Semantic Data Integration

Michelle Cheatham and Catia Pesquita

Abstract The growing volume, variety and complexity of data being collected for scientific purposes presents challenges for data integration. For data to be truly useful, scientists need not only to be able to access it, but also be able to interpret and use it. Doing this requires semantic context. Semantic Data Integration is an active field of research, and this chapter describes the current challenges and how existing approaches are addressing them. The chapter then provides an overview of several active research areas within the semantic data integration field, including interactive and collaborative schema matching, integration of geospatial and biomedical data, and visualization of the data integration process. Finally, the need to move beyond the discovery of simple 1-to-1 equivalence matches to the identification of more complex relationships across datasets is presented and possible first steps in this direction are discussed.

1 An Important Challenge

The world around us is an incredibly complex and interconnected system – one filled with phenomena that cannot be understood in isolation. At the same time, the volume and complexity of the data, theory, and models established to explain these phenomena have led scientists to specialize further and further, to the point where many researchers now spend their entire careers on extremely narrow topics, such as the characteristics of one particular class of star, or the habits of a single species of fish. While such specialization is important to increase humanity's depth of knowledge about many subjects, some of the greatest leaps forward in our understanding come at the intersection of traditional scientific disciplines. These advances require

Michelle Cheatham
DaSe Laboratory, Wright State University, Ohio, USA, e-mail: michelle.cheatham@wright.edu

Catia Pesquita
Universidade de Lisboa, Lisbon, Portugal e-mail: cpesquita@di.fc.ul.pt

the integration of data from many different scientific domains, and this integration must be done in a way that preserves the detail, uncertainty, and above all the *context* of the data involved.

Preserving these properties can be achieved through semantic data integration, a process through which semantically heterogeneous data can be integrated with minimal loss of information. This type of data integration is particularly relevant in domains where data models are diverse and entity properties are heterogeneous. For instance, health information systems, and in particular medical records employ a diversity of vocabularies to describe relevant entities. Health care facilities routinely use different software providers for different aspects of their functioning (outpatient, emergency, surgery, laboratory, billing, etc), each with their own set of vocabularies that many times employ different labels and assign different properties to the same entities. Moreover, the controlled vocabularies many times lack the information necessary to understand the data they describe. For instance, if during an emergency room visit the patient is assigned a primary diagnostic of "Acute upper respiratory infection" using ICD-10, how can we understand that the results of the lab test "Virus identified in Nose by Culture" coded using LOINC, are relevant for the diagnosis? Semantic data integration can provide the means to achieve the meaningful integration of data necessary to support more complex analysis and conclusions.

Unfortunately, semantic data integration is a challenging proposition, particularly for scientific data. Many obstacles stand in the way of synthesizing all of the data about an entity. One of the most obvious of these is accessing the data in the first place. Much of the data underpinning past and present scientific publications is not readily accessible – it exists only in isolated databases, as files on a grad student's computer, or in tables within PDF documents. Moreover, there can be various obstacles to retrieving this data, particularly due to a lack of consistency. For instance, some repositories might be accessible via websites or structured query mechanisms while others require a login and use of secure file transfer or copy protocols. Financial and legal concerns also inhibit data integration. Some data might be stored in proprietary databases or file formats that require expensive software licenses to read, and licenses indicating what users are allowed to do with the data can be missing or restrictive, resulting in legal uncertainty. These types of concerns led to a push towards Linked Open Data, which is described in the next section.

