Instance Matching Benchmarks in the Era of Linked Data

Evangelia Daskalaki, Giorgos Flouris, Irini Fundulaki, Tzanina Saveta
FORTH-ICS
{eva,fgeo,fundul,jsaveta}@ics.forth.gr

Abstract The goal of this survey is to present the state of the art instance matching benchmarks for Linked Data. We introduce the principles of benchmark design for instance matching systems, discuss the dimensions and characteristics of an instance matching benchmark, provide a comprehensive overview of existing benchmarks, as well as benchmark generators, discuss their advantages and disadvantages, as well as the research directions that should be exploited for the creation of novel benchmarks, to answer the needs of the Linked Data paradigm.

Keywords Benchmarking, Instance Matching, Ontology Matching, Semantic Web Data, Ontologies, Linked Data.

1 Introduction

The number of datasets published in the Web of Data as part of the Linked Data Cloud is constantly increasing. The Linked Data paradigm is based on the unconstrained publication of information by different publishers, and the interlinking of Web resources; the latter includes "same-as" interlinking, i.e., the identification of resources described in different datasets that correspond to the same real-world entity. In most cases, the latter type of identification is not explicit in the dataset and must be automatically determined using *instance matching* tools (also known as record linkage [29], duplicate detection [6], entity resolution [4, 56, 75, 76], deduplication [57], merge-purge [59], entity-identification [58], object identification [35], and data fusion [70]).

For example, searching into the Geonames[1] dataset for the resource "Athens" would return the city of Athens in Greece, accompanied with a map of the area and information about the place. Additional information about the same place can be found also in other datasets, for instance in DBpedia[2]; exploiting both information sources requires the identification that these two different web resources (coming from different datasets) correspond to the same real-world entity.

There are various reasons why the same real-world entity is described in different sources. For instance, as mentioned above, in open and social data, anyone is an autonomous data publicist, and simply chooses his preferred representation or the one that best fits his application. Further differences may be due to different data acquisition approaches such as the processing of scientific data. In addition, entities may evolve and change over time, and sources need to keep track of these developments, which is often either not possible or very difficult (especially when this happens in a synchronous way). Finally, when integrating data from multiple sources, the process itself may add (new) erroneous data. Clearly, these reasons are not limited to problems that did arise in the era of Web of Data, it is thus not surprising that instance matching systems have been around for several years [6][13].

The large variety of instance matching techniques requires their comparative evaluation to determine which one is best suited for a given context. Performing such an assessment generally requires well-defined and widely accepted benchmarks to determine the weak and strong points of the proposed techniques and/or tools. Furthermore, such benchmarks typically motivate the development of more performant systems in order to overcome identified weak points. Therefore, well-defined benchmarks help push the limits of existing systems, advancing both research and technology.

[1] Geonames http://www.geonames.org/

[2] DBpedia http://dbpedia.org/

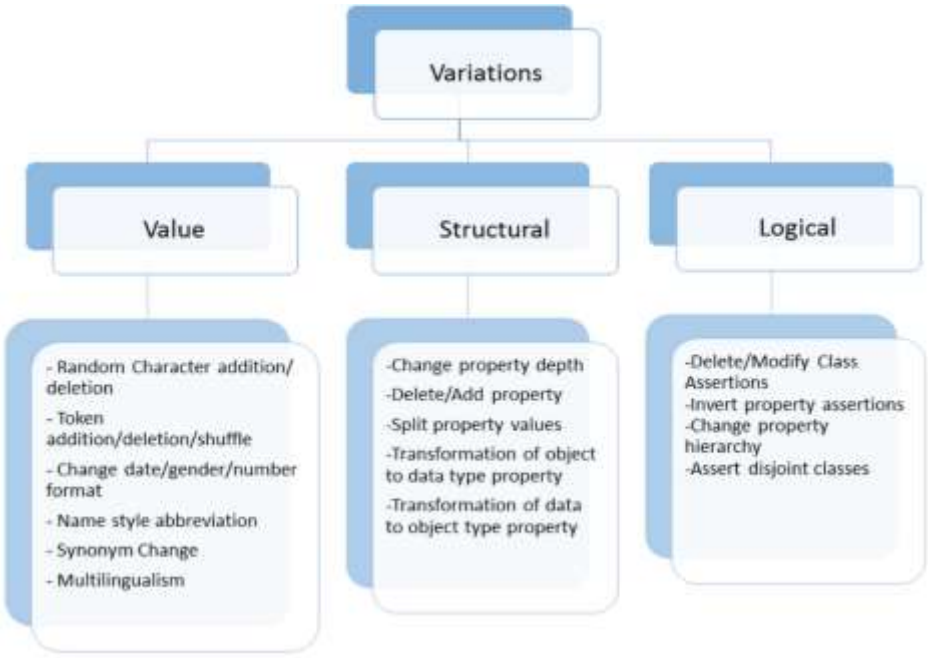


Fig. 1 Linked Data Variations Classification

A benchmark is, generally speaking, a set of tests against which the performance (quality of output, efficiency, effectiveness) of a system is measured.

This survey aims to assess the current state of the art instance matching benchmarks for Linked Data. In particular, we start by explaining why we choose to study the problem of instance matching benchmarks for Linked Data (section 2). In section 3, we describe the characteristics, objectives and main components of an instance matching benchmark. Then, we present benchmark generators for Linked Data (section 4). In section 5 we analyse the most important instance matching benchmarks that have been proposed in the literature; the presentation gives particular focus in comparing the characteristics of the different benchmarks and explaining their advantages and drawbacks. This analysis is used in section 6 to provide guidelines for selecting the proper benchmark for the different contexts and to propose interesting types of benchmarks that could be developed in the future.

Given the increasing importance of instance matching for Linked Data and the plethora of available tools for the task, we believe that a survey on benchmarks for such tools is timely in order to raise awareness on the different existing instance matching evaluation methodologies. To the best of our knowledge, this is the first survey of benchmarks for instance matching tools for Linked Data.

2 Setting the Scope

The instance matching problem has been considered for more than half a decade in Computer Science [7] and has been mostly considered for relational data. There has been significant work on instance matching techniques for relational data [29, 60, 54]. In this context the problem is well-defined: the data is well structured and the focus of the approaches was on discovering differences between values of relation attributes (i.e., value variations). Consequently, the proposed solutions did not have to focus on variations in structure or semantics but simply focus on value variations. In addition, the data processed by the proposed algorithms is dense and usually originated from a very limited number of, well curated, sources.

The first approaches for instance matching for general Web of Data addressed the problem for XML data [63]. In principal, XML data may exhibit strong structural variations (as no schema is necessarily imposed), however, solutions proposed for XML have typically assumed that the data conform to the same schema (i.e., data from different schemata need to be mapped to a common schema before performing instance

matching) [63]. Thus, the structural variations between instances are limited to the instance level (e.g., number of occurrences, optional elements, etc.) and not at the schema level. Finally, the proposed methods focus on data that are typically dense.

In the era of Linked Data the picture is different. Linked Data are described by expressive schemas that carry rich semantics expressed in terms of the RDF Schema Language (RDFS) and the OWL Web Ontology Language. RDFS and OWL vocabularies are used by nearly all data sources in the LOD[3] cloud. According to a recent study[4], 36.49% of LOD use various fragments of OWL so it is imperative that we consider the constraints expressed in such schemas when developing instance matching tools and benchmarks. Consequently, the variations in the huge number of data sources are value, structural as well as logical [61]. As far as semantics are concerned, when paired with a suitable reasoning engine, Linked Data allow implicit relationships to be inferred from the data [61], which was not possible with relational data and XML data.

Due to these reasons, instance matching systems that have been designed for relational or XML data cannot fully exploit the aforementioned heterogeneities and thus failed to deliver good matching results.

Furthermore, according to [70], there exist specific requirements that distinguish the Linked Data from other instance matching workloads, which arise from the autonomy of data sources and the uncertainty of quality-related meta-information. Thus, it is required to assess data quality in order to resolve inconsistencies.

This survey aims at *describing the current state of the art in instance matching benchmarks, with particular focus on the case of Linked Data*, which, as explained above, present differences in the nature of the data (values and structure), but also in the semantic load they carry.

[3] https://www.w3.org/DesignIssues/LinkedData.html

[4] http://linkeddatacatalog.dws.informatik.uni-mannheim.de/state/

3 Instance Matching Benchmarks

A benchmark is a set of tests against which the performance (quality of output, efficiency, effectiveness) of a system is measured. Benchmarking, from a philosophical point of view, is "the practice of being humble enough to admit that someone else is better at something and wise enough to try to learn how to match and even surpass them at it" [2]. The underlying meaning of the above quotation is that it is certainly not easy to be the best, but what matters most is trying to become the best and this can only be done through assessment and identification of weak points, which can be worked upon and improved. So, benchmarking aims at providing an objective basis for assessments. In this way, benchmarks help computer systems to compare, to assess their performances, and last but not least, to push systems to get further. Due to the fact that the performance of the systems varies enormously from one application domain to another [17], there does not exist a single benchmark that can measure the performance of computer systems on all contexts. Thus, it is essential to have domain-specific benchmarks that specify a typical workload for the corresponding domain; this survey focuses on IM benchmarks for Linked Data, for the reasons that have been explained in section 2.

The results of various systems on a benchmark gives a rough estimate of their performance. However, such estimates are always relative, i.e., in relation to the results of other systems for the same benchmark [17]. Along these lines, if a system shows good results for one specific benchmark, we could conclude that the system can handle very well the workload of the benchmark, but we cannot easily come to a conclusion about the benchmark itself (e.g., how well it addresses the challenges it sets) . If though the majority of systems provide good results, we can say that the benchmark addresses trivial cases.

In order for the systems to be able to use the benchmarks and report reliable results, the benchmarks must have specific characteristics. First of all, they have to be *open and accessible* for all interested parties, so that the results can be compared to each other. Open means that they are

free to use and accessible means that they are easily available to the interested parties.

Moreover, benchmarks also have to be *persistent*. By this we mean that the components of one benchmark should not evolve or change through time, so as to make the results of different systems (obtained at different times) comparable.

Note that this requirement rules out testing datasets (sometimes called "benchmarks") which are based on datasets obtained from the Web (without being versioned). For example in [64], [66], [55], and [71] authors use very well-known datasets like LinkedMDB [67] and DBpedia to evaluate their well-known systems, but since these datasets evolve through time, it is very difficult to reproduce the exact same test-sets, and thus run the same tests again. Due to these reasons these testing datasets are not considered as benchmarks.

