Evaluation of Instance Matching Tools: The Experience of OAEI

A. Ferrara[a], A. Nikolov[b], J. Noessner[c], F. Scharffe[d]

[a] *DICo, Università degli Studi di Milano, Via Comelico 39, 20135 Milano, Italy*
[b] *Knowledge Media Institute, The Open University Walton Hall, Milton Keynes, MK7 6AA, United Kingdom fluid Operations AG, Altrottstrasse 31, 69190 Walldorf, Germany*
[c] *KR & KM Research Group University of Mannheim, B6 26, 68159 Mannheim, Germany*
[d] *LIRMM, University of Montpellier 161 rue Ada, 34095 Montpellier Cedex 5, France*

Abstract

Nowadays, the availability of large collections of data requires techniques and tools capable of linking data together, by retrieving potentially useful relations among them and helping in associating together data representing same or similar real objects. One of the main problems in developing data linking techniques and tools is to understand the quality of the results produced by the matching process. In this paper, we describe the experience of instance matching and data linking evaluation in the context of the Ontology Alignment Evaluation Initiative (IM@OAEI). Our goal is to be able to validate different proposed methods, identify most promising techniques and directions for improvement, and, subsequently, guide further research in the area as well as development of robust tools for real-world tasks.

Keywords: instance matching evaluation, data linking, semantic web

1. Introduction

Problems concerning the automatic matching of data and ontology instances, such as data linking and instance matching, are becoming crucial for the future directions of the Semantic Web and the Web in general. The availability of large collections of data requires techniques and tools capable of linking data together, by retrieving potentially useful relations among them and helping in associating together data representing same or similar real objects. One of the main problems in developing these kind of techniques and tools is to have a methodology and a set of benchmarks for understanding the quality of the results produced by the matching process. Moreover, matching tools developers need a framework in which their tools can be compared with other similar tools on the same data in order to understand where improvements and new solutions are possible and needed. We addressed this needs by organizing the Instance Matching track of the Ontology Alignments Evaluation Initiative (IM@OAEI).

In the remainder of this section we first give a short background outlining the instance matching problem in the context of semantic data. Then, we present the requirements for evaluating instance matching and data linking approaches.

1.1. Instance matching problem

The instance matching problem can be informally defined as an operation which takes two collections of data as input and produces a set of *mappings* between entities of the two collections as output. Mappings denote binary relations between entities which are considered equivalent one to another. The following can serve as a high-level formal definition:

Email addresses: alfio.ferrara@unimi.it (A. Ferrara), a.nikolov@open.ac.uk (A. Nikolov), jan@informatik.uni-mannheim.de (J. Noessner), francois.scharffe@lirmm.fr (F. Scharffe)

***Definition**: Let $\mathcal{D}_1$ and $\mathcal{D}_2$ represent two datasets, each one containing a set of data individuals $\mathcal{I}_i$ and structured according to a schema $\mathcal{O}_i$. Each individual $I_{ij} \in \mathcal{I}_i$ describes some entity ω_j. Two individuals are said to be equivalent $I_j \equiv I_k$ if they describe the same entity $\omega_j = \omega_k$ according to a chosen identity criterion. The goal of the entity resolution task is to discover all pairs of individuals $\{(I_{1i}, I_{2j}) | I_{1i} \in \mathcal{I}_1, I_{2j} \in \mathcal{I}_2\}$ such that $\omega_{1i} = \omega_{2j}$.*

Actual format of data individuals depends on the format of the datasets $\mathcal{D}_i$. If $\mathcal{D}_i$ represent relational databases, then database records serve as individuals I_i and are identified by the primary key values. In the context of Semantic Web data, datasets $\mathcal{D}_i$ are represented by RDF graphs. Individuals $I_i \in \mathcal{I}_i$ are identified by URIs and described using the classification schema and properties defined in the corresponding ontology $\mathcal{O}_i$.

The task can be approached using different types of available evidence. Based on this, existing techniques for instance matchning can be classified into three main categories [16]:

- *Value matching.* These techniques usually serve as basic building blocks of data linking tools and focus on identifying equivalence between property values of instances. Typical examples of these techniques are string similarity metrics like edit distance or Jaro-Winkler.

- *Individual matching.* These techniques decide whether two individuals represent the same real-world object. They operate with descriptions of a pair of individuals, which may contain multiple attributes. They utilise aggregation of similarities between corresponding property values.

- *Dataset matching.* These techniques take into account all individuals in two datasets and try to construct an optimal alignment between these whole sets of individuals. They rely on the results of the individual matching and can further refine them. These techniques utilize different methods such as similarity propagation, optimization algorithms, logical reasoning, etc.

Existing tools normally combine different techniques from several categories and aim to specify an instance matching workflow which would yield high quality results. Detailed surveys of the instance matching techniques and tools created in the databse community can be found in [12] and [25], while [16] lists the approaches applied to the Semantic Web domain and linked data.

1.2. Requirements for the evaluation of instance matching and data linking approaches

The main goal of developing common evaluation approaches for data linking is to be able to validate different proposed methods, identify most promising techniques and directions for improvement, and, subsequently, guide further research in the area as well as development of robust tools for real-world tasks. In order to achieve this, the evaluation procedure must satisfy several requirements, which present non-trivial challenges.

The first subset of these requirements relates to the *representative capabilities* of the evaluation approach. Evaluation results must provide useful information about expected performance of evaluated tools and techniques if they are applied to other real-world tasks as well as compare different methods and choose the best suited ones. The research performed in the area of data linking primarily builds on top of two research areas: database record linkage [14] and ontology matching [13, 17]. In both these areas, the primary evaluation approach, which fits these requirements, is benchmarking, where a set of pre-defined tests are used on which the results produced by methods can be measured with respect to a well-defined scale. However, real-world tasks can differ with respect to many parameters: for example, domain of processed datasets, richness of information represented in these datasets, dataset size, chosen data format, availability of background knowledge which can be utilized. These parameters present different challenges to the data linking tools. Given the multitude of different possible combinations of these parameters, it is impossible to devise a single benchmarking test which would approximate all the variety of different real-world tasks. The set of benchmarking tests used for evaluation of data linking tools must aim at achieving two diverse goals:

- being *comprehensive*: including as many challenges occurring in real-world matching tasks, as possible (e.g., diversity of data formats, attributes, schemas)

- being *illustrative*: reflect the distribution of different data features similar to the most likely

parameters of real-world tasks, i.e., a data feature which rarely occurs in real-world tasks should not dominate test data.

Besides the parameters of test data, the evaluation procedure must utilize appropriate evaluation criteria. In existing research, the most important criterion corresponds to the quality of data linking results. This is normally measured in terms of precision (proportion of correct mappings among the method results) and recall (proportion of correct mappings identified by the tool among all actual mappings). However, other evaluation criteria can be important as well: for example, given the large volumes of information which must be processed on the web of data, the computation time required by the linking tool becomes an orthogonal evaluation dimension.

The second kind of requirements are *pragmatic* and related to the evaluation procedure itself. In particular, the evaluation procedure should not require extensive effort, as this would discourage both tool developers and users from adopting it. Moreover, the chosen approach should ensure that the evaluation results are correctly measured with respect to the chosen criteria. This is not always trivial to achieve, especially with real-world data: e.g., constructing the gold standard alignments is problematic for large-scale datasets.

Since these requirements to some extent are mutually contradictory, it is hardly possible to satisfy all them to the full extent. However, while devising a reusable evaluation method, it is necessary to consider all of them and aim at a reasonable compromise. Significant relevant work on the evaluation of both record linkage and ontology matching tools has been performed in the corresponding domains. In order to determine suitable evaluation methods for instance matching tools in the semantic web domain, it is necessary to consider these approaches first and analyze the possibilities for reuse.

The paper is organized as follows: in Section 2, we discuss the main contributions given in literature to the problem of evaluating instance matching tools and techniques. In Section 3, we present the experience of instance matching evaluation in the context of OAEI. In Section 4, we present our real-data benchmark, while in Section 5 we present our approach for the generation of automatically generated benchmarks. In Section 6, we discuss the main open issues and future directions in the field of instance matching evaluation. Finally, in Section 7, we give our concluding remarks.

2. State of the art

The task of data linking is closely related both to the record linkage problem studied in the database community and to the ontology matching task. Given that both these tasks require approximate methods, their evaluation requires estimating the output quality by performing experiments with realistic benchmarks. The evaluation initiatives performed to evaluate tools developed in these areas are relevant to the data linking domain both because they have to deal with similar requirements and because they can be partially reused.