Of course, accessibility is only the first step to semantic data integration. For data to be truly useful, scientists need to be able to interpret and use it after they acquire it. Doing this requires semantic context. By semantic context, we mean the situation in which a term or entity appears. As a simple example, 'chair' would be considered a piece of furniture if it was seen in close proximity to 'couch' and 'table', but as a person if used in conjunction with 'dean' and 'provost'. Similarly, if temperature is included in a dataset that contains entirely Imperial units, it might be assumed to be in Fahrenheit rather than Celsius, particularly if the values correspond to what might be expected (e.g. values near 98 degrees for body temperatures). In relational databases and spreadsheets, semantic context is sometimes lacking because important information about what the various data fields mean and how they relate to one another is often implicit in the names of database tables and column headers, some

of which are incomprehensible to anyone other than the dataset's creator. What is needed is a way to express the semantic connections between different pieces of data in a way that is expressive enough to capture nuanced relationships while at the same time formalized and restrictive enough to allow software as well as humans to make inferences based on the links. Ontologies, described in Section 1.2, have been proposed for this purpose.

Even when data is made accessible by following the Linked Open Data principles and is organized according to a machine-readable ontology, challenges *still* remain. An ontology imposes order on a domain of interest, but order is in the eye of the beholder: if five different publishers of the same type of data were tasked to develop an ontology with which to structure their data, the result would very likely be five different ontologies. One obvious approach is to try to get all data publishers from a domain to agree on a single ontology. This tends to be unfeasible in many instances, for example due to a provider's data causing a logical inconsistency when it is shoe-horned into the agreed upon ontology. A "one ontology to structure them all" approach also conflicts with the inherently distributed paradigm championed by the Semantic Web. An alternative to this strategy is to allow data providers to create or choose whatever ontology best suits their data, and then to establish links that encode how elements of this ontology relate to those within other ontologies.

Establishing semantic links between ontologies and the data sets that they organize can be very difficult, particularly if the datasets are large and complex, as is routinely the case in scientific domains. The fields of ontology alignment and co-reference resolution seek to develop tools and techniques to facilitate the identification of links between datasets. Scientific datasets are particularly challenging to align for several reasons. Perhaps most obviously, such datasets can be extremely large, often over a petabyte of data, which is more than enough to swamp most existing data integration techniques. Additionally, scientific datasets generally have a spatiotemporal aspect, but current alignment algorithms struggle with finding relationships across this type of data because of the variety of ways to express it. For example, spatial regions can be represented by geopolitical entities (whose borders change over time), by the names of nearby points of interest, or by polygons whose points are given via latitude and longitude. Similarly, issues pertaining to measurement resolution, time zones, the international dateline, etc. can confuse the comparison of timestamps of data observations. Furthermore, scientific datasets frequently involve data of very different modalities, from audio recordings of dolphin calls to radar images of storms, to spectroscopy of cellular organisms. Such data is also collected at widely differing scales, from micrometers to kilometers. And oftentimes the data that needs to be integrated is from domains with only a small degree of semantic overlap, as is the case with, for example, one dataset containing information about NSF project awards and another with the salinity values for ocean water collected during oceanography cruises, several of which were funded by NSF.

We have identified a number of challenges in semantic data integration, namely: the accessibility of the data; providing data with semantic context to support its interpretation; and the establishment of meaningful links between data. These challenges are expanded in the following subsections. Section 2 addresses several state

of the art topics in semantic data integration, while section 3 lays out the path forwards in this area.

1.1 Linked Data

Tim Berners-Lee originally envisioned a world wide web that is equally accessible to both humans and computers [5]. Unfortunately, even after several decades we have yet to make this vision a reality. When we look at a webpage today, say, one that presents data about the publications of a group of researchers, we are likely to find that data within an HTML table with columns containing headers such as "Researcher", "Title", "Journal", "Publication Year", etc. If we additionally want to know which researchers are publishing in journals with a high impact factor, we would need to look for the journal's title in the appropriate column of the table, search for the journal's website using a search engine, and find the impact factor on the journal's website by looking for it (hopefully) on the journal's homepage. This is tedious for humans, but extremely difficult for computers. For example, recognizing that the table contains information about researchers' publications and identifying the meaning of each of the columns requires background knowledge and natural language processing, as does realizing that a journal's impact factor is not in the table. Pulling the journal's title out of the HTML table and submitting it to a search engine requires writing code that depends on the format of the table and the API of the search engine, both of which are likely to break if the website or search engine provider makes any changes to those resources. After the query has been made, determining if a particular query result actually contains the impact factor for the journal in question again requires natural language processing. Furthermore, the provider of the data concerning these researchers' publications may not consent to its use for the type of analysis we seek to perform.