Finally, benchmarks have to be *unbiased*, which means that there should not exist any conflict of interest between the creators of the benchmark and the creators of the system under test. Up until now, due to the lack of standard benchmarks, systems have created their own synthetic testing datasets (called "benchmarks") and reported results on them [12]. This created the problems that systems in test, sometimes unintentionally, tested only the strong points of their systems with those testing datasets. In this survey, we only consider benchmarks that are open, accessible, persistent, and unbiased.

An instance matching benchmark consists of different test-cases, where each test case addresses a different kind of requirement. A test case comprises of the source and target datasets, the reference alignment (or gold standard or ground truth) which contains the correct alignments between the source and the target dataset, and the evaluation metrics.

In the following subsections we give more details on each of these components (subsections 3.1-3.3), and provide also a set of evaluation metrics for assessing the quality of benchmarks (subsection 3.4).

3.1 Datasets

Datasets are the raw material of a benchmark. In the case of instance matching benchmarks, each test case contains a source dataset and a target dataset, which are compared to identify "matched" instances. Datasets are characterized by:

- their nature (*real* or *synthetic*),
- the *schemas/ontologies* they use,
- their *domains*,
- the *variations* appearing in the dataset.

Nature of Datasets

As far as the nature of datasets is concerned, they are divided into real and synthetic.

Real datasets are those that contain real data and can be used either as a whole (i.e., the entire ontology with its instances) or partly (i.e., only a part of the ontology and the corresponding data instances, without any modifications). Real datasets are widely used in benchmarks since they offer (a) realistic conditions for addressing heterogeneity issues, and (b) they have distributions that reflect real world situations. The disadvantage of using real datasets in benchmarks is that they often do not come with adequate reference alignments, so a manual or semi-automatic process of reference alignment creation must be undertaken, which often leads to incomplete or incorrect reference alignments.

Synthetic datasets are generated using automated data generators (please refer to section 4 for existing Linked Data Benchmark generators). Such generators may be used to create a dataset from scratch (i.e., create new data based on a schema), or to create a dataset based on an existing datasets (multiply the existing instances).

Those generators try to overcome the problem of unrealistic data distributions during synthetic data generation by starting from a real dataset (which is either pre-selected), and creating a synthetic dataset by applying "transformations" over the source (real) dataset (see section 4). This way, the generated (synthetic) dataset is closely based on the actual data, and thus can be claimed to enjoy realistic distributions.

Synthetic datasets are useful because: (a) they offer fully controlled test conditions; (b) they

have accurate reference alignments (as they are synthetically generated); and, (c) they allow setting the focus on specific types of heterogeneity problems in a systematic manner. These characteristics are playing a crucial role when the scope of the benchmark is to test specific features of the systems. The disadvantage of synthetic datasets is that if the generation is not done by considering characteristic of real data, then the datasets produced usually come with unrealistic distributions.

Benchmark Schemas & Domains
Datasets are also characterized by their schema. Datasets can follow the same schema or different ones. In the former case, matching is only done at the instance level, whereas in the latter, schema mapping might also be conducted and the resulting mappings take part in the instance matching process from the systems under test.

Furthermore, datasets can belong to the *same* or to *different* (but overlapping) domains. In the latter case, matches will exist in the overlapping parts of the datasets/domains.

Benchmark Variations
Last, but not least, datasets are characterized by the different types of *variations* they consider. Variations correspond to the differences that exist (or are applied, in the case of synthetic datasets) between the matched instances of the source and target datasets. Datasets (and benchmarks) may contain different kinds of variations. According to Ferrara et al. [11, 12], three kinds of variations exist for Linked Data, namely *value variations*, *structural variations* and *logical variations*.

Value variations address differences in the data level, which can occur due to typographical errors, differences in the format (e.g., different date and number formats), name style abbreviations, etc. Multilingualism is also considered a type of value variation. This is a useful feature, as the Linked Data cloud is multilingual by nature.

Structural variations refer to differences in the structure of the Linked Data schemas. These variations can occur, e.g., because of the deletion of properties, or by splitting two properties, or by the transformations of datatype properties to object properties, etc.

Logical variations come from the semantically rich constructs of the RDF/OWL languages[5] [6]. Such variations can be quite complex and are mostly related to the interplay between the schema and instance level, where the instance matching systems can get hints for or against the match of two Web resources by applying reasoners and by using schema information. An example of such a variation appears when an instance is modified in a way that is classified under disjoint classes in the source and target schema(s). As mentioned also in section 2, these kinds of variations are only applicable in Linked Data constructed in RDF/OWL languages, due to the rich semantics of the languages.

Fig. 1 shows the types of variations that can occur in both real and synthetic datasets. The common case in real benchmarks is that the datasets to be matched contain different kinds (combinations) of variations. On the other hand, synthetic datasets may be purposefully designed to contain specific types (or combinations) of variations (e.g., only structural or a combination of value, structural and logical ones), or may be more general, in an effort to illustrate all the common cases of discrepancies that appear in reality between individual descriptions.

3.2 Reference Alignment

A *reference alignment* (known also as ground truth or gold standard) is considered as the "correct answer sheet" of the benchmark, and is used to judge the completeness and soundness of the matched instances produced by the benchmarked matching algorithms. For benchmarks employing synthetic datasets, the reference alignment is always automatically generated, as the errors (variations) that are added into the datasets are known and systematically created. When it comes to real datasets, the reference alignment can be either manually curated, or (semi-) automatically generated. In the first case, domain experts manually find the matches between the datasets, whereas in the second, supervised systems and crowdsourcing techniques can be used in finding the matches.

[5] http://www.w3.org/TR/rdf-schema/

[6] http://www.w3.org/TR/owl-semantics/

This process is often time consuming and error prone.

Note that in the Linked Data setting, it might also be the case that a reference alignment already exists in the dataset in the form of "same-as" links. In this case a possible problem is incompleteness, as it is not known whether the same-as links connect all the same entities that exist between two datasets, or only a portion of them.

It is important to add at this point that all instance matching benchmarks should have a reference alignment; a set of datasets without a reference alignment, cannot be reliably used to compare instance matching systems, and, thus, datasets are not considered a benchmark for the purposes of this survey.

3.3 Evaluation Metrics for Systems under Test

The *evaluation metrics* for systems under test are used to determine and assess the systems behaviour and performance. Various metrics have been proposed for different benchmarks. The most widely known benchmarks are those proposed by the Transaction Processing Performance Council (TPC)[7]. TPC is a non-profit corporation founded to define transaction processing and database benchmarks and to disseminate objective, verifiable TPC performance data to the industry. The metrics TPC benchmarks employ, are related to performance and efficiency (e.g. throughput – source rows processed per second, transaction rate, transactions per second, Query-per-Hour Performance, etc.) [31]. Even though, performance is indisputably important in instance matching benchmarks, what matters most is the quality of the instance matching systems results, i.e., returning the correct answers without any errors. This is because instance matching is (a) mostly an offline process and (b) an imperfect process by definition, where false positive and false negative matches can (and usually do) occur.

For this reason, the evaluation metrics that are dominantly employed for instance matching benchmarks are the standard *Precision*, *Recall* and *F-measure* [15]. Precision is the fraction between the correct answers of the tested system and the total number of answers in its result set. Recall is the fraction between the correct answers of the tested system and the total number of correct answers reported in the reference alignment. Thus, precision is more about how many errors are contained into the dataset (false positive answers), while recall counts how many correct answers are missing (false negative answers). F-measure is the harmonic mean of recall and precision.

3.4 Evaluation Criteria for Benchmarks

The aim of this survey is not only to present, but also to compare and evaluate the different state of the art benchmarks for instance matching systems for Linked Data. To perform this comparison, we will rely on both qualitative and quantitative benchmark evaluation criteria.

The qualitative evaluation criteria are the following:

- **Systematic Procedure**: This criterion assesses whether the matching tasks required by the benchmark are reproducible and whether their execution is comparable.
- **Quality**: Used to judge if the benchmark offers precise evaluation rules and high quality ontologies.
- **Equity**: Assesses the equal treatment of the systems and their results during the evaluation process.
- **Reference Alignment**: This criterion assesses the accuracy and completeness of the reference alignment.
- **Availability**: Assesses whether the datasets, the test cases and the reference alignment of the benchmark are readily (and always) available to be used by various systems.

Apart from the qualitative evaluation criteria, the following quantitative ones will also be used:

- **Dissemination**: This criterion measures how many systems have used this benchmark for evaluation showing the impact of the

[7] http://www.tpc.org

benchmark and its acceptance by the community.

- **Volume**: Addresses the volume of the datasets employed, and thus whether the benchmark is also suitable for determining the scalability of the tested systems.
- **Datasets Variations**: This metric assesses the types and complexity of the variations that the datasets consider.

4 Instance Matching Benchmark Generators for Linked Data

In this section, we will shortly present the existing benchmark generators that have been used for the creation of various synthetic instantiations of IM benchmarks. They are deterministic frameworks that take as input various parameters, ran their functions and give as output synthetically created datasets and reference alignments. These Linked Data IM generators are SWING [12], SPIMBENCH [37], and LANCE [62].

4.1 SWING

SWING [12] is the first general framework for creating benchmarks for instance matching applications. It takes as input a source dataset expressed in RDF/OWL format and returns various transformed ontologies, accompanied with a reference alignment for every test case created. It has been implemented as a Java application and it is available at: http://code.google.com/p/swing.

A benchmark is created by SWING in three phases. The first phase is data acquisition, where the appropriate datasets/ontologies are selected from the Linked Data cloud and enriched with appropriate schemas. The ontology enrichment phase is necessary because Linked Data ontologies are usually poor in semantic information, so they are enriched by the addition of more OWL constructs. The second phase is the data transformation phase, in which the schemas and the instances are transformed by employing data, structural, and logical variations into the source ontologies. The last phase is the data evaluation phase, where the reference alignments are created for each test case, and the instance matching tools are tested with the created benchmarks.

4.2 SPIMBENCH

SPIMBENCH [37] is a recently proposed domain specific benchmark. It implements data, structural and logical variations, organized into the so-called simple and complex transformations. A simple transformation involves a combination of variations from the same category, whereas combining variations from different categories results to a complex transformation.

