Correspondences (AgreementMaker)

| **Found Correspondences** | $JC$ | $JC_{min}$ | $JC_{res}$ | $JC_{min+res}$ | $SotA$ |
|---|---|---|---|---|---|
| *Correct Correspondences* | | | | | |
| geo = LOCATION | 0.002 | **1** | 0.688 | **1** | 0 |
| indic_bt = VAR | 0.132 | 0.909 | **1** | **1** | 0 |
| nace_r2 = ISIC3 | 0.006 | 0.979 | 0.959 | **1** | 0 |
| obs_status = OBS_STATUS | 0.75 | **1** | x | x | 0.969 |
| timeformat = TIME_FORMAT | 0.571 | **1** | **1** | **1** | 0.872 |
| *False Positives* | | | | | |
| geo = ISIC3 | 0.004 | 0.354 | 0.369 | 1 | 0 |

**Table 5.** Similarity Measures for detected Correspondences (FALCON-AO)

| **Found Correspondences** | $JC$ | $JC_{min}$ | $JC_{res}$ | $JC_{min+res}$ | $SotA$ |
|---|---|---|---|---|---|
| *Correct Correspondences* | | | | | |
| geo = LOCATION | 0.002 | 0.909 | 0.588 | **0.909** | 0 |
| indic_bt = VAR | 0.012 | 0.090 | 0.083 | **0.333** | 0 |
| nace_r2 = ISIC3 | 0.007 | 0.188 | 0.103 | **0.191** | 0 |
| obs_status = OBS_STATUS | 0.647 | **0.917** | x | x | 1 |
| timeformat = TIME_FORMAT | 0.571 | **1** | **1** | **1** | 1 |
| *False Positives* | | | | | |
| geo = ISIC3 | 0.004 | 0.354 | 0.340 | 1 | 0 |
| nace_r2 = VAR | 0.001 | 0.090 | 0.017 | 0.1 | 0 |
| nace_r2 = OBS_STATUS | 0.012 | 0.938 | 0.306 | 0.306 | 0 |
| freq = VAR | 0.053 | 0.111 | 0.1 | 0.1 | 0 |
| freq = OBS_STATUS | 0.389 | 0.778 | x | x | 0 |
| freq = TIME_FORMAT | 0.3 | 0.75 | 1 | 1 | 0 |

The different overlap values computed for each detected correspondence are shown in Tables 4 and 5 and are compared to the confidence values of the state-of-art approach (SotA). An `x` means that no value could have been computed, because no classes of the referenced code lists have been linked in the data set (this would result in a divide by zero). Balanced values make it easier to distinguish false positives, because the difference between valid correspondences and non-valid correspondences is increased. For example, the overlap for the correspondence between `geo` and `LOCATION` is at 0.002 for $JC$, but much higher for the others. The best approach, in this sample, would be to use $JC_{min+res}$ and to use $JC_{min}$, when that fails. However, the actual effect is minimal. Only one false positive is excluded. More thorough testing might bring a clearer distinction. So far, it seems that the choice of the similarity measure to compute an overlap is much less relevant than the choice of the matcher to increase precision.

## 7 Conclusion and Outlook

In this paper, we have shown that object properties in statistical data are used differently than in data sets typically used for ontology matching. By leveraging this difference for object property matching, we gain an improvement of recall up to 2.5 times. Loss in precision occurs, but is relatively small in comparison. Since this loss occurs while matching the imported ontologies, adjusting the matching systems towards this problem may be helpful. For these experiments, we have used the standard parameters for both matchers, in order to keep it clearer.

While our use case has been motivated by statistical data, a lot of Linked Data sources share this data model structure, since many of them are derived from relational databases. We chose statistical data, because 1) there is clear need to integrate the data and 2) although the data sets are covering semantically similar topics, standardization usually does not cover the object properties, only the code lists themselves, if at all. This demand may increase with the number of Linked Open Data sets.

## References


1. Bilke, A., Naumann, F.: Schema Matching using Duplicates. ICDE, 2005, 69-80
2. Cruz, I., Antonoelli, F., Stroe, C.: AgreementMaker: Effcient Matching for Large Real-World Schemas and Ontologies, VLDB 09, 2009
3. David, J., Euzenat, J., Šváb-Zamazal, On.: Ontology similarity in the alignment space. Proceedings of the 9th international semantic web conference on The semantic web, 2010, 129-144
4. Doan, A., Madhavan, J., Domingos, P., Halevy, A.: Ontology Matching: A Machine Learning Approach. In Staab, S., Studer, R. (Eds.): Handbook on Ontologies in Information Systems, Springer, 2003, 397-416
5. Ehrig, M., Haase, P., Stojanovic, N., Hefke, M.: Similarity for Ontologies - a Comprehensive Framework. European Conference on Information Systems (ECIS), 2005
6. Engmann, D., Maßmann, S.: Instance Matching with COMA++. BTW Workshops, 2007, 28-37
7. Euzenat, J., Valtchev P.: Similarity-Based Ontology Alignment in OWL-Lite. Proceedings of the 16th Eureopean Conference on Artificial Intelligence, ECAI'2004, 333-337
8. Euzenat, J., Shvaiko, P.: Ontology matching. Springer-Verlag, 2007
9. Euzenat, J., Ferrara, A., Meilicke, C., Pane, J., Scharffe, F., Shvaiko, P., Stuckenschmidt, H., Sváb-Zamazal, O., Svátek, V., dos Santos, C. T.: Results of the Ontology Alignment Evaluation Initiative, 2010
10. Euzenat, J., Ferrara, A., van Hage, W. R., Hollink, L., Meilicke, C., Nikolov, A., Ritze, D., Scharffe, F., Shvaiko, P., Stuckenschmidt, H., Sváb-Zamazal, O., dos Santos, C. T.: Results of the ontology alignment evaluation initiative, 2011
11. Ferrara, A., Lorusso, D., Montanelli, S., Varese, G.: Towards a Benchmark for Instance Matching. In Shvaiko, P.; Euzenat, J., Giunchiglia, F., Stuckenschmidt, H. (Eds.): Ontology Matching (OM 2008), CEUR-WS.org, 2008, 431
12. Halevy, A.: Why Your Data Won't Mix Queue, ACM, 2005, 3, 50-58
13. Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space. Morgan & Claypool, 2011