Publishing information as linked data alleviates many of these challenges. Linked data builds upon existing web standards such as HTTP, RDF, and URIs to create web pages that are machine-readable and, ideally, machine-understandable. According to Berners-Lee[1], the four rules of linked data are:

1. Use URIs to denote things.
2. Use HTTP URIs so that these things can be referred to and looked up ("dereferenced") by people and user agents.
3. Provide useful information about the thing when its URI is dereferenced, leveraging standards such as RDF and SPARQL.
4. Include links to other related things (using their URIs) when publishing data on the Web.

Linked data is generally published as RDF subject-predicate-object triples. For instance, the following triple indicates an article with the URI cspublications.org/TheSemanticWeb was written by Tim Berners-Lee.

[1] http://www.w3.org/DesignIssues/LinkedData.html

```
<www.w3.org/People/Berners-Lee>
swrc:author
<cspublications.org/TheSemanticWeb> .
```

Similarly, the triples below specify that the article is titled "The Semantic Web", that it was published in 2001, and that the journal it was published in has the URI cspublications.org/ScientificAmerican.

```
<cspublications.org/TheSemanticWeb>
swrc:title
"The Semantic Web"@en .

<cspublications.org/TheSemanticWeb>
swrc:year
"2001"^^xsd:date .

<cspublications.org/TheSemanticWeb>
swrc:journal
<scientificamerican.com> .
```

The expectation is that following the URI scientificamerican.com allows us to learn more information about the journal in which this article was published *even if that information comes from an entirely different data source.*

Publishing data as RDF rather than HTML separates information about data's meaning and context from information about how to format it for presentation. This enables software applications to easily access the data. Additionally, it is possible to express the terms of use for a data set as linked data as well, thus allowing software agents to read and respect these constraints. While this detail is often overlooked, legal issues are often as big of a hindrance to data re-use as technical concerns. Fortunately, addressing this issue is not difficult. Many commonly used licenses have already been encoded in RDF, and datasets can simply add the appropriate triple to refer to them. For example, the triple below indicates that this dataset is available according to the conditions of version 3.0 of the Creative Commons "Share-Alike" license.

```
<cspublications.org/publications.rdf>
cc:license
<http://creativecommons.org/licenses/by-sa/3.0/> .
```

A very large amount of data has already been published as linked open data: according to the most recent survey, there are hundreds of linked datasets, which contain billions of facts about a wide variety of subjects, from music, to biology, to social networks [93]. The website www.linkeddata.org contains pointers to many such datasets. As the linked open data cloud continues to grow, the ability of data providers to contextualize their data by linking it to already-existing data will encourage the creation of more linked data, creating a virtuous feedback loop.