The supported data variations include transformations on instance datatype properties, typographical errors and the use of different data formats (numerical, date etc.). Each variation takes as input a datatype property as specified in SPIMBENCH schema and a severity that determines how important this modification is. SPIMBENCH uses SWING [12] to implement the supported variations, which are a superset of

Property	Original Instance	Transformed Instance
type	"Actor"	"Actor"
wikipedia-name	"James Anthony Church"	"qJaes Anthnodziurcdh"
cogito-Name	"Tony Church"	"Toty fCurch"
cogito-description	"James Anthony Church (Tony Church) (May 11, 1930 - March 25, 2008) was a British Shakespearean actor, who has appeared on stage and screen"	"Jpes Athwobyi tuscr(nTons Courh)pMa y1sl1,9 3i- mrc 25, 200hoa s Bahirtishwaksepearna ctdor, woh hmwse appezrem yo nytmlaenn dscerepnq"

Fig. 2 Mapped instances example from IIMB 2009 with insertion of value variations

those considered in the state of the art instance matching benchmarks.

Structural variations are applied on properties of instances such as property splitting, aggregation, addition, and deletion.

SPIMBENCH puts great emphasis on the logical variations, and is the first benchmark framework to support logical variations (and test cases) that go beyond the standard RDFS constructs. These are primarily used to examine if the matching systems take into consideration OWL and OWL2 axioms to discover matches between instances that can be found only when considering schema information. Namely these constructs are:

- instance (in)equality (*owl:sameAs, owl:differentFrom*)
- class and property equivalence (*owl: equivalentClass, owl:equivalentProperty*)
- class and property disjointness (*owl: disjointWith, owl:AllDisjointClasses, owl:propertyDisjointWith, owl: AllDisjointProperties*)
- class and property hierarchies (*rdfs:subClassOf, rdfs:subPropertyOf*)
- property constraints (*owl:FunctionalProperty, owl:InverseFunctionalProperty*)
- complex class definitions (*owl:unionOf, owl:intersectionOf*)

Briefly, SPIMBENCH includes the phases of data generation, data transformation and reference alignment generation. Concerning the data generation phase, SPIMBENCH extends the data generator proposed by the Semantic Publishing Benchmark (SPB) [14] and produces RDF descriptions of creative works that are valid instances of classes of the ontologies considered by SPB. Therefore, SPIMBENCH is restricted to only one domain. SPIMBENCH is an open source framework and available at http://www.ics.forth.gr/isl/spimbench/.

4.3 LANCE

Linked Data instance matching Benchmark Generator (LANCE) [62] is a framework for assessing instance matching techniques for RDF data that are published with an associated schema. LANCE supports a set of test cases based on transformations that distinguish different types of matching entities. LANCE is a descendant of SPIMBENCH and supports the *value-based* (typos, date/number formats, etc.), *structure-based* (deletion of classes/properties, aggregations, splits, etc.), and *sematic-aware* test cases. Semantics-aware test cases allow testing the ability of instance matching systems to use the semantics of RDFS/OWL axioms to identify matches and include tests involving instance (in)equality, class and property equivalence and disjointness, property constraints, as well as complex class definitions.

The main difference between SPIMBENCH and LANCE, among others, is that it is *domain independent*: it can accept any linked dataset and its accompanying schema as input to produce a target dataset implementing test cases of varying levels of difficulty.

LANCE supports simple combination test cases (implemented using the aforementioned transformations applied on different triples pertaining to the same instance), as well as

Original Instance	Transformed Insance
type (uri1, "Actor")	**type** (uri2, "Actor")
cogito-Name (uri1, "Wheeler Dryden")	**cogito-Name** (uri2, "Wheeler Dryden")
cogito-first_sentence (uri1, "George Wheeler Dryden (August 31, 1892 in London - September 30, 1957 in Los Angeles) was an English actor and film director, the son of Hannah Chaplin and" ...)	**cogito-first_sentence** (uri2,uri3)
	hasDataValue (uri3, "George Wheeler Dryden (August 31, 1892 in London - September 30, 1957 in Los Angeles) was an English actor and film director, the son of Hannah Chaplin and" ...)
cogito-tag (uri1, "Actor")	**cogito-tag** (uri2,uri4)
	hasDataValue (uri4, "Actor")

Fig. 3 Mapped instances example from IIMB 2009 with insertion of structural variations

Property name	Original instance	Transformed instance
type	"Sportsperson"	owl:Thing
wikipedia-name	"Sammy Lee"	"Sammy Lee"
cogito-first_sentence	"Dr. Sammy Lee (born August 1, 1920 in Fresno, California) is the first Asian American to win an Olympic gold..."	"Dr. Sammy Lee (born August 1, 1920 in Fresno, California) is the first Asian American to win an Olympic gold ..."
cogito-tag	"Sportperson"	"Sportperson"
cogito-domain	"Sport"	"Sport "

Fig. 4 Mapped instances example from IIMB 2009 with insertion of logical variations

complex combination test cases (implemented by combinations of individual transformations on the same triple). However, there has not been an official call for a specific benchmark that has been created using LANCE, so that systems can participate and report results, but this is in the future plans of the authors of [62].

LANCE is open source and is available at http://www.ics.forth.gr/isl/lance/.

5 State of the Art in Instance Matching Benchmarks for Linked Data

In this section we present the state of the art instance matching benchmarks. We will describe in detail each benchmark and the reported results of the systems tested using this benchmark. In this way, we will show the (relative) performance of the tested systems. Our goal is not to evaluate the systems themselves, but to assess and evaluate the benchmarks by taking into consideration the systems' results.

The most important challenge, introduced for judging the performance of instance matching techniques and tools, is the Ontology Alignment Evaluation Initiative (OAEI)[8], which is the most popular framework for testing instance matching systems for Linked Data. It is a coordinated international initiative to conduct the evaluation of ontology matching solutions and technologies using a fixed set of benchmarks. OAEI organizes an annual campaign for ontology matching since 2005. Starting from 2009, the instance matching (IM) track was introduced by OAEI, which focuses on the evaluation of instance matching techniques and tools for RDF and OWL. Each IM track provides different benchmarks to judge the completeness and correctness of instance matching approaches. The instance matching tracks are either created by OAEI or individually created from other parties and hosted under OAEI such as for instance SPIMBENCH at the 2015 track.

We continue by analysing and evaluating other state of the art benchmarks, dividing our analysis into two categories based on the nature of the used datasets (real or synthetic).

5.1 Synthetic Benchmarks

In the next section we will review the synthetic benchmarks that have been proposed so far by the research community. Fig. 6 presents the overall evaluation of the benchmarks that we study in this section.

IIMB 2009

The first synthetic benchmark proposed for instance matching Linked Data, is the ISLab Instance Matching Benchmark (IIMB) in 2009, introduced by OAEI [10]. The benchmark was created by using the ontology created in the context of the OKKAM project[9] that contains actors, sports persons, and business firms. The ontology comprises of 6 classes and 47 datatype properties. It is limited in volume, since it only contains up to 200 instances. The benchmark is divided into 37 test cases, where each test case

[8] Ontology Alignment Evaluation Initiative http://oaei.ontologymatching.org/

[9] OKKAM project: http://project.okkam.org/

Fig. 5 Example from reference alignment API/EDOL format of IIMB 2009 benchmark

contains a source dataset, a target dataset and a reference alignment. The target datasets are obtained by employing data, structural and logical variations. More specifically, test cases 2-10 contain only data variations (e.g. typographical errors, use of different formats, etc.), test cases 11-19 contain only structural variations (e.g. property deletion, change of property types, etc.), test cases 20-29 contain only logical variations (e.g. subsumption assertions, modified class assertions, etc.), and test cases 30-37 contain combinations of the above variations.

Fig. 2 shows an example of value variations where typographical errors have been used to obtain the transformed instance of the target dataset and Fig. 3 depicts an example of structural variations, where datatype properties from the source instance have been transformed to object properties of the target instance.

Examples of logical variations are shown in Fig. 4. The type of the source instance has been altered in the target dataset; since the tested instance matching systems can infer from the ontology schema that class "sportsperson" is a sub-class of class "Thing" and the two instances share the other properties, one can conclude that the two instances refer to the same person.

The reference alignment for each test case is automatically created and thus, is accurate without any errors or missing answers. The reference alignment is represented in an RDF/XML file (see Fig. 5), and follows the API/EDOL format [72] which contains the match in the form of a cell, where each cell includes the matched instances, the measure element (in that case a float value always equal to 1.0), and a relation element (which is the equal sign "="). The same format is also used in all the benchmarks of OAEI that will be discussed in the following.

The systems that participated in the IIMB benchmark 2009 campaign are the Aflood [38], ASMOV [23], DSSim [30], HMatch [19], FBEM [40], and RiMOM [46]. Apart from these systems the OtO sytem [68] has reported results by running the same benchmark.

The overall reported results from [10] (p. 39, table 26) and those of the OtO system, provide interesting conclusions. Some tests appear to be easy, for instance in the value transformation test cases 2-10 all systems except DSSim had excellent results. Other test cases proved to be more difficult to the systems: test cases 30-37 that exhibit a combination of transformations challenged some of the tested systems. We can conclude that for the majority of the systems the benchmark was considered as easy, since they all show very good results except in certain cases. An overall conclusion summarized in Fig. 6 b ``shows that it is a balanced benchmark, with two main disadvantages. The first disadvantage is that it the datasets are very small (only up to ~200 instances), so it cannot be used to test the scalability aspect of the instance matching systems. The second is that the logical variations employed to obtain the test cases are only for testing the ability of the systems to understand subsumption relationships and anonymous classes. Thus, the supported logical variations are very limited.