2.1. Evaluation initiatives in the database community

In parallel with the development of record linkage algorithms, work on their evaluation has been conducted in the database community for a long time. Evaluation test sets used to validate these methods can generally be classified into two types:

- Real-world data sources. Usually, such a benchmark includes two or more publicly available datasets which originate from different sources but describe the same domain. Gold standard mappings between records in these datasets are either created manually or validated manually after an initial automatic generation of candidates.

- Artificially generated datasets. An artificially generated benchmark is normally created by taking one reference dataset in advance and introducing artificial distortions into it in a controlled way: e.g., by removing/adding the attributes and changing their values randomly. In this way, creation of the gold standard set of mappings is straightforward: it includes all mappings between an original record and its distorted version.

Both approaches have their advantages and disadvantages. Taking real-world data sources allows the matching techniques to be evaluated in realistic conditions. In particular, this concerns the presence of heterogeneity problems, such as specific format differences, missed and incorrect attribute values, as well as the distribution of these problems in the dataset. However, this approach also

has its disadvantages: given that the parameters of the datasets depend on the domain, it is difficult to generalize the results obtained on these datasets to other application cases. Moreover, creating the set of gold standard mappings for such datasets is problematic. Ideally, all mappings have to be checked manually, which is difficult to achieve for large-scale datasets. On the other hand, artificially generated datasets provide fully controlled test conditions, in which matching challenges can be added at will. However, matching problems in the artificially generated datasets usually do not cover domain-specific features, and the distribution of these problems can be unrealistic.

During the earlier stages of research public databases describing the domain of scientific publications were particularly popular as sources of evaluation data. This was caused both by the fact that citation matching is one of particularly important application domain in academic environment as well as by the public availability of these datasets. In particular, Cora[1] is one of the commonly used evaluation datasets in the database domain (e.g., used in [33], [32], [9]). It includes citations collected from the Web, which referenced a set of pre-selected academic papers. Another dataset, ACM-DBLP was used in [24], [36] and created by matching references to the same publications mentioned in the ACM[2] and DBLP[3] web repositories. Several other datasets used by different researchers to evaluate their tools separately were collected within the RIDDLE repository[4].

These datasets have also been adapted to the semantic web standards and used to evaluate the instance matching algorithms in the semantic web domain: e.g., Cora was used in [9], [29], and [19], while the Restaurants dataset from the RIDDLE repository was used by [34]. The advantage of reusing these is the possibility to compare with the techniques developed in the database community, despite the differences in the format of processed data. However, these benchmark datasets are not fully representative of the challenges of the linked data environment. In particular, the data do not utilize the specific semantic web features such as, e.g., class and property hierarchy, heterogeneous schema ontologies. The second problem commonly occurring with the datasets is the lack of version consistency. Sometimes, different versions of the same dataset exist, as researchers can introduce minor modifications into a dataset in order to conduct specific experiments and report them. Then, a modified version is re-used for other experiments. In the process, it becomes difficult for the users to track provenance of the datasets reported in different publications, which leads to their results being compared with each other. For instance, the popular Cora dataset exists in at least four different versions[5].

In order to avoid these problems when evaluating data linking techniques, there is a need to create benchmarks which represent realistic matching challenges occurring in linked data sources and to maintain "canonical" versions of benchmark datasets. These requirements were the primary motivations for the instance matching evaluation initiative within the OAEI evaluation campaign.

2.2. Evaluation of ontology matching tools

In the area of ontological schema matching, evaluation efforts have been joined within the Ontology Alignment Evaluation Initiative (OAEI) [10] to produce a comprehensive set of benchmark tests which test different aspects of existing matching tools. Since 2005, several datasets were included into the evaluation campaign in order to evaluate various aspects of the ontology matching task.

Originally, the OAEI evaluation campaign featured a single artificial benchmark (known as the *benchmark* dataset) that includes various tests illustrating different challenges occurring in real-world matching tasks: controlled modifications concern both the level of atomic elements (e.g., random modifications of elements/labels) and the structure level (e.g., deleting/inserting classes in the hierarchy). This serves well the purpose of checking the capabilities of schema matching tools to deal with the presence or absence of different features occurring in the ontologies. However, as evaluation experience has shown [10], this artificial benchmark is less suited for comparing the overall performance of tools: each test focuses on a specific type of situation while not providing a realistic test as a whole. To deal with this problem, the evaluation campaign was extended to include several realistic

[1] http://www.cs.umass.edu/ mccallum/data.html
[2] http://dl.acm.org
[3] http://www.informatik.uni-trier.de/ ley/db/
[4] http://www.cs.utexas.edu/users/ml/riddle/

[5] Experiments with different Cora versions are reported in [9], [32], [33], and [38]

benchmarks involving real-world ontologies covering the same topics. These formed the basis of other benchmarks, in particular:

- *Conference*, which consists of a set of 16 ontologies dedicated to the topic of conference organisation and developed within the OntoFarm project[6].
- *Anatomy*, which includes detailed ontologies describing human and mouse anatomy[7].

Thus, experiences of the ontology matching evaluation led to a conclusion that to achieve effective evaluation of the tools, benchmark tests have to utilise both artificial and real-world datasets.

While some ontologies used in the ontology matching benchmarks contain instance data as well, they are not fully suitable for reuse to evaluate data linking methods due to important differences between these tasks. These differences involve the following aspects.

Larger datasets. One of the main problems in matching instances with respect to the problem of matching ontology concepts is that instance datasets are usually much larger than ontologies in terms of number of entities, properties and values. This puts more emphasis on the problem of matching tool performances and complexity.

Large number of literal data values. Even if literals are often used also in ontologies as labels, comments, and values of property restrictions, in instance datasets the number of properties which have literals as values is usually larger. This puts more emphasis on the problem of having specific data matching functions for strings, dates, numbers.

Identity and similarity: The same-as problem. In ontologies, the semantics of equivalence and subsumption is well defined and can be checked trough standard reasoning techniques. On the contrary, in instance datasets the semantics of same-as relations is ambiguous and, in fact, same-as relations are often used as links with different meanings. This puts emphasis on the problem of providing a formal interpretation of links resulting from the matching process.

Different role of names and property values. Both in ontology and instance matching we deal with labels, used as names for concepts and instances, and with concrete values of properties, used to represent features of concepts and instances. Many matching tools work with names and property values under the assumption that they are useful to determine concepts meaning and real-world objects denoted by instances. However, names and values have a different relevance when dealing with ontologies or instances. In fact, in ontologies, concept names are usually meaningful, although the semantics of concepts depends basically on the mutual concept relations and constraints involving concepts, while concrete values of properties are limited to labels and comments, which are less useful for matching purposes. On the contrary, names of instances are only a limited portion of the information available about instances, since they are associated with several concrete values denoting their features. As an example, we can take the concept *Person* from the DBpedia ontology and the individual *Aristotle* from the DBpedia dataset. *Person* is a subclass of *Agent* and is associated with seven different concrete values of the property *label*. However, all these labels report the term "Person" in different languages, which is only partially useful to determine the concept meaning. On the contrary, *Aristotle* is denoted by many property values, providing information about his birth and death dates and locations, his philosophical interests and works, and so on. Thus, property values have a more relevant role in instance matching than ontology matching, in that we can use some properties to determine the identity of each instance, especially for those properties which contain unique values. This makes instance datasets more similar to relational database records and puts emphasis on the use of record linkage techniques for matching instances.

Different kinds of data heterogeneities. When matching labels and strings, the main problem of ontology matching is due to the fact that labels are used to describe the concept meaning in human understandable terms. Thus, the heterogeneity is mainly due to the fact that often the same label is used to denote different concepts or, on the contrary, that the same concept is labeled with different strings. These are language heterogeneities associated with the use of common words (nouns, verbs, adjectives). This happens also in instance matching, but, in this case, there

[6] http://nb.vse.cz/ svatek/ontofarm.html

[7] http://oaei.ontologymatching.org/2011/anatomy/index.html

are other problems more related to the representation of named entities and enumerated items. In fact, data heterogeneity is often due to errors in data or different conventions about name abbreviations, acronyms and other string/date/number format heterogeneities. This requires more sophisticated techniques for syntactic data value matching. Moreover, in the generic ontology matching, diversity between descriptions of matching concepts and properties can be caused by different factors [23]: in one case, usage of synonyms, in another, rephrasing of descriptions, in the third one, slight semantic difference, etc. In the instance matching tasks, it is more common that the same factor occurs for many pairs of instances and is specific for a particular pair of datasets. Thus, while for an ontology matching tool it is important to be able to apply a wide range of techniques to recognize a mapping between a pair of entities, for an instance matching method the focus is more on discovering the most appropriate set of techniques and applying them in a consistent way. The capability to adapt to specific types of data at hand is thus particularly valuable.