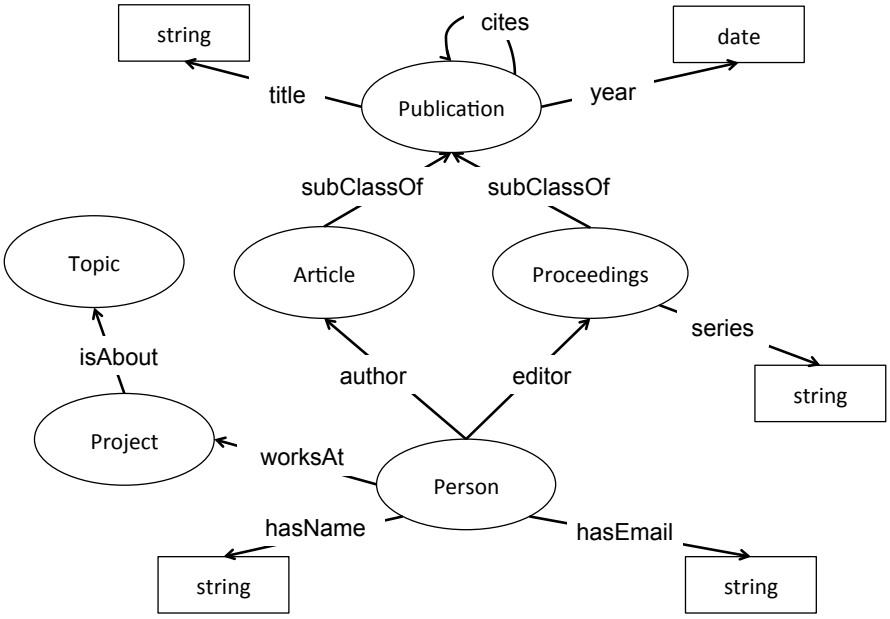
Keeping with our example of medical records, recent work has transformed a clinical datawarehouse into a semantic clinical datawarehouse by applying the Linked Data principles [78]. This enabled clinical data to be integrated with publicly available biomedical repositories, enabling for instance the identification of disease genes.

1.2 Ontologies

Tom Gruber, one of the early voices on knowledge representation (and the creator of Siri), defines an ontology as a "specification of a conceptualization." He elaborates that an ontology defines the concepts and relationships within a domain [35]. Figure 1 shows a snippet of the Semantic Web for Research Communities (SWRC) ontology [105]. The subset of entities shown represent key concepts within the publication domain. The entities shown in ovals, such as *Person* and *Publication* are called classes. A class represents a grouping of objects with similar characteristics. Classes are often arranged in a hierarchy using subclass relationships. For instance, in our example *Article* and *Proceedings* are both subclasses of *Publication* (i.e. every *Article* is a *Publication* but not every *Publication* is an *Article*). An instance (also sometimes called an individual) is a particular object. An instance has a type that is some class within the ontology. For example, an instance of type *Article* may be Weaving the Semantic Web and an instance of type *Person* may be Tim Berners-Lee. This is somewhat analogous to classes and instances of those classes in object-oriented programming languages. Relationships between instances, such as *hasName* and *author*, are called properties. All properties are directed binary relations that map an instance with a type from the domain to something in the range. These are represented as labeled arrows in Figure 1, with the arrow pointing from the domain to the range. Properties that map an instance to another instance (e.g. *editor*, which maps an instance of type *Person* to an instance of type *Proceedings*) are object properties, whereas properties that map an instance to a literal value (e.g. *year*, which maps an instance of type *Publication* to a date value) are datatype properties. Common data types include integers, doubles, strings, and dates. Both object properties and data properties must involve an instance. A third type of property, called an annotation property, can be used to describe relationships between any types of entities (e.g. instances, classes or other properties).

Critically, an ontology should not require an agent, either human or computer, to understand the entity labels in order to leverage the ontology for data publication or consumption. Labels are human-centric, and the underlying goal of the Semantic Web is to put humans and machines on equal footing. Instead of relying on labels to convey meaning, the ontology designer should constrain the possible interpretation of entity labels through judicious use of logical axioms. For example, DBPedia, the linked data version of Wikipedia, contains a property called *hasGender*. The vast majority of uses of this property are to express a person's gender. However, because the domain and range of this property are very vague (i.e. any *Thing* can

Fig. 1 A snippet from the Semantic Web for Resource Community ontology.



have a gender), some of the uses of *hasGender* are very different. For instance, DBPedia asserts that the name "Alexander" *hasGender* "Alexandria" and that a secondary school in England *hasGender* "unisex education." This can cause difficulty for software applications that are attempting to use the *hasGender* property. Misunderstandings can be avoided if the axioms are added to the ontology to constrain the possible meaning of the terms it uses. In this case, the domain of *hasGender* could be changed to be something like *LivingThing*, as shown below.

```
dbpedia:hasGender rdfs:domain dbpedia:LivingThing .
```