IIMB 2010

IIMB 2010 [9] proposed by OAEI, is a benchmark employing synthetic datasets, and the second synthetic benchmark of the IIMB benchmark sequel. Created using as base the Freebase Ontology[10], the benchmark included a small version (consisting of 363 instances) and a

[10] Freebase Ontology http://www.freebase.com/base/fbontology

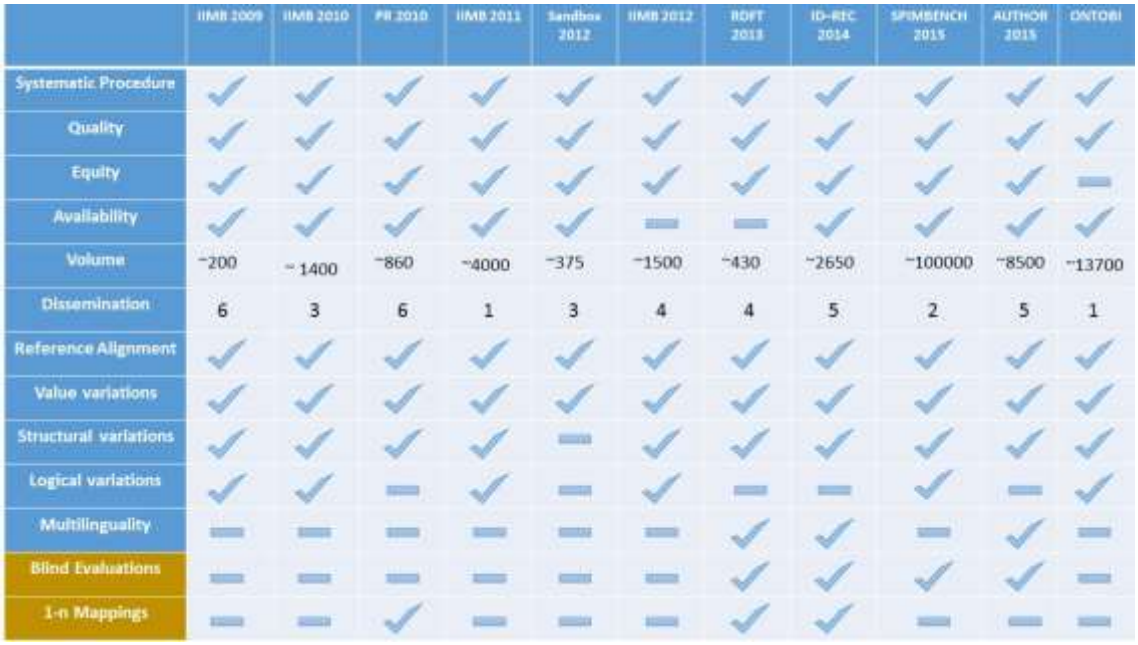


Fig. 6 Overall evaluation of synthetic benchmarks

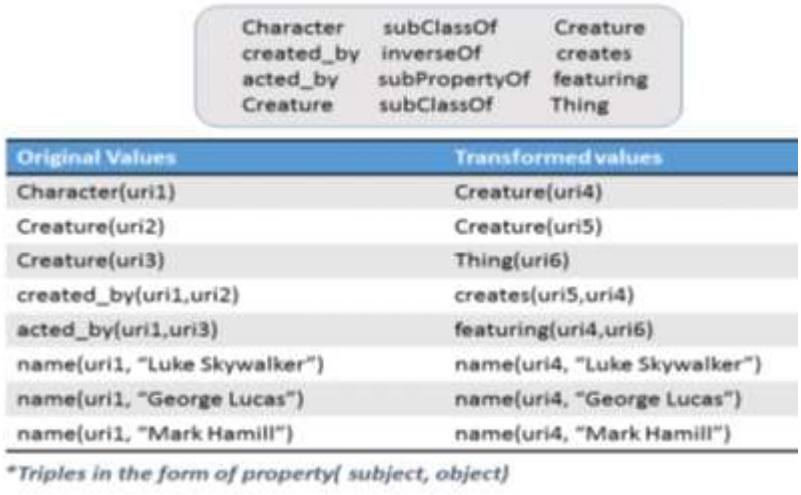
larger one (containing 1416 instances) thereof. The small version of the Freebase Ontology consists of 29 classes, 32 object and 13 datatype properties, while in the large version the ontology considers 81 classes. IIMB 2010 contains a larger volume of instances, compared to IIMB 2009, but still remains a small scale instance matching benchmark. The benchmark is divided into 80 test cases (all of them addressing, the variations presented in Fig. 1) as follows: test cases 1-20 contain all data variations except multilinguality. Test cases 21-40 contain all structural variations and test cases 41-60 contain all logical variations. Finally test cases 61-80 contain a combination of the above.

Fig. 7 shows one particular example of the dataset used in this benchmark. The left column shows an instance in the source ontology and the right column shows the instance in the target ontology obtained from the former by applying a set of variations discussed previously. For this particular example, the tested systems had to consider schema level information. In particular, the combination of the facts that (a) class "Character" is a subclass of class "Creature", (b) property "created_by" is the inverse property of "creates", (c) property "acted_by" is a sub-property of "featuring", and (d) class "Creature" is a subclass of class "Thing", one can conclude that the two instances possibly refer to the same real matching process.

The reference alignment was automatically created using the same format as IIMB 2009 (see Fig. 5). Moreover, the benchmark was created using the SWING Tool [12] (see also subsection 4.1).

The systems that participated in OAEI IIMB 2010 were ASMOV [24], RiMOM [43], and CODI [34]. The reported results of the systems are shown in [21] (p. 28). All systems reported excellent results for the test cases that considered data and logical variations. Lower-quality results were reported in structural transformation test cases and even lower in cases considering combinations of transformations. Apart from these systems, also LINDA [55] reported results on the small scale task of this benchmark. LINDA reports very good results in precision, but not so good when it comes to recall: it includes almost all the correct results in the result set, but it also includes many false results (false negatives) in it.

The results illustrate that systems were not ready to deal with the scenario considering combinations of transformations, which is actually the most realistic one. An overall evaluation of IIMB 2010 is shown in Fig. 6. The main points here are that (a) compared to IIMB 2009, it was more voluminous, but still a small scale benchmark and (b) that the logical variations considered by the test cases are all the variations types mentioned in Fig. 1. It is the first time that a



Character	subClassOf	Creature
created_by	inverseOf	creates
acted_by	subPropertyOf	featuring
Creature	subClassOf	Thing

Original Values	Transformed values
Character(uri1)	Creature(uri4)
Creature(uri2)	Creature(uri5)
Creature(uri3)	Thing(uri6)
created_by(uri1,uri2)	creates(uri5,uri4)
acted_by(uri1,uri3)	featuring(uri4,uri6)
name(uri1, "Luke Skywalker")	name(uri4, "Luke Skywalker")
name(uri1, "George Lucas")	name(uri4, "George Lucas")
name(uri1, "Mark Hamill")	name(uri4, "Mark Hamill")

*Triples in the form of property(subject, object)

Fig. 7 Example from IIMB 2010

benchmark includes all the variation types in the semantics-aware test cases.

Persons-Restaurants (PR) 2010

PR benchmark [9] was also proposed in 2010 as the second synthetic benchmark of OAEI. This benchmark was created by using datasets from the Febrl project[11] for the test cases "Person 1" and "Person 2", and the Fodor and Zagat's Restaurant Guide[12] datasets for the test case "Restaurant". "Person 1" and "Person 2" test cases, contain information about persons in the source and the target datasets that should be matched. The difference between these two test cases is that "Person 2" has more modifications considered into the datasets, thus adds a higher degree of difficulty for the instance matching systems.

Another difference between these test cases is that "Person 2" includes clustering matchings, i.e., 1-n matchings in the reference alignment. That means that one instance from the source could be a match with more than one instances from the target dataset. Last, but not least, the "Restaurant" dataset contains information about restaurants such as their name, street, city, phone, and category. In all the datasets the number of instances is quite limited (about 500-600 instances). In all test cases a combination of value and structural modifications is included, but not logical modifications, as the exact same schema is used for the source and target dataset. This is due to the fact that, the benchmark datasets were relational that have been converted into RDF.

Again, the reference alignment for each test case were automatically created and have the same format as IIMB 2009 (see Fig. 5).

The systems that participated in PR 2010 were ASMOV [24], ASMOV-D [24], CODI [34], LN2R [36], ObjectCoRef [21], and RiMOM [43]. Apart from these systems, also LINDA [55] reported results for PR. PR was disseminated well, since it contains a small number of instances to be matched, resulting in a matching task that is affordable in terms of time required for comparisons, but apart from this it also does not contain any semantics-aware challenges for the systems. Authors in [21] (page 30), show the reported F-measure results of the systems. It can be noted that half of the systems in addition to the LINDA system exhibit significantly worse results in "Person 2" test case. This is because some systems could not handle the 1-n matching provided by this test case, resulting to a lower F-measure.

The interested reader can refer to Fig. 6 for an overall evaluation of the benchmarks. PR 2010 can be considered an "easy" benchmark when dataset volume is considered. It considers value and structural transformations, but no logical ones. It was successful in terms of participation and it is the first instance matching benchmark that tested the ability of the systems to return 1-n mappings.

[11] Febrl project, http://sourceforge.net/projects/febrl/

[12] Fodor and Zagat's Resaturant Guide http://userweb.cs.utexas.edu/users/ml/riddle/data.html

IIMB 2011

IIMB 2011 is the third IIMB benchmark proposed by OAEI in 2011 [8]. The Freebase Ontology was again used as the schema of the datasets (as was the case also in IIMB 2010). Another common feature shared between IIMB 2011 and IIMB 2010, is that they were both created using the SWING tool [12].

The test cases are divided into the usual four categories: test case 1- 20 contain value variations, test cases 21-40 contain only structural variations, test cases 41-60 contain only logical ones, and test cases 61-80 contain a combination of the above. All types of variations are the same as in IIMB 2010. Once more, the reference alignments were created automatically, without any errors or missing results and are in the same format as in IIMB 2009. The maximum number of instances for the test cases was approximately 4K, unlike previous benchmarks that contained up to 1400 instances. This is possibly the reason why only the CODI [8] system has reported results for this benchmark. From the results reported for CODI, discussed in [22], it is once more obvious that the worse results in terms of F-measure, are reported in the test cases which include a combination of value, structural, and logical variations.

The overall evaluation of IIMB 2011 (see Fig. 6) shows that it is a synthetic benchmark with the highest number of instances offered so far from the OAEI synthetic benchmarks. However, dissemination was poor, possibly due to the size of the dataset in combination with the need of the usage of semantic reasoners, which prevented the systems from reporting results.

Sandbox 2012

The Sandbox 2012 benchmark was proposed by OAEI in 2012 [1], and was again created using the Freebase Ontology. It included up to ~400 instances and all the 10 test cases of the benchmark contained only value variations. The reference alignments were automatically created, using the same format as all the previous OAEI benchmarks.