14. Hu, W.; Qu, Y. Falcon-AO: A practical ontology matching system. Web Semant., Elsevier, 2008, 6, 237-239
15. Hu, W., Chen, J., Cheng, G., Qu, Y.: Objectcoref and falcon-ao: results for oaei 2010. Ontology Matching 2010, 2010
16. Inmon, W. H.: Building the Data Warehouse. John Wiley & Sons, Inc., 2005
17. Isaac, A., Van Der Meij, L., Schlobach, S., Wang, S.: An empirical study of instance-based ontology matching. Proceedings of the 6th ISWC and 2nd ASWC, Springer-Verlag, 2007, 253-266
18. Rahm, E.; Bernstein, P. A.; Rahm, E.; Bernstein, P. A. A survey of approaches to automatic schema matching. The VLDB Journal, Springer, 2001, 10, 334-350
19. Jain, P., Yeh, P.Z., Verma, K., Vasquez, R.G., Damova, M., Hitzler, P., Sheth, A.P.: Contextual Ontology Alignment of LOD with an Upper Ontology: A Case Study with Proton. 8th Extended Semantic Web Conference, ESWC 2011, 2011
20. Jean-Mary, Y.R., Shironoshita, E.P., Kabuka, M.R.: Ontology matching with semantic verification. Web Semant. September, 2009, 235-251
21. Kämpgen, B., Harth, A.: No Size Fits All - Running the Star Schema Benchmark with SPARQL and RDF Aggregate Views. 10th Extended Semantic Web Conference, ESWC 2013, 2013
22. Leme, L.A.P.P., Casanova, M.A., Breitman, K., Furtado, A.L.: Instance-Based OWL Schema Matching. 11th International Conference on Enterprise Information Systems, ICEIS 2009, 2009
23. Li, W., Clifton, C.: SEMINT: a tool for identifying attribute correspondences in heterogeneous databases using neural networks. Data Knowl. Eng., April 2000, 33, 1, 49-84
24. Li, J., Tang, J., Li, Y., Luo, Q.: RiMOM: A Dynamic Multistrategy Ontology Alignment Framework. IEEE Trans. on Knowl. and Data Eng. August 2009
25. Maedche, A., Staab, S.: Measuring Similarity between Ontologies. Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW), 2002, 251-263
26. Melnik, S., Garcia-Molina, H., Rahm, E.: Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching. Proceedings of the 18th International Conference on Data Engineering, 2002
27. Ngomo, A., Lehmann, J., Auer, S., Honer, K.: Raven - active learning of link specications. Ontology Matching 2011, 2011
28. Nunes, B.P., Caraballo, A.A.M., Casanova, M.A., Breitman, K., Leme, L.A.P.P.: Complex matching of RDF datatype properties. Proceedings of the 6th International Workshop on Ontology Matching, Bonn, Germany, 2011
29. Ontology Alignment Evaluation Initiative, http://oaei.ontologymatching.org/
30. OECD, IMF, ILO, Interstate Statistical Committee of the Commonwealth of Independent States: Measuring the Non-Observed Economy: A Handbook, Annex 2, Glossary, 2002
31. ONeil, P., ONeil, E., Chen, X.: Star Schema Benchmark - Revision 3. Tech. rep., UMass, Boston, 2009, http://www.cs.umb.edu/ poneil/StarSchemaB.pdf
32. van Rijsbergen, C. J.: Information Retrieval. Butterworth, 1979
33. Rong, S., Niu, X., Xiang, E.W., Wang, H., Yang, Q., Yu, Y.: A machine learning approach for instance matching based on similarity metrics. Proceedings of the 11th international conference on The Semantic Web - Volume Part I., 2012
34. Shvaiko, P., Euzenat, J.: Ontology Matching: State of the Art and Future Challenges. IEEE Transactions on Knowledge and Data Engineering, 2011, 158-176
35. Stuckenschmidt, H.: A Semantic Similarity Measure for Ontology-Based Information. FQAS, 2009