Constraints on ontology entities expressed through axioms, together with instance data published relative to those entities, enables a piece of software called a reasoner to infer additional facts that are not actually in the data. For example, if the dataset contained the fact that Tim Berners-Lee wrote "The Semantic Web" and the knowledge base contained an axiom stating that the domain of the property *wrote* is *Person*, a reasoner would be able to infer that Berners-Lee is a person, even if that fact was not explicitly in the knowledge base. A query to return all of the *Persons* in the knowledge base would then correctly include Tim Berners-Lee among the results. This is accomplished without any natural language processing, which can be error-prone in many situations.

Because constraints make the meaning of entity names and relationships more precise, they hold great potential to facilitate accurate data integration. Unfortunately, many existing ontologies do not contain significant numbers of axioms.

However, as we will see in the next section, many existing data and schema integration systems are already capable of leveraging such axioms when they do exist.

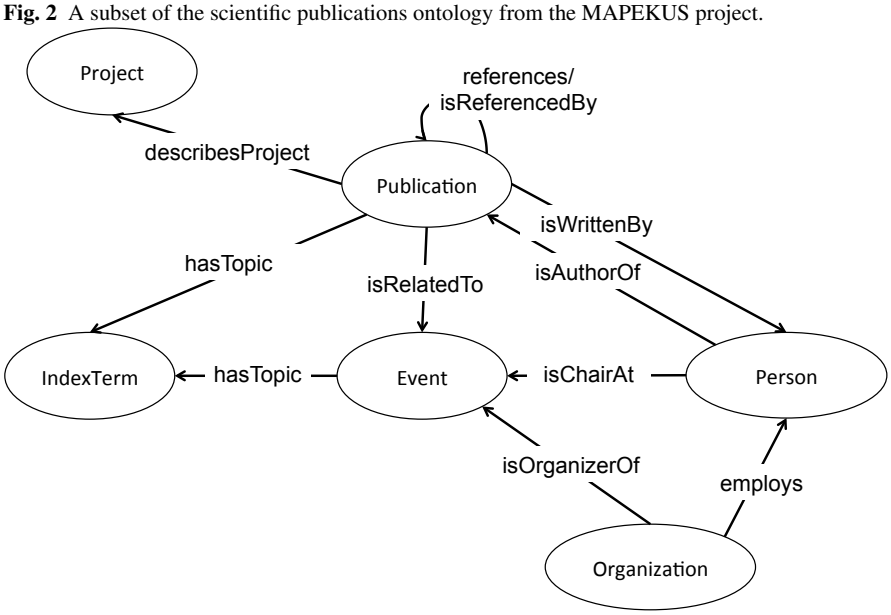
There is a balance to be struck here: too few axioms can lead to many different interpretations of entities, making the ontology less useful; however, too many axioms can constrain the ontology so much that is only applicable in a narrow set of circumstances. For instance, it may seem reasonable to create an axiom that mandates that a *LivingThing* has exactly one gender, this is not the case for some beings. Ontologies are often encoded in the Web Ontology Language (OWL) [69]. Besides property domain and range and cardinality constraints, OWL allows one to state that two entities are equivalent or disjoint, that a property is reflexive, symmetric, transitive, or functional, or that one property is the inverse of another. All of this information: classes, properties, and axioms that restrict their interpretation, is called the schema, or T-box (for terminology), of the ontology. Conversely, the instance data, or A-box (for assertions), contains assertions about individuals using data from the T-box.