Structural differences between ontology and instances as graphs. Graph-based matching techniques are crucial and much used both in ontology and in instance matching. However, when seen as graphs, there are some differences between ontologies and instance datasets. Usually, the density of instance graphs is higher than ontologies and the average number of other entities connected to a given instance is also higher. This leads to more complex graph matching problems, where the identity of an instance often depends on the meaning of other instances connected with the former one. In addition, while the ontology schema usually represents a single graph, at the data level, information is commonly described using a set of homogeneous subgraphs. This makes it important to be able to extract and analyse relevant subgraphs which carry the instance identity.

Relations between datasets and the real-world. Typically, ontologies are used to represent a shared vocabulary of concepts described by common nouns. On the contrary, instances are usually intended to represent real world objects or digital documents identified by named entities. This implies that thesauri and top ontologies that are useful resources for ontology matching are often less useful for instance matching. On the other side, instance matching may be based on external data sources of interest so that is possible to use existing collection of objects as a reference for the validation of matching results: for example, a publicly available repository such as Wikipedia or Geonames can be used to check whether a particular name describes a single entity (e.g., the city "Dnepropetrovsk") or several entities (e.g., "St Petersburg" which may possibly refer to places in Russia and USA).

Mutual relations between ontology and instance matching. Ontology matching may use instance matching as part of the matching process, in that instances characterize the meaning of concepts: they serve as the main type of evidence for instance-based ontology matching techniques. On the other hand, ontological heterogeneity presents an additional challenge for instance matching, as corresponding sets of instances and properties relevant to the instance identity can be represented in different ways. In these cases, ability to perform partial ontology matching is important for an instance matching tool.

These differences have motivated the need to develop a different set of benchmarks specifically for the task of instance matching in the semantic web domain. For this reason, it was decided to establish the instance matching evaluation as a separate subtrack within the OAEI evaluation campaign, which by now has been performed 3 times (in 2009, 2010, and 2011). In the following sections, we describe the organization of the campaign, the proposed benchmarks as well as the results of the tests.

3. Instance Matching at the Ontology Alignment Evaluation Initiative

The Ontology Alignment Evaluation Initiative (OAEI) [10] aims to evaluate the performance of ontology matching systems on a various number of problems related to ontology matching. The instance matching track was organized over the last three editions of the OAEI.[8] Given the requirements and issues presented in sections 1 and 2, organization of evaluation testing required making decisions at different levels: both technical (choice of benchmark datasets and evaluation metrics) and pragmatic (campaign organization). In this section,

[8] Results of every OAEI campains are available on `http://oaei.ontologymatching.org`.

we describe the general choices we made related to the evaluation campaign as a whole: section 3.1 outlines the main stages of the evaluation camaign, while section 3.2 discusses the choice of evaluation metrics. Then, sections 4 and 5 describe two chosen types of datasets: natural and synthetic.

3.1. Organization

The evaluation is performed as follows:

preparation phase Datasets to be matched and reference alignments are provided in advance. This gives potential participants the occasion to send observations, bug corrections, remarks and other test cases to the organizers. The goal of this preparatory period is to ensure that the delivered tests make sense to the participants. The final test base is then released after a month. The data sets do not evolve after this period.

execution phase During the execution phase, participants use their systems to automatically match the instance data from the test cases. Participants are asked to use one algorithm and the same set of parameters for all tests in all tracks. It is fair to select the set of parameters that provide the best results. Beside parameters, the input of the algorithms must be the two datasets to be matched and any general purpose resource available to everyone, i.e., no resource especially designed for the test. In all cases datasets are serialized in RDF and contain the ontology declarations in some cases. The expected alignments are provided in the Alignment format expressed in RDF/XML [8]. Participants also provided the papers that are published in the Ontology Matching Workshop proceedings and a link to their systems and their configuration parameters.

evaluation phase The organizers evaluate the alignments provided by the participants and return comparisons on these results. In order to ensure that it is possible to process automatically the provided results, the participants are requested to provide (preliminary) results after two months. The standard evaluation measures are precision and recall computed against the reference alignments. For the matter of aggregation of the measures, weighted harmonic means are used (weights being the size of the true positives). This clearly helps in the case of empty alignments. Another technique that has been used is the computation of precision/recall graphs so it was advised that participants provide their results with a weight to each correspondence they found. New measures addressing some limitations of precision and recall have also been used for testing purposes as well as measures compensating for the lack of complete reference alignments.

3.2. Evaluation metrics

In addition to the choice of the test datasets, the evaluation methodology must include the choice of the valid quantitative evaluation measures. A variety of evaluation measures has been used to validate record linkage algorithms:

- Maximum F-Measure: harmonic mean between pairwise precision and recall achieved with the optimal settings of an algorithm [2] [6].
- Pairwise accuracy for the optimal number of pairs [35].
- Percentage of the correct equivalence classes (sets of equivalent instances obtained by computing the transitive closure over obtained mappings) [26].
- Proportions of true matching pairs at different error rate levels [39].
- Precision-recall curves which visualize the algorithm's performance over the whole range of possible threshold values [3].

These similarity measures highlight different performance aspects and also have their advantages and disadvantages. F-Measure combines the precision and recall in one balanced metric, but it does not take account of true negative matches. Pairwise accuracy, on the contrary, counts both correctly identified positive and negative matches. However, the disadvantage of this is that in case of predominance of negative examples in the gold standard (which is normal for an instance matching task) the metric becomes non-discriminative: in this case, a simple matching algorithm which would reject all possible mappings between two datasets would already obtain high accuracy. Counting equivalence classes instead of atomic mappings does not consider errors within each cluster. Error rate thresholds assume a specific matching methodology [14]. Finally,

precision-recall curves can visualize complex behavior patterns of algorithms, but assume that these algorithms use a threshold-based cut-off, which is not always the case.

In the area of ontology matching, novel evaluation metrics have been proposed to improve the standard precision/recall: in particular, in [11] several extended metrics have been proposed, such as symmetric proximity, correction effort, and relaxed precision/recall. These metrics take into account the semantics of mappings: e.g., a class-subclass mapping between two classes which are equivalent in the gold standard will still contribute to the final score, although with a lower weight. This provides flexibility needed to deal with relationships between ontological concepts which are not always clear-cut. Although more appropriate to the schema matching environment, these measures can be useful for instance matching as well, e.g., if the datasets to be matched model instances at different levels of granularity.

Given these pros and cons, for the evaluation of Semantic Web data linking tools we found the precision and recall measures the most informative: given the large volumes of data containing largely distinct individuals, considering correctly identified non-matches measuring the performance of the tools does not add valuable information. To show the balance between these metrics, we used the maximum F-Measure as a single quantitative indicator and precision-recall curves as a more fine-grained illustration means. While more involved metrics such as those defined in [11] were not suitable for the benchmark datasets chosen so far, we study the possibility to use them in the following iterations.

In the rest of the paper we describe the benchmarks we developed for evaluating instance matching tools in this context, which include both real-world (Section 4) and automatically generated (Section 5) datasets, provide an overview of our experience with using these datasets to evaluate proposed tools and discuss the lessons learnt from this experience (Section 6).

4. The real-data benchmark

In this section we describe three benchmarks proposed to OAEI instance matching participants. These benchmarks are based on datasets actually available as web data and describing data used in applications. We also give the evaluation results on each benchmark.

4.1. OAEI 2009: interlinking scientific publications data

Starting from 2009 we proposed a track to OAEI participants focusing on instance matching.

4.1.1. Benchmark

In the 2009 edition the test bed was made of three datasets in the domain of scientific publications.

- AKT EPrints archive, containing information about papers produced within the AKT research project.[9]
- Rexa dataset, extracted from the Rexa search server[10], which was constructed at the University of Massachusetts using automatic information extraction algorithms.
- SWETO-DBLP dataset[11], a publicly available dataset listing publications from the computer science domain.

The SWETO-DBLP dataset was originally represented in RDF. Two other datasets (AKT EPrints and Rexa) were extracted from the HTML sources using specially constructed wrappers and structured according to the SWETO-DBLP ontology. This heterogeneity resulted in many non-trivial cases of data mismatches. Sometimes, the sources contained misrepresentation of data fields, missing values, and even incorrect values (e.g., incorrect publication dates and missing authors).