Sandbox 2012 was admittedly an easy benchmark [1]. It was created with a simple dataset that has been specifically conceived to provide examples of some specific matching problems (like name spelling and other controlled value variations). It was intended as a test for tools that are in the initial phases of their development process and/or for tools that are facing focused tasks. In addition, Sandbox 2012 served the purpose of attracting new instance matching systems to participate in the instance matching track. In fact, all the systems that have reported results for the Sandbox benchmark are new in the IM track of OAEI.

The results for Sandbox and for systems LogMap [25], LogMap Lite [25], and SBUEI [41] are shown in [1] (p.37, Table 15). All the results are excellent and the F-measure is over 0.90 for all systems.

When comparing Sandbox 2012 with the remaining benchmarks (see Fig. 6) we can see that it is not a very challenging benchmark in terms of volume and complexity. The complexity of the modifications in the test cases is low, since they only contain value variations in the datasets.

IIMB 2012

IIMB 2012 [1], again proposed by OAEI was considered as the *enhanced* Sandbox Benchmark. It was based on the Freebase Ontology and contained various kinds of test cases including value, structural, and logical variations. The SWING benchmark generator was used in the creation of the benchmark, like in IIMB 2010 and IIMB 2011. All variation types described in Fig. 1 have been used in the generation of the datasets.

The systems that participated in the IIMB 2012 benchmark were LogMap [25], LogMap Lite [25], SBUEI [41], and Semsim [1]. The results for this benchmark for different systems are reported in [1] (p.39, Table 16). In general the results are very good, but we see again that systems report worse results for the test-cases 61-80, which are the datasets that contain the combination of variations (structural and logical variations). Fig. 6 shows the overall evaluation of IIMB 2012. As one can notice, IIMB 2010, IIMB 2011, and IIMB 2012 are very similar. They all contain the same types of variations (since all of them have been created with SWING), and furthermore they use the Freebase Ontology.

RDFT 2013

Property	Original Instance
type	namespace#Book
label	"Flood (Baxter novel)"
publisher	"Victor Gollancz Ltd"
subject	"2008 novels"
subject	"21st century in fiction"
literaryGenre	"Fantascienza"
Genre	"Fantascienza"
Author	"Stephen Baxter"

Property	Transformed Instance
type	namespace#Book
label	"categorytranshumanist books, orion publishing, stephen baxter"
publisher	"Orion Publishing Group"
publisher	"Victor Gollancz Ltd"

Fig. 8 Example from ID-REC 2014

In 2013 OAEI proposed the RDFT benchmark [16]. This benchmark was created by extracting data from DBpedia. It is divided into five test cases, were the first considers only value variations for the target dataset, the second structural variations, the third language variations (English to French), and the last two contain combinations of the above.

RDFT is the first synthetic benchmark for IM proposed by OAEI that contained multilingual datasets. It is also the first synthetic benchmark that conducted the so called "blind evaluations", i.e., evaluations were the participating systems were not aware of the reference alignment when running the test. This means that the systems could not optimize their results (by running learning algorithms for example), since they were not aware of the correct matches. They were only given training data, in order to "train" their systems with the particular needs of each test case. Last, but not least, one of the proposed test cases contained clusters of mappings, i.e., 1-n mappings, which was also the case in PR 2010 benchmark.

The systems that reported results for RDFT benchmark (see Table 15 of [16]) are LilyIOM [16], LogMap [26], RiMOM [47], and SLINT+ [32]. All systems exhibited good performance when dealing with a single type of variations, i.e., test cases 1-3 that contained either data, structural or language variations. Performance drops when different kinds of transformations are combined together (i.e., in test cases 4-5), except for RiMOM, whose precision and recall was close to 1.0. This suggests that a challenge for instance matching tools is to work in the direction of improving the combination and balancing of different matching techniques in one dataset. Furthermore, systems seem to handle multilinguality well when the English and French languages are considered (test case 3).

Last but not least, this benchmark also does not address the ability of the systems to handle large datasets since the size of the considered data instances (the maximum number being around 430) is very limited.

ID-REC 2014

In 2014 call, OAEI proposed another instance matching track [5] with two tasks, namely the *identity recognition task* ("id-rec" task) and the *similarity recognition task* ("sim-rec" task). Since we are interested in the identity recognition problem, we will analyse the first benchmark task only.

The "id-rec" benchmark contains one test case, where the participants are asked to match books from the source dataset to the target dataset. Constraints are expressed in terms of OWL constructs. The source dataset contains 1330 instances described using 4 classes, 5 datatype properties, and 1 annotation property. The target dataset contains 2649 instances described using 4 classes, 4 datatype properties, 1 object property, and 1 annotation property.

Primarily, the main kind of transformation that was performed into the source data was to transform the structured information into an unstructured version of the same information. Specifically, labels are substituted with a set of keywords taken from the instance description. Furthermore, data was translated from Italian to

English and vice-versa. Fig. 8 shows an example of the source and target instances considered as matches. As one can see from the example, the only structured data element that the transformed instance shares with the original instance, is the publisher. Other than that, the transformed instance contains into the label property the keyword “stephen baxter” which is the author of the original book.

The problem of matching structured data with a semi- or un-structured version of such data is still an open issue for the research community, and that was the reason why the quality of the reported alignments were in general not very high.

The systems that have reported results for this benchmark were InsMT [28], InsMTL [28], LogMap [27], LogMap-C [27], and RiMOM-IM [39]. These results, in terms of precision, recall, and F-measure are provided in [6] (p.37). Note that the majority of the systems (i.e., all except RiMOM) either reported fairly good results for precision and bad results for recall, or vice-versa. This means that the systems either returned very few results that were correct, but at the same time missed a lot of correct results (high precision - low recall), or returned many results, the majority of which were wrong (high recall - low precision).

Other characteristics of the benchmark include blind evaluations and the fact that the reference alignment contained 1-n mappings.

Concluding, the instance matching benchmark did not contain logical variations. This means that the semantics expressed at the schema level were not considered when the matching process was performed. The benchmark did not address the ability of systems to handle large datasets since it contains a small number of instances. Nevertheless, the benchmark was well disseminated.

SPIMBENCH 2015

In the 2015 IM call, OAEI introduced the SPIMBENCH 2015 track [69]. The track included 3 test cases, namely value-semantics ("val-sem"), value-structure ("val-struct"), and value-structure-semantics ("val-struct-sem") (all created by using SPIMBENCH [37]).

The source and target datasets have been produced by altering a set of source data with the aim to generate descriptions of the same entity where *value-based*, *structure-based* and *semantics-aware* transformations are employed in order to create the target data. The target dataset is produced in a way such that an instance in the source dataset can have none or one matching counterpart in the target dataset. The ontology contains 22 classes, 31 datatype properties, and 85 object properties. From those properties, 1 is an inverse functional property and 2 are functional properties. The test cases are available in two different scales namely the "sandbox" scale that contains 10.000 instances and the "mainbox" that scale contains 100.000 instances.

The participants of these tasks were LogMap [49] and STRIM [48]. For evaluation, a reference alignment was available which contains a set of expected matches.

Both systems have presented good results [69], something that shows that the benchmarks did not pose any big challenge for them, although the benchmark had up to 100K instances, which is the biggest synthetic benchmark so far that also considered logical variations. After all this benchmark was created to be the easiest benchmark in comparison with the Author 2015 [69]. LogMap and STRIM have consistent behaviour for the "sandbox" and the "mainbox" tasks, a fact that shows that both systems can handle different sizes of data without reducing their performance. LogMap’s performance drops for tasks that consider structure-based transformations (val-struct and val-struct-sem). Also, it produces links that are quite often correct (resulting in a good precision) but fails in capturing a large number of the expected links (resulting in a lower recall). STRIM’s performance drops for tasks that consider semantics-aware transformations (val-sem and val-struct-sem) as expected. The probability of capturing a correct link is high, but the probability of a retrieved link to be correct is lower, resulting in a high recall but not equally high precision.

Author 2015

The Author benchmark [69] introduced the "*author-dis"* and "*author-rec"* subtasks whose

goal is to discover links between pairs of OWL instances referring to the same person (i.e., author) based on their publications. In both tasks, it is expected that one person of the source dataset corresponds to exactly one person of the target dataset and vice versa. The task also provided two datasets of different sizes: the "sandbox" and "mainbox" that are used to test the performance of the instance matching systems for small and larger datasets respectively.

Author and publication information is described in a different way in the two datasets. For example, only the first letter of author names and the initial part of publication titles is shown in the target dataset while the full strings are provided in the source datasets. The matching challenge regards the capability of systems to resolve such types of ambiguities on author and publication descriptions.

Regarding the *"author-rec" task*, author and publication descriptions in the source dataset are analogous to those in the *"author-dis"* task. As a difference, in the target dataset, each author/person is only associated with a publication titled "Publication report" containing aggregated information, such as number of publications, h-index, years of activity, and number of citations. The matching challenge regards the ability of systems to link a person in the source dataset with the person in the target dataset containing the corresponding publication report.

Participants to author-dis and author-rec tasks are EXONA[53], InsMT+[52], Lily[51], LogMap[49] and RiMOM[50].

On the *author-dis* task, the results are good in terms of precision and recall. As a general remark, precision values are slightly better than recall values. This behaviour highlights the consolidated maturity of instance matching tools when the alignment goal is to handle syntax modifications in instance descriptions.

On the *author-rec* task, the differences in tool performances are more evident. In particular, Lily, LogMap, and RiMOM have better results than EXONA and InsMT+. Probably, this is due to the fact that the capability to associate the summary publication report with the corresponding author requires reasoning functionalities that are available to only a subset of the participating tools. Regarding the ability of systems to address datasets of different sizes: LogMap and RiMOM are the best performing tools on the "mainbox" tests. This is a challenging issue in the field of instance matching, on which further experiments and tests need to focus in the future competitions.

ONTOlogy matching Benchmark with many Instances (ONTOBI)

ONTOBI is a synthetic benchmark [45] that uses the schema and datasets from DBpedia and it is not introduced by OAEI. It is divided into 16 test cases, which contain data, structural and logical variations, as well as combinations thereof. Logical variations are only limited to "expanded structure" and "flatten structure", without exploiting any other construct of the RDF or OWL languages. Another issue of the ONTOBI benchmark is that the only system that has reported results for the benchmark is the MICU system [44], which was created by the authors of ONTOBI. Thus, we consider ONTOBI as an isolated benchmark and we will not proceed with commenting the results of their system.