A more formal and extensive treatment of ontology design and representation can be found in [41]. Many ontologies exist today. Some of these, such as the Suggested Upper Merged Ontology (SUMO) [82] and the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [32] begin modelling the world at the highest level of abstraction and working towards more detail. The top-level entities in DOLCE, for instance, are *Entity*, *Endurant*, *Perdurant*, and *Abstract*. There are also numerous domain-specific ontologies, such as the Gene Ontology, which models the structure and molecular processes of eukaryotic cells [2], and NASA's Semantic Web for Earth and Environmental Terminology (SWEET) ontology [86]. In the clinical domain, many providers have begun migrating from simple terminologies (such as ICD-10) to more complex ones that have an ontological foundation (such as SNOMED-CT)[14]. Lately many researchers have also begun to publish ontology "snippets," sometimes referred to as ontology design patterns, that model much more constrained topic areas. The website ontologydesignpatterns.org currently has dozens of ontology snippets, including models of a Hazardous Situation, a Species Habitat, and a Chess Game.

1.3 Ontology and Data Alignment

While the amount of linked data available on the Semantic Web has grown continually for more than a decade, the links between different datasets have not gown at the same rate. These links provide the context that makes the data more useful. The fields of ontology and data alignment attempt to discover links between datasets in an automatic or semi-automatic way. Ontology alignment systems tend to focus on finding relationships between schema-level entities, while co-reference resolution systems attempt to identify cases in which the same individual is referred to via different URIs.

Fig. 2 A subset of the scientific publications ontology from the MAPEKUS project.



1.3.1 Ontology Alignment

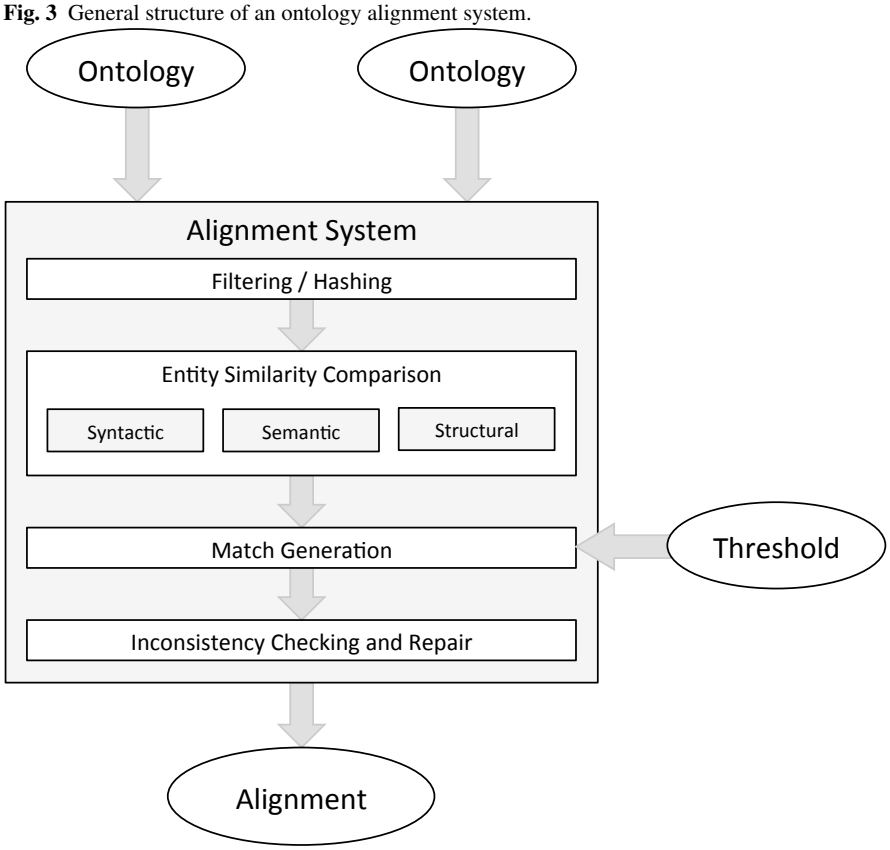
Engineering new ontologies is not a deterministic process – many design decisions must be made, and the designers' backgrounds and the application they are targeting will influence their decisions in different ways. The end result is that even two ontologies that represent the same domain will not be the same. They may use synonyms for the same concept or the same word for different concepts, they may be at different levels of abstraction, they may not include all of the same concepts, and they may not even be in the same language. And this is in the best case. In real-world datasets there are often problems with missing information, inconsistent use of the T-box when describing individuals, and logically inconsistent axioms. The goal of ontology alignment is to determine when an entity in one ontology is semantically related to an entity in another ontology (for a comprehensive discussion of ontology alignment, including a formal definition, see [24]).