The ontology describes information about scientific publications and their authors and extends the commonly used FOAF ontology. Authors are represented as individuals of the *foaf:Person* class, and a special class *sweto:Publication* is defined for publications, with two subclasses *sweto:Article* and *sweto:Article_in_Proceedings* for journal and conference publications respectively. The participants were invited to produce alignments for each pair of datasets (AKTRexa, AKTDBLP, and RexaDBLP).

This benchmark presents the advantage to contains three datasets about the same domain. Matching instances sets overlap across the three

[9] http://www.aktors.org/
[10] http://rexa.info/
[11] http://lsdis.cs.uga.edu/projects/semdis/swetodblp/

datasets, giving the possibility to validate some matching results using transitivity. The publications domain is also familiar to the ontology alignment evaluation initiative.

4.1.2. Results

Five systems participated to the evaluation: DSSim [28], RiMOM [27], OKKAM [34], HMatch [5], and ASMOV [21]. In this first instance matching track, 4 systems out of 5 represented generic ontology matching tools, which included instance matching as a part of their functionality, while only one (OKKAM) was specifically aimed at resolving data level coreferences.

Table 1 shows the results.

The AKT/Rexa test scenario was the only one for which the results for ASMOV were available and the only one for which all the systems provided alignments for both *foaf:Person* and *sweto:Publication* classes. OKKAM for the AKT/DBLP test case and RiMOM for the Rexa/DBLP test case only produced alignments for *Publication* instances, which reduced their overall recall. For the class *Publication* the best F-measure in all three cases was achieved by HMatch with RiMOM being the second. OKKAM, which specifically focused on precision, achieved the highest precision in all three cases at the expense of recall. It is interesting to see the difference between systems in the Rexa/DBLP scenario where many distinct individuals had identical titles (e.g., "Editorial.", or "Minitrack Introduction."): this primarily affected the precision in the case of HMatch and RiMOM, but reduced recall for OKKAM.

The performance of all systems was lower for the class *Person* where ambiguous personal names and different label formats reduced the performance of string similarity techniques. The highest F-measure was achieved by RiMOM for the AKT/Rexa scenario and by HMatch for the AKT/DBLP and Rexa/DBLP cases. Again, it is interesting to note the difference between HMatch and OKKAM in the Rexa/DBLP case where the first system focused on F-measure and the second one on precision. This distinction of approaches can be an important criterion when a tool has to be selected for a real world use case: in some cases the cost of an erroneous correspondence is much higher than than the cost of a missed one (e.g., the large-scale entity naming service such as OKKAM) while in other scenarios this might not be true (e.g., assisting the user who performs manual alignment of datasets). In contrast, in the AKT/Rexa scenario the performance of OKKAM was lower than the performance of other systems both in terms of precision and recall. This was caused by different label formats used by AKT and Rexa datasets ("FirstName LastName" vs "LastName, FirstName"), which affected OKKAM most.

In this first evaluation track, the benchmark provided a representative set of realistic challenges occurring in real-world data, which gave interesting insights about the capabilities of different algorithms. However, in itself this benchmark was insufficient, as it did not cover some important aspects. Most notably, these included the large scale and ontological heterogeneity. Two out of three datasets were relatively small scale: this meant that in each pairwise matching task one of the datasets was small, which reduced the complexity of the matching task. Moreover, all three datasets were structured using the same ontology (SWETO) which also simplified the challenges, which the tools had to tackle.

4.2. OAEI 2010: interlinking health-care data

In OAEI 2010, participants were asked to interlink together four datasets, selected for their potential to be interlinked, for the availability of curated interlinks between them, and for their size.

4.2.1. Benchmark

All datasets are on the health-care domain and all of them contain information about drugs (see [22] for more details on the datasets):

dailymed is published by the US National Library of Medecine and has for topic marketed drugs. Dailymed contains information on the chemical structure, mechanism of action, indication, usage, contraindications and adverse reactions for the drugs.

diseasome contains information about 4300 disorders and genes.

drugbank is a repository of more than 5000 drugs approved by the US Federal Drugs Agency. It contains information about chemical, pharmaceutical and pharmacological data along with the drugs data.

sider contains information on marketed drugs and their recorded adverse reactions. It was originally published on flat files before being con-

Table 1: Results of the real-data benchmark subtrack.

Concept	sweto:Publication			foaf:Person			Overall		
System	Prec.	Rec.	FMeas.	Prec.	Rec.	FMeas.	Prec.	Rec.	FMeas.
AKT/REXA									
DSSim	0.15	0.16 0.16		0.81	0.43	0.30	0.60	0.38	0.28
RiMOM	0.96	0.96	0.97	0.93	0.79	0.70	0.94	0.80	0.59
OKKAM	0.99	0.76	0.61	0.73	0.03	0.02	0.94	0.18	0.10
HMatch	0.97	0.93	0.89	0.94	0.56	0.39	0.95	0.62	0.46
ASMOV	0.32	0.46	0.79	0.76	0.37	0.24	0.52	0.39	0.32
AKT/DBLP									
DSSim	0	0	0	0.15	0.19	0.11	0.17	0.13	0.15
RiMOM	1.0	0.84	0.72	0.92	0.79	0.70	0.93	0.73	0.70
OKKAM	0.98	0.88	0.80	0	0	0	0.98	0.28	0.16
HMatch	0.93	0.95	0.97	0.58	0.57	0.57	0.65	0.65	0.65
REXA/DBLP									
DSSim	0	0	0	0	0	0	0	0	0
RiMOM	0.94	0.94	0.95	0.76	0.71	0.66	0.80	0.76	0.72
OKKAM	0.98	0.26	0.15	1.00	0.20	0.11	0.99	0.21	0.12
HMatch	0.45	0.61	0.96	0.40	0.37	0.34	0.42	0.45	0.48

verted as linked-data through a relational database.

These datasets were semi-automatically interlinked using Silk [4] and ODD Linker [18] providing the reference alignments for this task and participants were asked to retrieve these links using an automatic method.

As for the 2009 edition, this benchmark provides four interlinked datasets on a same domain. Reference links were available, provided by the Linked Opened Drug Data working group [22]. On the other hand, datasets are of a relatively small size, thus not evaluating the ability of the systems to scale. The fact that the datasets are all about the same domain is both positive as it allows systems to use links between a pair of datasets to validate (or invalidate) links between another pair of datasets. It is also negative as systems can be tuned to be more efficient for this particular domain, for example using domain specific background knowledge. Another inconvenient of this benchmark lies in the fact the matching task was rather easy. This is due to the domain having strong naming conventions, making that two drugs will very likely have the same name across every drug datasets.

4.2.2. Results

Two systems participated in the data interlinking task: ObjectCoref [18] and RiMOM [27]. Table 3 shows the results.

Table 2: Health-care benchmark composition

Dataset	Dailymed	Diseasome	Drugbank	Sider
Dailymed	0	0	0	1,592
Diseasome	0	0	0	238
Drugbank	0	0	0	283
Sider	1,986	0	1,140	0

The results are very different for the two systems, with ObjectCoref being better in precision and RiMOM being better in recall. A difficult task with interlinking real data is to understand if the results are due to a weakness of the matching system or because links can be not very reliable. In any case, what we can conclude from this experiment with linked data is that a lot of work is still required in three directions: i) providing a reliable mechanism for systems evaluation; ii) improving the performances of matching systems in terms of both precision and recall; iii) work on the scalability of matching techniques in order to make affordable the task of matching large collections of real data. We have thus built the OAEI 2011 instance matching track with these chalenges in mind.

4.3. OAEI 2011: interlinking the New York Times data

While using subsets of publicly available linked data repositories has shown its value, in particular,

Table 3: Results of the real-data benchmark subtrack.

	ObjectCoref		
Dataset	Prec.	FMeas.	Rec.
dailymed	0,55	0,09	0,05
diseasome	0,84	0,10	0,05
drugbank	0,30	0,05	0,03
sider	0,00	NaN	0,00
H-mean	0,50	0,08	0,04

	RiMOM		
Dataset	Prec.	FMeas.	Rec.
dailymed	0,08	0,13	0,30
diseasome	0,08	0,13	0,47
drugbank	0,05	0,08	0,42
sider	0,62	0,53	0,47
H-mean	0,08	0,13	0,35

to illustrate matching challenges occurring in real-world domains, the evaluation procedure in OAEI 2010 also identified some issues. In particular, the topic of datasets was restricted to the medical domain, which biased the overall benchmark towards the specific features of this domain. Second, due to the need to support large-scale matching tasks, the set of gold standard mappings could not be constructed manually, and evaluation had to rely on pre-existing mappings between datasets. These pre-existing mappings were constructed by semi-automated tools, which themselves did not always produce results with 100% quality.