Fig. 6 shows the overall evaluation of the ONTOBI benchmark according to the dimensions discussed in section 3.4. It is worth noting that ONTOBI only includes a few logical variations, which are the most expensive type (in terms of computational resources). As far as equity is concerned, we cannot possibly know since no other system has reported results for this benchmark, other than MICU system.

5.2 Overall Evaluation of Synthetic Benchmarks

Fig. 6 shows a summary of the evaluation of the synthetic instance matching benchmarks that have been presented so far.

It is obvious that each benchmark has been created for a specific reason. For example, the IIMB series and SPIMBENCH 2015 ("sandbox" version) of OAEI benchmarks have all been created to test the ability of the systems to cope with different kinds of variations, i.e., data, structural, and logical, but they are all limited in volume (up to 10K instances). PR, RDFT, and

ID-REC benchmarks test the ability of the systems to cope with 1-n mappings. Furthermore, SPIMBENCH, RDFT, ID-REC and Author benchmarks of OAEI evaluations also include multilingual tests. Sandbox is considered to be a trivial benchmark, with no special challenges, with the aim to attract new systems to the instance matching track of OAEI. ONTOBI seems not to consider many logical variations and does not test the ability of the systems to use semantic information of the accompanied schemas when those performing the matching task.

The question that remains to be answered by looking at the overall picture of synthetic benchmarks is related to the size of the considered datasets: even though the number of instances that are uploaded in the Linked Data cloud is in the order of billions of triples, all the proposed synthetic benchmarks consider small datasets (at the range of thousands of instances). Before addressing that question, it is interesting to have a look at the real benchmarks that have been published so far.

5.3 Real Benchmarks

In this section we describe the benchmarks employing real datasets (or parts of real datasets) without introducing synthetically produced variations. An overall evaluation of real benchmarks is shown in Fig. 9.

ARS 2009

The ARS benchmark [10], proposed by OAEI in 2009, is created by using three different datasets, namely the AKT-Eprints archive (which contains information about papers produced within the AKT project[13]), the Rexa dataset (which includes computer science research literature, people, organizations, venues and data about research communities'), and the SWETO-DBLP dataset (which is a publicly available dataset listing publications from the computer science domain). All three datasets were structured using the same schema, namely SWETO-DBLP ontology[14].

The benchmark is divided into three test cases: a test case for matching instances in AKT Eprint to instances in Rexa; a test case for matching instances in AKT Eprint to instances in DBLP; and a test case for matching instances in Rexa to instances in DBLP.

The datasets contain two classes, namely foaf:Person and sweto:Publication. The volume of the datasets is as follows:

- AKT-Eprints: **564** instances of foaf: Person and **283** instances of sweto:Publication
- Rexa : **11.050** instances of foaf: Person and **3.721** instances of sweto:Publication
- SWETO-DBLP : **307.774** instances of foaf:Person and **983.337** instances of sweto:Publication

The challenges of the benchmark are threefold, specifically: (a) it has a large number of instances (up to almost 1M); (b) it contains ambiguous labels (person names and paper titles); and (c) it contains noisy data.

The reference alignment of the test cases was manually created, which was a time-consuming and error-prone process. The AKT-Rexa test case contains 777 overall mappings in the reference alignment, the AKT-DBLP contains 544 mappings, and the Rexa-DBLP contains 1540 mappings.

The benchmark was successful in terms of participation, since five systems reported results. Table 25 (page 37) of [10] shows results for the tested systems, namely DSSim [30], RiMOM [46], FBEM [40], HMatch [19], and ASMOV [23]. DSSim did not publish any results for the Rexa-DBLP test case, as well as the sweto:publication matchings for the AKT/DBLP test case. This may imply that DSSim system had scalability problems, since the DBLP was the most voluminous dataset. The same goes for the ASMOV system, which reported only results for the test case AKT/Rexa, which the test case that considered the smaller number of instances.

Another interesting finding is that in almost all cases of foaf:Person instances, systems report worse results in comparison to the respective results for sweto:Publication instances. This is due to the fact that instances of persons contained structural variations which seemed to pose

[13] AKT project http://www.aktors.org/

[14] SWETO-dblp: http://datahub.io/dataset/sweto-dblp

	ARS	DI 2010	DI 2011
Systematic Procedure	✓	✓	✓
Quality	✓	✓	✓
Equity	✓	✓	✓
Availability	✓	✓	–
Volume	~1M	~6000	?
Dissemination	5	2	3
Reference Alignment	✓	✓	+
Value variations	✓	✓	✓
Structural variations	✓	✓	✓
Logical variations	–	–	–
Multilinguality	–	–	–
Blind Evaluations	–	–	–

Fig. 9 Overall evaluation of real benchmarks

difficulties for some of the systems, and lowered their F-measure.

To conclude, there are systems that exhibit good results, e.g., RiMOM, and others that do not have very good results like FBEM and DSSim; this makes this benchmark one of the most balanced ones.

From the overall evaluation of the ARS benchmark (see Fig. 9) we see that it is a benchmark which tests the ability of the systems to scale, but at the same time contains a possibly erroneous reference alignment (since it was manually curated). Another characteristic of ARS is that it does not contain any logical variations, which means that systems did not have to consider schema information to match the instances.

Data Interlinking (DI) 2010

In 2010, OAEI proposed one more real benchmark, the DI benchmark [9]. All datasets that have been used in the DI benchmark are taken from the health-care domain and contain information about drugs:

- the DailyMed Ontology , which provides marketed drug labels containing 4308 drugs,
- the Diseasome Ontology, containing information about 4212 disorders and genes,
- the DrugBank Ontology, which is a repository of more than 5900 drugs approved by the US FDA, and
- the SIDER Ontology, which contains information on marketed medicines (996 drugs) and their recorded adverse drug reaction (4192 side effects).

The task posed by the DI benchmark is to interlink the input datasets. The DI benchmark was the first real benchmark where the reference alignments were created in a semi-automatic way. Two instance matching systems were used for this purpose, namely the Silk system [42] and the LinQuer system [18]. The format that the reference alignments were expressed in, is the same format used in IIMB 2009.

Only two systems were evaluated using this benchmark, namely RiMOM [43] and ObjectCoRef [21] (the results are shown in [9], p. 26). But apart from these, other systems have reported results, namely the ActiveGenLink in [64] and GenLink in [65].

From the results it is obvious that the two systems had different strategies, as RiMOM generally exhibits better results in recall and worse in precision, whereas ObjectCoRef exhibits better results in precision and worse results in recall. In other words, RiMOM did report most of the correct results, but it also had a large number of false positives. On the other hand, ObjectCoRef

did not contain many of the correct results in its result set, but also did not report any false positives. When it comes to ActiveGenLink and its ancestor GenLink, the results reported in [64] and [65] are very good when compared to the reference alignment. This is also due to the fact that RiMOM and ObjectCoRef conducted blind evaluations (unsupervised systems), so they had no a priori knowledge of the entire reference alignment. On the other hand, ActiveGenLink and GenLink used the existing reference alignments in order to learn the linkage rules (supervised learning system).

The interesting point of OAEI regarding this benchmark is that *"the main task with real interlinked data is to understand if the results are due to a weakness of the matching systems or because links cannot be very reliable"* [9]. So the main point here is that it is important to have good quality reference alignments, otherwise it is hard to assess systems' quality.

The instances considered in DI 2010 are in the order of thousands, much less than the ARS benchmark. Moreover, this benchmark did not contain any logical variations, so systems did not need to conduct any reasoning to perform the matchings. Last but not least, the dissemination of the benchmark was not very successful, as only two systems participated in the evaluation.

Data Interlinking (DI) 2011

The DI 2011 benchmark is the second benchmark in the series of Data Interlinking benchmarks from OAEI [8]. This benchmark focuses on retrieving New York Times subject headings links with DBpedia locations, organizations and people, Freebase locations, organizations and people, and Geonames locations.

The reference alignments are based on already existing links in the NYT dataset. So, the provided mappings are accurate but may not be complete. These links have been removed and participants' goal was to restore them.

The systems that participated in this benchmark are AgreementMaker [5], ZhishiLinks [33], and SERIMI [3]. Table 11 in [8] shows the results of the above systems. But apart from these, ActiveGenLink [64] and GenLink [65] have also reported results for this benchmark.

ZhishiLinks produces consistently high quality matches over all data sets and obtains the highest overall scores, but AgreementMaker and SERIMI also report very good results.

As far as the ActiveGenLink and GenLink results are concerned, they have very high F-scores according to the reference alignment. Again here we have to mention that both of these systems employ supervised learning algorithms.

The hypothesis for the fact that all the systems reported fairly good results is that they had to deal with well-known domains, but also with a well-known dataset. Apart from that, the systems did not use any schema information when running the benchmark since no logical variations were included. From the overall evaluation of DI 2011 (Fig. 9), two things are worth commenting: the first is that the datasets were not available online, and the second is that this benchmark did not include any logical variations.

5.4 Overall Evaluation of Real Benchmarks

Fig 9 shows a summary of the overall evaluation of the real benchmarks that we discussed in this paper. We can see that there is one benchmark that tests the ability of the systems to scale, namely the ARS benchmark with 1M instances.

Moreover, none of the real benchmarks addresses logical variations since they consider mostly value and structural variations.

Furthermore, the first two benchmarks included reference alignments that contained errors, while the last benchmark, DI 2011, contained an error-free reference alignment, due to the fact that the links between the datasets where already included in the NYT dataset. Regarding the last point, it is still an open issue for the research community to provide a reliable (semi-)automated mechanism to create good quality reference alignments for real datasets.

6 Discussion

6.1 Selecting the Proper Benchmark

As already mentioned, one of the main objectives of this survey is to help with the identification of the appropriate benchmark for a given setting. In this subsection we revisit the presented benchmarks from the perspective of the systems, in order to answer this question.

One of the most important considerations is whether the selected benchmark should be based on real or synthetic datasets. Benchmarks with synthetic datasets offer fully controlled test conditions and accurate reference alignments, but potentially unrealistic distributions and systematic heterogeneity problems. On the other hand, benchmarks based on real datasets provide more realistic data variations and distributions, but their reference alignment is error-prone impacting negatively the performance of instance matching tools.

If multilinguality support is important, then RDFT 2013, ID-REC 2014, and Author 2015 should be the benchmarks of choice. RDFT 2013 considers the English and the French language, while the others the Italian and English languages.