An alignment algorithm takes as input two ontologies and produces a set of matches consisting of a URI specifying one entity from each ontology, a relationship, and an optional confidence value that is generally in the range of 0 to 1, inclusive. For example, Figure 2 shows a second ontology describing publications. This ontology was created as part of the MAPEKUS project.[2] An alignment system given the ontologies in Figures 1 and 2 might produce matches including:

```
mapekus:Person, swrc:Person, =, 1.0
```

[2] http://mapekus.fiit.stuba.sk

Fig. 3 General structure of an ontology alignment system.




```
mapekus:Publication, swrc:Publication, =, 0.9
mapekus:references, swrc:cites, =, 0.8
mapekus:IndexTerm, swrc:Topic, <, 0.6
```

Note that matches can relate any type of entities, including classes (e.g. Person) and properties (e.g. references, cites). Additionally, a match can indicate a variety of relationships. The most common are to state that two things are equivalent (e.g. all mapekus:Publications are swrc:Publications and all swrc:Publications are mapekus:Publications) or that one subsumes the other (e.g. all mapekus:IndexTerms are swrc:Topics but all swrc:Topics are not mapekus:IndexTerms). Though not necessary, in practice alignments are often interpreted under the closed world assumption, in the sense that any entity pairs not mentioned in an alignment are assumed to have no relationship.

Many alignment systems share a common general organization, shown in Figure 3. Because ontologies can contain millions of entities, it is often infeasible to compare every entity in one ontology to every entity in the other. Therefore, alignment

systems sometimes employ a filtering or hashing step to determine which entities to compare [20, 40]. Alignment systems typically use a combination of three different approaches to evaluate entity similarity: syntactic, semantic, and structural similarity metrics. Entity similarity is related to how much two entities have in common; it can be thought of as a measure of the degree one class, property, or individual could be used in place of another. Syntactic metrics compare entities from each of the ontologies to be aligned based on strings associated with the entities. The strings are generally the entity label, but can also include comments or other annotations of the entity. Semantic similarity metrics attempt to use the meanings of entity labels rather than their spellings. External resources such as thesauri, dictionaries, encyclopedias, and web search engines are often used to calculate semantic similarity [46, 107]. Structural techniques consider the neighborhoods of two entities when determining their similarity. For instance, two entities with the same superclass that share some common instances are considered more similar than entities that do not have these things in common. Graph matching techniques are often used for this [31, 18]. An alignment system may use zero or more of each type of similarity metric. The values from multiple approaches may be combined to form a single measure of similarity, or they may be used in a serial fashion to filter potential matches down to the most likely candidates. At some point, a final list of related entities is generated, frequently by including any matches with a confidence (similarity) value higher than some threshold. Additionally, alignment systems may use some form of inconsistency checking and repair after the matching process in order to ensure a merged ontology produced using the alignment is logically consistent [62, 90, 83].

Each year since 2005, the Ontology Alignment Evaluation Initiative has invited researchers to compare the performance of their alignment systems on a set of benchmark tasks. Current alignment systems have become very proficient at finding 1-to-1 equivalence relationships between classes and instances (the type of matches contained within the benchmarks). In fact, the top-performing systems now attain a 0.75 f-measure on one of the OAEI test sets that is designed to reflect real-world matching tasks [10]. This is nearing the level of consensus that humans familiar with ontology design have for alignment tasks involving this test set [12]. Unfortunately, the performance on finding relationships between properties is not nearly as good as that for classes and instances [13]. Additionally, there is some evidence that most of the accuracy of existing alignment systems is due to basic string similarity measures [11], which raises some concern that further gains may be more difficult to achieve.