4.3.1. Benchmark

To deal with these issues, the instance matching track in the OAEI 2011 included the set of tests involving the New York Times (NYT) linked data[12]. The NYT repository includes three subsets describing different types of entities mentioned in the New York Times articles: people, organisations, and places. These three subsets were linked to three commonly used semantic web data repositories: DBpedia[13], Freebase[14], and Geonames[15]. These links were provided by the data publishers, which improved the gold standard quality.

[12] `http://data.nytimes.com/`
[13] `http://dbpedia.org`
[14] `http://www.freebase.com/`
[15] `http://www.geonames.org/`

The data in the NYT datasets is structured using the commonly used SKOS vocabulary[16]: individuals are modelled as instances of the class *skos:Concept*, and instance labels use the property *skos:label* rather than generic *rdfs:label*. Other vocabularies are used to represent domain-specific properties, such as number of relevant NYT articles and geo-coordinates (for locations).

While the links provided in the benchmark where mostly correct, it was found they were not complete as some participants found. Also the size of the benchmark is still not large enough to really evaluate the scaling capabilities of participating systems.

4.3.2. Results

In the OAEI 2011 evaluation initiative, test results with the NYT benchmark dataset were produced using three instance matching tools: AgreementMaker, SERIMI, and Zhishi.links [7, 1, 30]. Table 5 shows an overview of the Precision, Recall and F_1-measure results per dataset for these tools, while Figure 1 shows the Precision-Recall graph for their results. In the experiments, Zhishi.links managed to produce high quality mappings consistently over all datasets: it obtained the highest scores on 4 tests out of 7, and the highest average scores. SERIMI performed particularly well on Freebase datasets (it outperformed other tools on 2 tests), while AgreementMaker was particularly successful on linking people from NYT and Freebase.

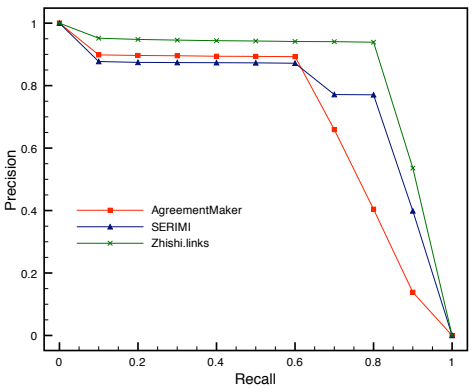

Figure 1: Precision/recall of tools participating in the DI subtrack.

However, the results also highlighted some important issues. The first one is still the quality of the gold standard mappings: despite the fact the

[16] `http://www.w3.org/TR/skos-reference/`

Table 4: The New York Times benchmark composition

Facet	# Concepts	Links to Freebase	Links to DBPedia	Links to Geonames
People	4,979	4,979	4,977	0
Organizations	3,044	3,044	1,965	0
Locations	1,920	1,920	1,920	1,920

Table 5: Results of the real-data benchmark subtrack.

	AgreementMaker			SERIMI			Zhishi.links		
Dataset	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.
DI-nyt-dbpedia-loc.	0,79	0,69	0,61	0,69	0,68	0,67	0,92	0,92	0,91
DI-nyt-dbpedia-org.	0,84	0,74	0,67	0,89	0,88	0,87	0,90	0,91	0,93
DI-nyt-dbpedia-peo.	0,98	0,88	0,80	0,94	0,94	0,94	0,97	0,97	0,97
DI-nyt-freebase-loc.	0,88	0,85	0,81	0,92	0,91	0,90	0,90	0,88	0,86
DI-nyt-freebase-org.	0,87	0,80	0,74	0,92	0,91	0,89	0,89	0,87	0,85
DI-nyt-freebase-peo.	0,97	0,96	0,95	0,93	0,92	0,91	0,93	0,93	0,92
DI-nyt-geonames.	0,90	0,85	0,80	0,79	0,80	0,81	0,94	0,91	0,88
H-mean.	0,92	0,85	0,80	0,89	0,89	0,88	0,93	0,92	0,92

links were checked by the data publisher at the time of their generation, errors in the gold standard still occurred. These errors were caused by several factors: evolution of datasets since the time of interlinking, omitted mappings (false negatives) which are particularly difficult to discover manually, as well as the presence of ambiguous and duplicate instances in target repositories. The second issue concerns the use of specific techniques by the matching systems. In particular, the use of domain-specific knowledge such as common abbreviations. There are different points of view on the use of domain knowledge: for example, it is explicitly forbidden in the OAEI schema matching tests, as the schema matching tools must be able to deal with generic knowledge models. Although this can be seen as too restrictive for instance matching tools, as many cases of heterogeneity in instance matching tasks cannot be resolved without possessing domain knowledge, the cases where some tool is specifically targeted at solving the benchmark tasks have to be prevented to ensure that evaluation results can be generalized. Establishing the rules concerning the use of domain knowledge constitutes an important challenge for the OAEI instance matching track and future instance matching initiatives in general as it influences the quality of comparative evaluation.

5. The automatically generated benchmark

The automatically generated benchmark (called IIMB) is based on the idea of automatically acquiring a potentially large set of data from an existing datasource and to represent data in form of an OWL Abox, serialized in RDF (Either in 2010 and in 2011, the dataset was extracted from Freebase[17]). Then, starting from the initial set of data, we programmatically introduce several kinds of data transformations, with the goal of producing a final set of Aboxes in a controlled way. Participants are then required to match each of the transformed Aboxes against the initial one, trying to find the correct mappings between the original entities and the transformed ones. The main advantage in such an approach is that we have a control over the type and strength of each transformation, which means that it is possible to analytically evaluate the results produced by each tool, by highlighting potential points of strength and weakness of each tool.

5.1. Creation of the benchmark

The benchmark is created using the SWING approach (Semantic Web INstance Generation) [15] a

[17] http://www.freebase.com

disciplined approach to the semi-automatic generation of benchmarks to be used for the evaluation of matching applications. The SWING approach has been implemented as a Java application and it is available at `http://code.google.com/p/swing`.

The SWING approach is articulated in three phases as shown in Figure 2. These phases will be briefly described in the Sections 5.2 - 5.4. These sections are a comprehensive summary of [15].

5.2. Data acquisition techniques.

The SWING data acquisition phase is based on the idea of acquiring data from a linked data repository in a controlled way using a set of predefined queries and then, to enrich the structural and semantic complexity of acquired data from the description logic $\mathcal{ALE}(\mathcal{D})$ up to $\mathcal{ALCHI}(\mathcal{D})$. The main reason of the enrichment step is that linked data are typically featured by a limited level of semantic complexity, while we are interested in providing a benchmark suitable for the evaluation also of logical and reasoning capabilities of the matching tools at hand. In order to briefly summarize the extensions added during the enrichment step, we report the main operations supported by SWING:

- Add super classes and super properties.
- Convert attributes to class assertions.
- Determine disjointness restrictions.
- Enrich with inverse properties.
- Specify domain and range restrictions.

All these operations are semi-automatically performed by focusing on data features like property values and class relations. The benchmark designer can choose which operations have to be applied to data in order to control the semantic complexity of the final Abox.

5.3. Data transformation techniques.

In the subsequent data transformation activity the initial ABox is modified in several ways by generating a set of new ABoxes, called *test cases*. Each test case, is produced by transforming the individual descriptions in the reference ABox in new individual descriptions that are inserted in the test case at hand. In particular, the SWING approach supports the following automatic transformation techniques. In our implementation the evaluation designer has control over these techniques with an easy understandable parameter file.

Deletion/Addition of Individuals.. The SWING approach allows the evaluation designer to select a portion of individuals that must be deleted and/or duplicated in the new ontology. The reason behind this functionality is to obtain a new ontology where each original individual can have none, one, or more matching counterparts. The goal is to add some noise in the expected mappings in such a way that the resulting benchmark contains both test cases where each original instance has only one matching counterpart (i.e., one-to-one mappings) and test cases where each original instance may have more than one matching counterpart (i.e., one-to-many mappings).

Data value transformation. operations work on the concrete values of data properties and their datatypes when available. The output is a new concrete value. In the standard transformation typos are simulated as well as special value transformations for dates, names, gender attributes, and numbers like integers and float. Furthermore, in our synonym transformation we extract synonyms from WordNet (e.g. `Jackson has won multiple awards` is transformed to `Jackson has gained several prizes`).