If data and structural variations are the most important cases that need to be tested, and it is a requirement that the chosen instance matching system should perform well for data and structural variations, then all of the benchmarks that have been analysed in this survey are appropriate, except Sandbox 2012. Sandbox supports only value variations, but all others stress the instance matching systems for both value and structural variations.

However, if the identification of matches that are related to the ontology schema is also relevant (logical variations), then the most appropriate benchmarks are IIMB 2009, IIMB 2010, IIMB 2011, IIMB 2012, SPIMBENCH and ONTOBI. Note however, that IIMB and ONTOBI benchmarks test only the basic axioms of the RDF and OWL languages and only SPIMBENCH 2015 goes beyond the the standard RDFS constructs.

Finally, if combinations of variations are needed (as is the case with most realistic instance matching cases), then the appropriate benchmarks are IIMB 2009, IIMB 2010, IIMB 2011, IIMB 2012, SPIMBENCH and ONTOBI.

Another important feature is the ability of the instance matching system to scale for voluminous datasets. If scalability or the combination of all variations is important, then the ARS and SPIMBENCH benchmark is the most appropriate. Even with these benchmarks the scalability remains a crucial topic for further research.

6.2 Future Benchmark Directions

An interesting point regarding future benchmark directions, is that there is a need of benchmarking systems that work with special domain specific data with specific characteristics. Such data are for example geospatial and tempo-spatial data,.

Geo-spatial data, are an interesting category of data. In addition to their *semantic dimension* (information knowledge represented in terms of geospatial ontologies) and *topological dimension* (incomplete knowledge of topological names) they also consider knowledge based on information that describes the position, the extension, and implicitly the shape and relationship between spatial features [73]. Tempo-spatial data are even more complicated due to the fact that the dimension of time perplexes the facts of existence or not existence (as well as the meaning) of places and/or names.

But apart from domain specific special data, another interesting benchmark is a kind of triangular (or transitive) benchmark. In our survey we have described benchmarks that contain source and target datasets, but what we propose to is a new kind of instance matching benchmark for Linked Data that will be composed of three (or even more) datasets where we would like to discover links between the source and target dataset through the network of the other interlinked ones. In effect, such a benchmark will evaluate whether the additional information elicited by the third dataset can improve the matching performance (precision, recall) of systems. These tasks have already been applied in Open PHACTS project platform [74], and, in our opinion, constitute a very promising benchmark direction.

6.3 Conclusions

In this survey, we studied the state of the art instance matching benchmarks for Linked Data. We described many proposed benchmarks as well as benchmark generators, with diverse abilities and characteristics, and also identified open issues. Furthermore we have analysed instance matching systems' behaviour when evaluated with the different benchmarks.

According to Homoceanu et al. [20] "*instance matching is not yet ready for reliable automatic data interlinking*". Our survey has shown that some systems present good results in specific problems, but also showed that systems fail when it comes to the real world scenario, which is the combination of variations in the datasets.

Another major conclusion of our survey is the lack of a benchmark that tackles both scalability issues and logical variations into the datasets, except LANCE and SPIMBENCH. Datasets of all synthetic benchmarks are in the order of some thousands in terms of dataset size, which is not enough to test the ability of systems to scale; on the other hand, even though some of the real benchmarks do test scalability, they fail to support schema matching and logical variations. Since we are in the era of Linked Data, systems should be able to exploit all the given information that a dataset may contain.

The bottom line is that many instance matching benchmarks have been proposed, each of them answering to some of the needs of instance matching systems. It is high time now to start creating benchmarks that will "show the way to the future" and extend the limits of existing systems in the ways mentioned above. In other words, benchmarks should not be created only to answer to the needs of evaluation of the systems, but also new systems and algorithms should be created to answer to challenging benchmarks. To this end, our survey has proposed future directions on proposed instance matching benchmarks for Linked Data.

Acknowledgments

This work was supported by the project HOBBIT, which has received funding from the European Union Horizon 2020 research and innovation programme under grant agreement No. 688227.

7 References

1. J. L. Aguirre, K. Eckert, J. Euzenat, A. Ferrara, W. R. van Hage, L. Hollink, C. Meilicke, A. Nikolov, D. Ritze, F. Scharffe, P. Shvaiko, O. Svab-Zamazal, C. Trojahn, E. Jimenez-Ruiz, B. Cuenca Grau, and B. Zapilko, Results of the ontology alignment evaluation initiative 2012, Proceedings 7th ISWC workshop on ontology matching, OM 2012.
2. American Productivity & Quality Centre, 1993
3. S. Araujo, A.d. Vries, and D. Schwabe, SERIMI Results for OAEI 2011, Proceedings 6th ISWC workshop on ontology matching,OM 2011.
4. I. Bhattacharya, L. Getoor, Entity resolution in graphs, Mining Graph Data, Wiley and Sons 2006.
5. I. F. Cruz, C. Stroe, F. Caimi, A. Fabiani, C. Pesquita, F.M. Couto, ., M. Palmonari, Using AgreementMaker to Align Ontologies for OAEI 2011, Proceedings 6th ISWC workshop on ontology matching, OM 2011.
6. Z. Dragisic, K. Eckert, J. Euzenat, D. Faria, A. Ferrara, R. Granada, V. Ivanova, E. Jimenez-Ruiz, A. Oskar Kempf, P. Lambrix, S. Montanelli, H. Paulheim, D. Ritze, P. Shvaiko, A. Solimando, C. Trojahn, O. Zamaza, and B. Cuenca Grau, Results of the Ontology Alignment Evaluation Initiative 2014, Proceedings 9th ISWC workshop on ontology matching, OM 2014.
7. A. K. Elmagarmid, P. Ipeirotis, and V. Verykios, Duplicate Record Detection: A Survey, IEEE Transactions on Knowledge and Data Engineering, 2007.
8. J. Euzenat, A. Ferrara, Willem Robert van Hage, L. Hollink, C. Meilicke, A. Nikolov, F. Scharffe, P. Shvaiko, H. Stuckenschmidt, O. Svab-Zamazal, and C. Trojahn, Final results of the Ontology Alignment Evaluation Initiative 2011, Proceedings 6th ISWC workshop on ontology matching, OM 2011.
9. J. Euzenat, A. Ferrara, C. Meilicke, J. Pane, F. Schare, P. Shvaiko, H. Stuckenschmidt, O. Svab- Zamazal, V. Svatek, and C. Trojahn, Results of the Ontology Alignment Evaluation Initiative 2010, Proceedings 5th ISWC workshop on ontology matching, OM 2010.

10. J. Euzenat, A. Ferrara, L. Hollink, A. Isaac, C. Joslyn, V. Malaise, C. Meilicken, A. Nikolov, J. Pane, M. Sabou, F. Scharffe, P. Shvaiko, V. S. H. Stuckenschmidt, O. Svab-Zamazal, V. Svatek, C. Trojahn, G. Vouros, and S. Wang, Results of the Ontology Alignment Evaluation Initiative 2009, Proceedings 4th ISWC workshop on ontology matching, OM 2009.
11. A. Ferrara, D. Lorusso, S. Montanelli, and G. Varese, Towards a Benchmark for Instance Matching, Proceedings 3th ISWC workshop on ontology matching, OM 2008.
12. A. Ferrara, S. Montanelli, J. Noessner, and H. Stuckenschmidt, Benchmarking Matching Applications on the Semantic Web, Proceedings of 8th Extended Semantic Web Conference (ESWC 2011), 2011.
13. G. Flouris, D. Manakanatas, H. Kondylakis, D. Plexousakis, G. Antoniou, Ontology Change: Classification and Survey (2008) Knowledge Engineering Review (KER 2008), pages 117-152.
14. I. Fundulaki, N. Martinez, R. Angles, B. Bishop, V. Kotsev, D2.2.2 Data Generator, Technical report, Linked Data Benchmark Council, 2013, Available at http://ldbc.eu/results/deliverables.
15. C. Goutte, and E. Gaussier, A probabilistic interpretation of precision, recall, and F-score, with implication for evaluation, Proceedings of the 27th European Conference on Information Retrieval, ECIR 2005.
16. B. C. Grau, Z. Dragisic, K. Eckert, J. Euzenat, A. Ferrara, R. Granada, V. Ivanova, E. Jimenez-Ruiz, A. O. Kempf, P. Lambrix, A. Nikolov, H. Paulheim, D. Ritze, F. Schare, P. Shvaiko, C. Trojahn, and O. Zamazal, Results of the ontology alignment evaluation initiative 2013, Proceedings 8th ISWC workshop on ontology matching, OM 2013.
17. J. Gray, Benchmark Handbook: For Database and Transaction Processing Systems, Publisher M. Kaufmann, 1991, ISBN:1558601597.
18. O. Hassanzadeh, R. Xin, R.J Miller, A. Kementsietsidis, L. Lim, and M.Wang, Linkage query writer, PVLDB 2009.
19. M.A. Hernandez and S.J. Stolfo, The merge/purge problem for large databases, SIGMOD Rec. Volume 24(2), pages 127–138, 1995.
20. S. Homoceanu, J.-C. Kalo, and W.-T. Balke, Putting Instance Matching to the Test: Is Instance Matching Ready for Reliable Data Linking?, 21st International Symposium on Methodologies for Intelligent Systems (ISMIS), 2014.
21. W. Hu, J. Chen, C. Cheng, and Y. Qu, ObjectCoref & Falcon-AO: Results for OAEI 2010. Proceedings 5th ISWC workshop on ontology matching, OM 2010.
22. J. Huber, T. Sztyler, J. Noessner, and C. Meilicke, CODI: Combinatorial Optimization for Data Integration – Results for OAEI 2011, Proceedings 6th ISWC workshop on ontology matching, OM 2011.
23. Y. R. Jean-Mary, E. P. Shironoshita, M.R. Kabuka, ASMOV: Results for OAEI 2009, Proceedings 4th ISWC workshop on ontology matching, OM 2009.
24. Y. R. Jean-Mary, E. P. Shironoshita, M.R. Kabuka, ASMOV: Results for OAEI 2010 Proceedings 5th ISWC workshop on ontology matching, OM 2010.
25. E. Jimenez-Ruiz, B. Cuenca Grau, and I. Horrocks, LogMap and LogMapLt Results for OAEI 2012, Proceedings 7th ISWC workshop on ontology matching, OM 2012.
26. E. Jimenez-Ruiz, B. Cuenca Grau, and I. Horrocks, LogMap and LogMapLt Results for OAEI 2013, Proceedings 7th ISWC workshop on ontology matching, OM 2013.
27. E. Jimenez-Ruiz, B. Cuenca Grau, W. Xia, A. Solimando, X. Chen, V. Cross, Y. Gong, S. Zhang, and A. Chennai-Thiagarajan, LogMap family results for OAEI 2014, Proceedings 9th ISWC workshop on ontology matching, OM 2014.
28. A. Khiat, and M. Benaissa, InsMT / InsMTL Results for OAEI 2014 Instance Matching, Proceedings 9th ISWC workshop on ontology matching, OM 2014.
29. C. Li, L. Jin, and S. Mehrotra, Supporting efficient record linkage for large data sets using mapping techniques, WWW 2006.
30. M. Nagy, M. Vargas-Vera, P.Stolarski, DSSim Results for OAEI 2009, Proceedings 4th ISWC workshop on ontology matching, OM 2009.
31. R.O. Nambiar, M. Poess, A. Masland, H.R. Taheri, M. Emmerton, F. Carman, and M. Majdalany, TPC Benchmark Roadmap, Selected Topics in Performance Evaluation and Benchmarking, 2012, Volume 7755, pages 1-20, DOI:10.1007/978-3-642-36727-4_1.
32. K. Nguyen and R. Ichise, SLINT+ Results for OAEI 2013 Instance Matching, Proceedings 8th ISWC workshop on ontology matching, OM 2013.