Table 6: Examples of data transformation operations

Operation	Original value	Transformed value
Standard transformation	Luke Skywalker	L4kd Skiwaldek
Date format	1948-12-21	December 21, 1948
Name format	Samuel L. Jackson	Jackson, S.L.
Gender format	Male	M
Synonyms	Jackson has won multiple awards [...]	Jackson has gained several prizes [...]
Integer	10	110
Float	1.3	1.30

Data structure transformation. operations change the way data values are connected to individuals in the original ontology graph and change the type and number of properties associated with a given individual. A comprehensive example of data structure transformation is shown in Table 7, where an initial set of assertions A is transformed in the corresponding set of assertions A' by applying the property type transformation, property assertion deletion/addition, and property assertion splitting.

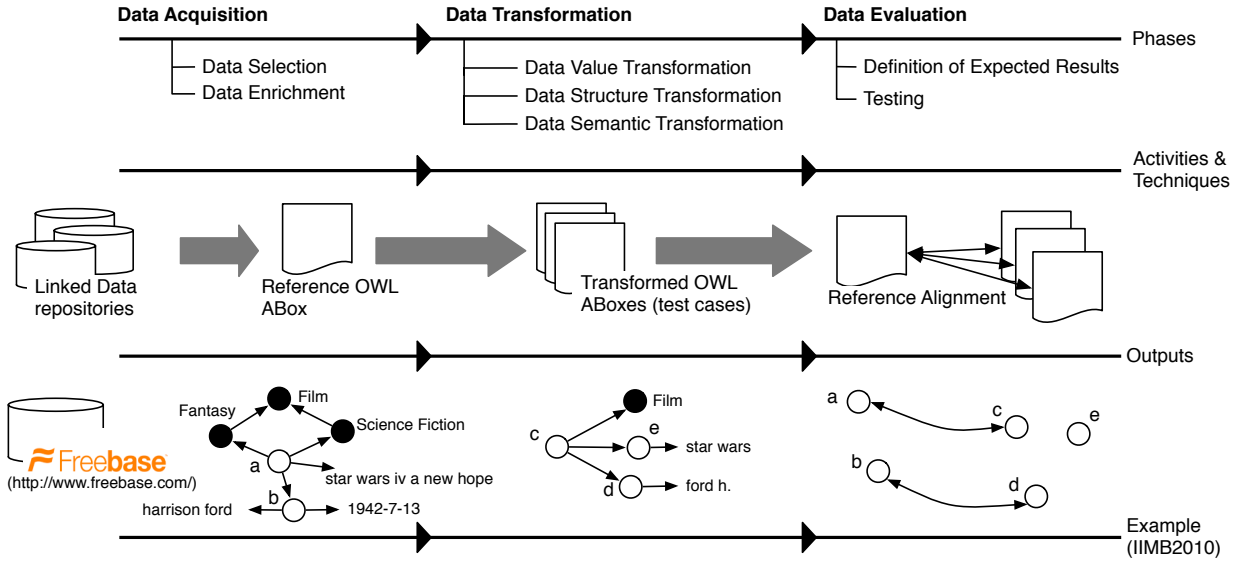

Figure 2: The SWING approach

Table 7: Example of data structure transformations

Original Abox	Transformed Abox
$name(n,$ "Natalie Portman")	$name(n,$ "Natalie")
$born\_in(n, m)$	$name(n,$ "Portman")
$name(m,$ "Jerusalem")	$born\_in(n, m)$
$gender(n,$ "Female")	$name(m,$ "Jerusalem")
$date\_of\_birth(n,$ "1981-06-09")	$name(m,$ "Auckland")
	$obj\_gender(n, y)$
	$has\_value(y,$ "Female")

Table 8: Example of data semantic transformations

Tbox

$Character \sqsubseteq Creature$, $created\_by \equiv creates^-$, $acted\_by \sqsubseteq featuring$, $Creature \sqcap Country \sqsubseteq \bot$

Original Abox	Transformed Abox
$Character(k)$	$Creature(k)$
$Creature(b)$	$Country(b)$
$Creature(r)$	$\top(r)$
$created\_by(k, b)$	$creates(b, k)$
$acted\_by(k, r)$	$featuring(k, r)$
$name(k,$ "Luke Skywalker")	$name(k,$ "Luke Skywalker")
$name(b,$ "George Lucas")	$name(b,$ "George Lucas")
$name(r,$ "Mark Hamill")	$name(r,$ "Mark Hamill")

Data semantic transformation. operations are based on the idea of changing the way individuals are classified and described in the original ontology. For the sake of brevity, we illustrate the main semantic transformation operations by means of the following example, by taking into account the portion of $\mathcal{T}_\mathcal{O}$ and the assertions sets A and A' shown in Table 8.

The challenge introduced by this kind of transformation is twofold: one one side, we expect that the matching tool will be able to conclude that the individual b of the original Abox cannot be the same individual b that appears in the transformed Abox. This task is quite trivial if the matching tools enforces a reasoning process: in fact concepts *Creature* and *Country* are disjoint concepts and, thus, the assertions *Creature(b)* and *Country(b)* are incompatible and this lead to the conclusion that b in A and b in A' must denote different individuals. But, if the matching tool does not enforce reasoning, the many similarities between the two descriptions may lead the tool to the wrong conclusion. On the other side, the two individuals k are expected to be considered as matching. But, again, reasoning is needed to conclude that the two assertions *Character(k)* and *Creature(k)* are fully compatible.

5.4. Data evaluation techniques.

Finally, in the data evaluation activity, we automatically create a ground-truth as a reference alignment for each test case. A reference alignment contains the mappings (in some contexts called "links") between the reference ABox individuals and the corresponding transformed individuals in the test

case. These mappings are what an instance matching application is expected to find between the original ABox and the test case.

5.5. The Benchmarks for the OAEI 2010 and 2011

For the OAEI 2010 campaign two datasets of different size have been used. We provided one small dataset containing about 400 individuals and one larger dataset with about 1400 individuals. In the OAEI 2011 campaign we increased the size of the dataset to 12.333 individuals since we were interested in being more realistic with the size of the benchmark. Table 9 summarizes the different characteristics of the datasets.

Table 9: Characteristics of the automatically generated benchmarks.

	2010 small	2010 large	2011
Individuals	363	1416	12,333
Classes	29	81	163
Object-Properties	32	32	45
Data-Properties	13	13	13
DL-Expressivity	ALCHI(D)	ALCHI(D)	ALCHI(D)

There exist 80 test cases for each of the datasets, divided into 4 sets of 20 test cases each. The first three sets are different implementations of data value, data structure, and data semantic transformations, respectively, while the fourth set is obtained by combining together the three kinds of transformations.

5.6. Results

The IIMB benchmark has been used for evaluation of matching systems in the campains of 2010 and 2011. In 2010 the three Systems ASMOV[20], CODI[31], and RiMOM[37] participated in both datasets, the small and the large version of IIMB.

Figure 3 shows the results of the large version. All the systems obtained very good results when dealing with data value transformations and logical transformations, both in terms of precision and in terms of recall. Instead, in case of structural transformations (e.g., property value deletion of addition, property hierarchy modification) and of the combination of different kinds of transformations we have worse results, especially concerning recall. Looking at the results, it seems that the combination of different kinds of heterogeneity in data descriptions is still an open problem for instance matching systems. When comparing the overall f-measures of the participating systems, all systems are comparable. CODI reached the highest f-measure score of 0.87, RiMOM had 0.84 f-measure, and ASMOV's f-measure was 0.82.

The systems produced almost similar results in the small IIMB benchmark. However, the average f-measure was slightly better for all participating systems in the small benchmark compared to the large one. CODI's f-measure was 0.89, RiMOM reached 0.88, and ASMOV had 0.84 f-measure.

In the 2011 campain we increased the size of the benchmark in order to create a more realistic testing scenario. In the IIMB 2011 dataset it was not suitable any more to calculate the similarity values of every possible indivdual correspondence because this would result in approximative $12,333 \cdot 12,333 = 152,102,889$ similarity comparisons. Unfortunately, only CODI could cope with this large dataset. CODI gained an average f-measure of 0.60. This score is not comparable to the IIMB 2010 benchmarks since the transformation complexity was heavily increased in the 2011 benchmark.

From these two years we can conclude that actual matching systems perform quite well on small to medium sized benchmarks, but have difficulties with large datasets. This, however, is an important requirement in linked open data where a vast amount of instances exist.

6. Current issues and open problems

The benchmarks proposed for instance matching at OAEI so far have been shown to be adequate for the evaluation of instance matching algorithms and tools. This approach has been derived from the work done at OAEI about ontology matching [10]. The main effort of the instance matching track has been devoted in providing specific datasets for instance matching and in introducing the idea of using artificially generated data with the goal of analytically controlling the performances of instance matching tools in different situations. However, there are several open issues about the evaluation of instance matching technologies with respect to ontology matching tools that are still open and that we have learned through our experience at OAEI.