33. X. Niu, S. Rong, Y. Zhang, and H. Wang, Zhishi.links Results for OAEI 2011, Proceedings 6th ISWC workshop on ontology matching, OM 2011.
34. J. Noessner and M. Niepert, CODI: Combinatorial Optimization for Data Integration – Results for OAEI 2010, Proceedings 5th ISWC workshop on ontology matching, OM 2010.
35. J. Noessner, M. Niepert, C. Meilicke, and H. Stuckenschmidt, Leveraging Terminological Structure for Object Reconciliation, Proceedings ESWC 2010.
36. F. Saıs, N. Niraula, N. Pernelle, M.C. Rousset, LN2R – a knowledge based reference reconciliation system: OAEI 2010 Results, Proceedings 5th ISWC workshop on ontology matching, OM 2010.
37. T. Saveta, E. Daskalaki, G. Flouris, I. Fundulaki, M. Herschel, A.-C. Ngonga Ngomo, Pushing the Limits of Instance Matching Systems: A Semantics-Aware Benchmark for Linked Data, WWW 2015.
38. Md. H. Seddiqui, and M. Aono, Anchor-Flood: Results for OAEI 2009, Proceedings 4th ISWC workshop on ontology matching, OM 2009, http://disi.unitn.it/~p2p/OM-2009/oaei09_paper1.pdf
39. C. Shao, L. Hu, J. Li, RiMOM-IM Results for OAEI 2014, Proceedings 9th ISWC workshop on ontology matching, OM 2014.
40. H. Stoermer and N. Rassadko, Results of OKKAM Feature Based Entity Matching Algorithm for Instance Matching Contest of OAEI 2009. Proceedings 4th ISWC workshop on ontology matching (OM 2009). http://ceur-ws.org/Vol-551/oaei09_paper10.pdf
41. A. Taheri and M. Shamsfard, SBUEI: Results for OAEI 2012, Proceedings 7th ISWC workshop on ontology matching, OM 2012.
42. J. Volz, C. Bizer, M. Gaedke, and G. Kobilarov, Discovering and maintaining links on the Web of Data, In Proceedings of the 8th International Semantic Web Conference, ISWC-2009, pages 650–665, Chantilly ,USA.
43. J. Wang, X. Zhang, L. Hou, Y. Zhao, J. Li, Y. Qi, J. Tang, RiMOM Results for OAEI 2010, Proceedings 5th ISWC workshop on ontology matching, OM 2010.
44. K. Zaiss, Instance-Based Ontology Matching and the Evaluation of Matching Systems, Dissertation, Heinrich-Heine-Universiat Dusseldorf, http://docserv.uni-duesseldorf.de/servlets/DerivateServlet/Derivate-18253/DissKatrinZai%C3%9F.pdf
45. K. Zaiss, S. Conrad, and S. A. Vater, Benchmark for Testing Instance-Based Ontology Matching Methods, In KMIS 2010.
46. X. Zhang, Q. Zhong, F. Shi, J. Li, J. Tang, RiMOM Results for OAEI 2009, Proceedings 4th ISWC workshop on ontology matching, OM 2009.
47. Q. Zheng, C. Shao, J. Li, Z. Wang, and L. Hu, RiMOM2013 Results for OAEI 2013, Proceedings 8th ISWC workshop on ontology matching, OM 2013.
48. A. Khiat, M. Benaissa, M-A. Belfedhal, STRIM Results for OAEI 2015 Instance Matching Evaluation, Proceedings 10th ISWC workshop on ontology matching, OM 2015.
49. E. Jimenez-Ruiz, C. Grau, A. Solimando, V. Cross, LogMap family results for OAEI 2015, Proceedings 10th ISWC workshop on ontology matching ,OM 2015.
50. Y. Zhang, J. Li, RiMOM Results for OAEI 2015, Proceedings 10th ISWC workshop on ontology matching, OM 2015.
51. W. Wang, P. Wang, Lily Results for OAEI 2015, Proceedings 10th ISWC workshop on ontology matching, OM 2015.
52. A. Khiat, M. Benaissa, InsMT+ Results for OAEI 2015: Instance Matching, Proceedings 10th ISWC workshop on ontology matching, OM 2015.
53. S. Damak, H. Souid, M. Kachroudi, S. Zghal, EXONA Results for OAEI 2015, Proceedings 10th ISWC workshop on ontology matching, OM 2015.
54. E. Ioannou, N. Rassadko, and Y. Velegrakis, On Generating Benchmark Data for Entity Matching, Journal of Data Semantics, DOI 10.1007/s13740-012-0015-8.
55. C. Böhm, G.de Melo, F. Naumann, and G. Weikum, LINDA: Distributed Web-of-Data-Scale Entity Matching, CIKM'12, October 29–November 2, 2012, Maui, HI, USA.
56. S. Whang, D. Menestrina, G. Koutrika, M. Theobald, H. Garcia-Molina, Entity resolution with iterative blocking, Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, pages 219-232.
57. S. Sarawagi, A. Bhamidipaty, Interactive deduplication using active learning, Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, KDD 2002, pages 269–278.
58. A. Morris, Y. Velegrakis, P. Bouquet, Entity Identification on the Semantic Web, SWAP 2008.

59. M. Hernández, S. Stolfo, Real-world data is dirty: data cleansing and the merge/purge problem, Data Mining Knowl Discov. 1998, Volume 2, pages 9–37.
60. Parag and P. Domingos, Multi-relational record linkage, MRDM workshop co-located with KDD 2004, pages 31–48
61. T. Heath, and C. Bizer, Linked Data: Evolving the Web into a Global Data Space (1st edition), Synthesis Lectures on the Semantic Web: Theory and Technology 2011, Morgan & Claypool, DOI: 10.2200/S00334ED1V01Y201102WBE001
62. T.Saveta, E. Daskalaki, G. Flouris, I. Fundulaki, M. Herschel, A.-C. Ngonga Ngomo, LANCE: Piercing to the Heart of Instance Matching Tool, ISWC 2015, pages 375-391.
63. P. Calado, M. Herschel, L. Leitão, An Overview of XML Duplicate Detection Algorithms, Soft Computing in XML Data Management 2010, Volume 255, pages 193-224.
64. R. Isele, and C. Bizer, Active Learning of Expressive Linkage Rules using Genetic Programming. Web Semantics Journal 2013, Volume 23.
65. R. Isele, and C. Bizer, Learning expressive linkage rules using genetic programming, Proceedings of the VLDB Endowment 2012, Volume 5, pages 1638–1649.
66. A.-C. Ngonga Ngomo, and K. Lyko, EAGLE: Efficient Active Learning of Link Specifications Using Genetic Programming, ESWC 2012, Heraklion, Crete, DOI 10.1007/978-3-642-30284-8_17.
67. O. Hassanzadeh, and M. Consens, Linked movie data base, Proceedings of the WWW2009 Workshop on Linked Data on the Web (LDOW 2009).
68. E. Daskalaki, and D. Plexousakis, OtO Matching System: A Multi-strategy Approach to Instance Matching, Advanced Information Systems Engineering: 24th International Conference, 2012 Gdansk, Poland.
69. M. Cheatham, Z. Dragisic, J. Euzenat, et. Al., Results of the Ontology Alignment Evaluation Initiative 2015, Proceedings 10th ISWC workshop on ontology matching, OM 2015.
70. C. Bizer, T. Heath, and T. Berners-Lee, Linked Data - The Story So Far, International Journal on Semantic Web and Information Systems 2009, Volume 5(3), pages 1-22. DOI:10.4018/jswis.2009081901
71. J. Volz, C. Bizer, M. Gaedke, G. Kobilarov, Silk-A Link Discovery Framework for the Web of Data, 2nd Workshop about Linked Data on the Web (LDOW 2009).
72. J. David, J. Euzenat, F. Scharffe, and C. Trojahn, The Alignment API 4.0, Semantic Web Journal, Volume 2, Issue 1, January 2011 pages 3-10.
73. A.-M. Olteanu-Raimond, S. Mustiere, and A. Ruas, Knowledge formalization for vector data matching using belief theory, Journal of Spatial Information Science, Issue 10, 2015.
74. A. Gray, P. Groth, A. Loizou, S. Askjaer, C. Brenninkmeijer, K. Burger, C. Chichester, C. T. Evelo, C. Goble, L. Harland, S. Pettifer, M. Thompson, A. Waagmeester, and A. J. Williams, Applying Linked Data Approaches to Pharmacology: Architectural Decisions and Implementation, Semantic Web, Volume 5, No. 2, 2014.
75. V. Christophides, V. Efthymiou, K. Stefanidis, Entity Resolution in the Web of Data, Synthesis Lectures on the Semantic Web: Theory and Technology, Morgan & Claypool Publishers, 2015.
76. P. Christen, Data Matching - Concepts and Techniques for Record Linkage,Entity Resolution, and Duplicate Detection, Data-Centric Systems and Applications, Springer, ISBN: 978-3-642-31163-5, 2012 .