The experience of the first editions of the instance matching track in the context of OAEI has provided materials, suggestions, and tools for improving the contest and trying to address the main issues still

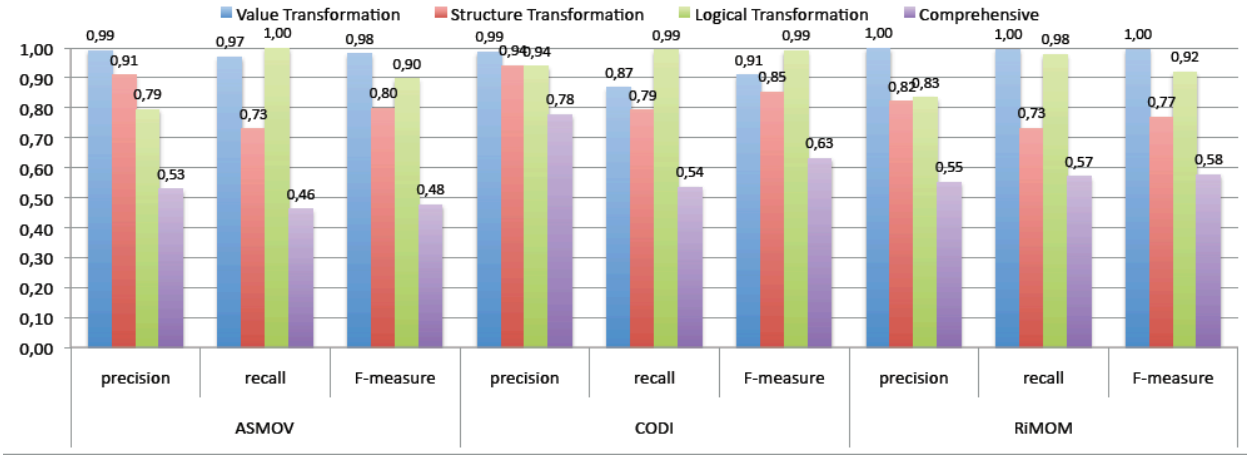

Figure 3: Results of IIMB 2010, large version

open in future editions. In particular, some of the open issues discussed in the previous section have been already partially addressed, while others require new actions or new tools. In Table 10, we summarize what we already have and what is still required in the future.

More in detail, possible future actions for IM@OAEI are the following:

Larger datasets. We already have large datasets, especially for what concerns the real-data benchmark. However, the contest has been focused in the previous editions only on precision and recall. In future editions, we will focus more on time performances of matching tools, trying to understand the scalability of matching techniques and the capability of matching tools to incrementally evaluate matching when the number of entities in a dataset grows. To this end, we will provide datasets with a different number of entities, relations, and properties, in order to put in relation tools time efficiency with the number of elements involved in the matching process.

Identity and similarity. One of the main problems in evaluating the capability of matching tools to identify different kind of "same-as" relations between instances is that a reference set of expected relations among data is usually missing. In order to address this problem, one of the possible solutions is to implement in the artificially generated benchmark tool a functionality for generating an expected set of mappings including several kinds of possible relations, ranging from strict identity to simple similarity. As an example, we could think to an instance I featured by two properties: the first property is sufficient to identify the object represented by the instance (i.e., an ID code), while the second property as a more generic value (e.g., a date or the gender). In such a case, we will generate two different instances, I' and I''. The first instance I' is featured by both the properties with values transformed by means of one of the string/date transformation functions already available in our transformation tools. In generating the second instance I'' we apply transformations functions but we also delete the ID property. Then, we generate two mappings for I. The first mapping $I \leftrightarrow I'$ denotes the identity between I and I', since both the instance refer to the same real-world object. The second mapping $I \leftrightarrow I''$ denotes a generic similarity between the two instances, since we do not have enough information to conclude the identity of I''. Finally, participants will be required to discover both the mappings with the correct meaning. In particular, in order to avoid obliging the tools to produce a non standard output, we could envisage three different solutions: in the first one, tools are requested to exploit the property "relations" which is foreseen in the RDF alignment mapping format[18] in order to state the difference between identity and similarity mappings. The second solution is based on the idea of considering similarity as a sort of "identity degree". In this case, we stress the role of

[18] http://alignapi.gforge.inria.fr/format.html.

Table 10: Current and future actions with respect to the open issues in the evaluation of instance matching tools

Issue	**Current actions / material**	**Future directions**
Larger datasets	Real-data benchmark	More focus on time performances of matching tools
Identity and similarity	Artificially generated benchmark	Generate different kinds of possible mappings according to different definitions of "same-as". Participants should be able to discriminate among the different mappings
Different role of names and property values	Artificially generated benchmark	Generate new transformations on the basis of the property values distribution
Different kinds of data heterogeneities	Artificially generated benchmark	Improve the transformations functions including more standard transformations (e.g., acronyms)
Structural differences between ontology and instances as graphs	Artificially generated benchmark	Add the graph density as a parameter for transformation functions
Relations between datasets and the real-world	Real-data benchmark	Include the usage of external sources as a reference for matching tools
Mutual relations between ontology and instance matching	Artificially generated benchmark	Improve transformations based on the logical structure of the reference Tbox

weights associated with mappings: the higher the weight, the higher the similarity relation. In this approach, identity is represented as a similarity relation associated with a weight equal to 1.0. Finally, a third solution is to run two different tests, one for identity and the other for similarity. This last solution is preferable, because we give participants the opportunity to change their tools configuration parameters according to the task at hand.

Different role of names and property values. This issue is already partially addressed in the artificially generated benchmark. In fact, we are already able to generate new property values by applying arbitrary transformation functions. However, the transformation process does not take into account the value distribution of each property. In other terms, if a property has a limited number of different values for many individuals, such as a property *genre*, which has only two possible values (i.e., Male and Female) for all the individuals, it is transformed by applying the same rules used for properties which have many different values for different individuals, such as a property *name*. Our plan is to include such a parameter in order to make it possible for the benchmark designer to choose if she wants to apply transformations only to properties featured by a limited range of possible values (i.e., no identifying properties) and/or also to properties with many possible values (i.e., highly identifying properties). The idea here is that this second kind of properties are more crucial in order to determine the identity of individuals and their mappings.

Different kinds of data heterogeneities. Also this issue is already partially addressed in the artificially generated benchmark. However, the majority of string transformation functions used in the transformation tool are based on random string transformations. In order to make transformations more realistic, we will implement more transformation functions for some kinds of standard data formats, such as acronyms.

Structural differences between ontology and instance as graphs. Actually, in the automatically generated benchmark it is possible to transform a property which has another instance as value into a property with a concrete value, and vice versa. As an example, suppose to have an instance representing a movie, featured by a property *director* having as value another instance D representing the director John Smith. D is featured by a property *name*, having the string "John Smith" as value. In our

benchmark, it is possible to transform the movie instance by deleting the instance D and changing the property *director* which will have the string "John Smith" as value. On the contrary, it is also possible to transform a string into a new individual. This benchmark generation capability helps the designer in controlling the density of the resulting graph of the test cases. However, this procedure is not iterable to obtain large graphs with a very high number of individuals and properties. In order to increment the complexity of the graphs resulting from transformations, we will add a parameter which will make it possible to control the number of individuals and new properties added to the graph by adopting this specific transformation.

Relations between datasets and the real-world. Our real-data benchmark already contains real-world data. However, till now, we have not explored the role played by external data sources of referent on the instance matching process. A possible approach to that end is to provide a specific subset of data which includes additional data to be used as a support in correctly determining the mappings between elements. Then, we will require participants to execute a first run of matching using only the dataset without the additional data, which will instead be available only for a second run. The idea is to compare evaluation results after the first run agains evaluation results after the second run, in order to observe the impact of using additional data in the matching process.

Mutual relations between ontology and instance matching. One section of test-cases produced in the artificially generated benchmark is devoted to transformations which are based on the Tbox structure that is transformed as well. About this issue, we plan to improve the transformations based on the logical structure of the original dataset, by including new ontological transformations by reducing at the same time the information provided by property values, especially for what concerns concrete values. In such a way, the information about instances derived from the Tbox constraints will become more crucial to the end of finding the correct mappings. This approach has the goal of determining if and how much matching tools are capable of exploiting the ontology during the instance matching process.

7. Concluding remarks

In this paper, we have presented the experience of IM@OAEI, an initiative to promote the evaluation of instance matching and data linking techniques and tools, in the context of OAEI, the Ontology Alignment Evaluation Initiative. In particular, we have presented our approach that is based on the idea of combining real-data and automatic generated data for the evaluation in order to provide on one side a realistic context for instance matching tools and, on the other side, a framework where we can reproduce different causes of data heterogeneity in order to analytically and programmatically verify the points of strength and weakness of each evaluated tool. Our future work in the next editions of IM@OAEI will be devoted to the study of new measures for the evaluation besides the classical precision and recall as well as on the improvement of our benchmarks with the goal of evaluating the behavior of the instance matching tools with respect to some crucial open problems in the field, such as the semantics of instance mappings and the efficiency of matching tools when dealing with large collections of data.

References

[1] Samur Araujo, Arjen de Vries, and Daniel Schwabe. Serimi results for OAEI 2011. In *CEUR-WS Vol-814*, 2011.

[2] Mikhail Bilenko and Raymond J. Mooney. Adaptive duplicate detection using learnable string similarity measures. In *9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2003)*, pages 39–48, Washington DC, 2003.

[3] Mikhail Bilenko and Raymond J. Mooney. On evaluation and training set construction for duplicate detection. In *ACM SIGKDD-03 Workshop on Data Cleaning, Record Linkage, and Object Consolidation*, Washington DC, 2003.

[4] Christian Bizer, Julius Volz, Georgi Kobilarov, and Martin Gaedke. Silk - a link discovery framework for the web of data. In *18th International World Wide Web Conference*, April 2009.

[5] Silvana Castano, Alfio Ferrara, Davide Lorusso, Tobias Henrik Näth, and Ralf Möller. Mapping validation by probabilistic reasoning. In *Proceedings of the 5th European semantic web conference on The semantic web: research and applications*, ESWC'08, pages 170–184, Berlin, Heidelberg, 2008. Springer-Verlag.

[6] William W. Cohen and Jacob Richman. Learning to match and cluster large high-dimensional data sets for data integration. In *8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2002)*, 2002.

[7] Isabel F. Cruz, Cosmin Stroe, Federico Caimi, Alessio Fabiani, Catia Pesquita, Francisco M. Couto, and Matteo Palmonari. Using agreementmaker to align ontologies for OAEI 2011. In *CEUR-WS Vol-814*, 2011.

[8] Jerome David, Jerome Euzenat, Francois Scharffe, and Cassia Trojahn dos Santos. The alignment api 4.0. *Semantic web journal*, 2(1):3–10, 2011.

[9] Xin Dong, Alon Halevy, and Jayant Madhavan. Reference reconciliation in complex information spaces. In *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 85–96, New York, NY, USA, 2005. ACM.

[10] Jérôme Euzenat, Christian Meilicke, Heiner Stuckenschmidt, Pavel Shvaiko, and Cassia Trojahn. Ontology alignment evaluation initiative: Six years of experience. *Journal of Data Semantics*, XV:158–192, 2011.

[11] Marc Ehrig, Jérôme Euzenat. Relaxed precision and recall for ontology matching. In *Proc. of the K-Cap Workshop on Integrating Ontologies*, pages 2532, Banff (Canada), 2005.

[12] Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, Vassilios S. Verykios. Duplicate Record Detection: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 19(1): 1–16, 2007.

[13] Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer-Verlag, Heidelberg (DE), 2007.

[14] Ivan P. Fellegi and Alan B. Sunter. A theory for record linkage. *Journal of the American Statistical Association*, 64(328):1183–1210, 1969.

[15] A. Ferrara, S. Montanelli, J. Noessner, and H. Stuckenschmidt. Benchmarking matching applications on the semantic web. In *The Semanic Web: Research and Applications (ESWC 2011)*, Lecture Notes in Computer Science Volume 6644, pages 108–122, 2011.

[16] A. Ferrara, A. Nikolov, F. Scharffe. Data Linking for the Semantic Web. *International Journal of Semantic Web and Information Systems*, 7(3):46–76, 2011.

[17] S. Castano, A. Ferrara, S. Montanelli, G. Varese. Ontology and Instance Matching. In *Knowledge-Driven Multimedia Information Extraction and Ontology Evolution*, Springer Berlin / Heidelberg, 2011.

[18] Oktie Hassanzadeh, Lipyeow Lim, Anastasios Kementsietsidis, and Min Wang. A declarative framework for semantic link discovery over relational data. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 1101–1102, New York, NY, USA, 2009. ACM.

[19] Robert Isele and Christian Bizer. Learning linkage rules using genetic programming. In *The 6th International Workshop on Ontology Matching (OM 2011), 10th International Semantic Web Conference (ISWC 2011)*, Bonn, Germany, 2011.

[20] Y.R. Jean-Mary, E.P. Shironoshita, and M.R. Kabuka. Asmov: Results for oaei 2010. *Ontology Matching*, 126, 2010.

[21] Yves R. Jean-Mary, E. Patrick Shironoshita, and Mansur R. Kabuka. Ontology matching with semantic verification. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7, 2009.

[22] Anja Jentzsch, Jun Zhao, Oktie Hassanzadeh, Kei-Hoi Cheung, Matthias Samwald, and Bo Andersson. Linking open drug data. In *Linking Open Data Triplification Challenge at the I-Semantics*, 2009.

[23] Michel Klein. *Change Management for Distributed Ontologies*. PhD thesis, Vrije Universiteit Amsterdam, 2004.

[24] Hanna Köpcke, Andreas Thor, and Edhard Rahm. Comparative evaluation of entity resolution approaches with fever. In *35th Intl. Conference on Very Large Databases (VLDB)*, Lyon, France, 2009.

[25] Hanna Köpcke, Erhard Rahm. Frameworks for entity matching: A comparison. *Data & Knowledge Engineering*, 69(2):197–210, 2010.

[26] Steve Lawrence, C. Lee Giles, and Kurt D. Bollacker. Autonomous citation matching. In *3rd International Conference on Autonomous Agents*, New York, USA, 1999.

[27] Juanzi Li, Jie Tang, Yi Li, and Qiong Luo. RiMOM: A dynamic multistrategy ontology alignment framework. *IEEE Transactions on Knowledge and Data Engineering*, 21(8):1218–1232, 2009.

[28] M. Nagy, M. Vargas-Vera, and P. Stolarski. Dssim results for oaei 2009. In *The 4th Workshop on Ontology Matching (OM 2009), 8th International Semantic Web Conference (ISWC 2009)*, 2009.

[29] Andriy Nikolov, Victoria Uren, Enrico Motta, and Anne de Roeck. Refining instance coreferencing results using belief propagation. In *3rd Asian Semantic Web Conference (ASWC 2008)*, pages 405–419, Bangkok, Thailand, 2008.

[30] Xing Niu, Shu Rong, Yunlong Zhang, and Haofen Wang. Zhishi.links results for OAEI 2011. In *CEUR-WS Vol-814*, 2011.

[31] J. Noessner and M. Niepert. Codi: Combinatorial optimization for data integration–results for oaei 2010. *Ontology Matching*, page 142, 2010.

[32] Parag and Pedro Domingos. Multi-relational record linkage. In *KDD Workshop on Multi-Relational Data Mining*, pages 31–48, Seattle, CA, USA, 2004. ACM Press.

[33] Steffen Rendle and Lars Schmidt-Thieme. Object identification with constraints. In *6th IEEE International Conference on Data Mining (ICDM '06)*, 2006.

[34] Heiko Stoermer, Nataliya Rassadko, and Nachiket Vaidya. Feature-based entity matching: The FBEM model, implementation, evaluation. In *Proceedings CAISE 2010*, pages 180–193, 2010.

[35] Sheila Tejada, Craig A. Knoblock, and Steven Minton. Learning object identification rules for information integration. *Information Systems Journal*, 26:635–656, 2001.

[36] Andreas Thor and Erhard Rahm. MOMA - a mapping-based object matching system. In *3rd Biennial Conference on Innovative Data Systems Research*, Asilomar, CA, USA, 2007.

[37] Z. Wang, X. Zhang, L. Hou, Y. Zhao, J. Li, Y. Qi, and J. Tang. Rimom results for oaei 2010. *Ontology Matching*, 195, 2010.

[38] Melanie Weis and Felix Naumann. Relationship-based duplicate detection. Technical Report HU-IB-206, Humboldt University of Berlin, 2006.

[39] William E. Winkler. Methods for record linkage and bayesian networks. Technical report, US Bureau of the Census, Washington, DC 20233, 2002.