1.3.2 Coreference Resolution

Coreference resolution algorithms attempt to determine when the same instance (i.e. individual) is referred to using different URIs. Note that because the term “ontology alignment” can either refer to aligning an entire ontology (the T-box and the A-box) or just the T-box, this section uses the term “schema alignment system” to refer to something that attempts to map only the T-box of an ontology.

Coreference resolution differs from schema alignment in several ways. One key difference is that the relationships sought by coreference resolution algorithms are only 1-to-1 equivalences: two individuals are either the same or distinct, whereas schema elements involve *sets* of individuals and can therefore have all of the traditional relationships that exist between sets, including subsumption, disjointness, and partial overlap. Another important contrast between coreference resolution and schema alignment is that the A-box of an ontology is often an order of magnitude larger than the T-box. This makes efficiency concerns even more important for coreference resolution algorithms than for schema alignment systems. Another distinction is that, while there is interplay in both directions between coreference resolution and schema alignment, it can be argued that coreferences generally place more constraints on schema alignments than the other way around. This is because many existing schema alignment systems employ some extensional comparators in the mapping process, i.e. they determine the likelihood that two schema elements are related by the degree of overlap between their instances. For example, if it is determined that $data1 : Tim\_Berners\_Lee$ in one dataset is equivalent to $data2 : TimothyLee$ in another dataset, and $data1 : Tim\_Berners\_Lee$ is a $data1 : Scientist$ while $data2 : TimothyLee$ is a $data2 : ComputerProgrammer$, a schema alignment algorithm is more likely to conclude that the classes $data1 : Scientist$ and $data2 : ComputerProgrammer$ are related in some way (i.e. that they are not disjoint). This is done for classes in [22] and for properties in [36]. Because equality cannot be defined extensionally for individuals, questions about what it fundamentally means for two things to be identical tend to arise in coreference resolution research [37]. Coreference resolution can be thought of as data de-duplication, which has been an area of research for decades. For instance, there has been extensive research regarding recognizing the duplicate records, stretching back to at least 1969 [30]. Many of the approaches currently employed to resolve coreferences on the Semantic Web are adapted from techniques that were established decades ago in database integration systems. A good survey of such techniques can be found in [21].

Of course, there are obviously differences between databases and linked data published according to an ontology. The most obvious of these is that databases operate under the "closed world" assumption, meaning that if something is not present in the database, it is assumed not to exist. In contrast, the Semantic Web uses an open world assumption. Also, as Castano and his colleagues point out in [9], the structure of a linked dataset can differ greatly from a relational database that represents the same domain due to the expressiveness of ontology specification languages such as OWL compared to database table definition and column constraint capabilities. The more complex relationships expressible in ontologies may convey implicit knowledge that can be inferred by a reasoner. This additional information is not generally available when integrating two databases. In terms of focus specifically on integrating linked data, schema alignment has more of a research history than coreference resolution. For example, the annual Ontology Alignment Evaluation Initiative has

existed since 2005, but it has only had a track dedicated to evaluating performance on coreference resolution tasks since 2009.[3]

The general decisions made by the designer of a coreference resolution system are: what instances to compare, how to compare them, and how to determine if the result of that comparison implies that the two instances are equivalent.

As mentioned previously, there are generally many more instances in a dataset than schema entities. As a result, it is not considered feasible to compare every instance in one dataset to every instance in the other in order to determine if they are the same. Instead, some method of deciding whether two instances are "close enough" that they are worth comparing must be established. The choice of this method reflects a trade-off between recall and utility, i.e. an overly zealous filtering algorithm may miss some equivalences, while a conservative filtering approach may cause the system to take a long time to generate results.

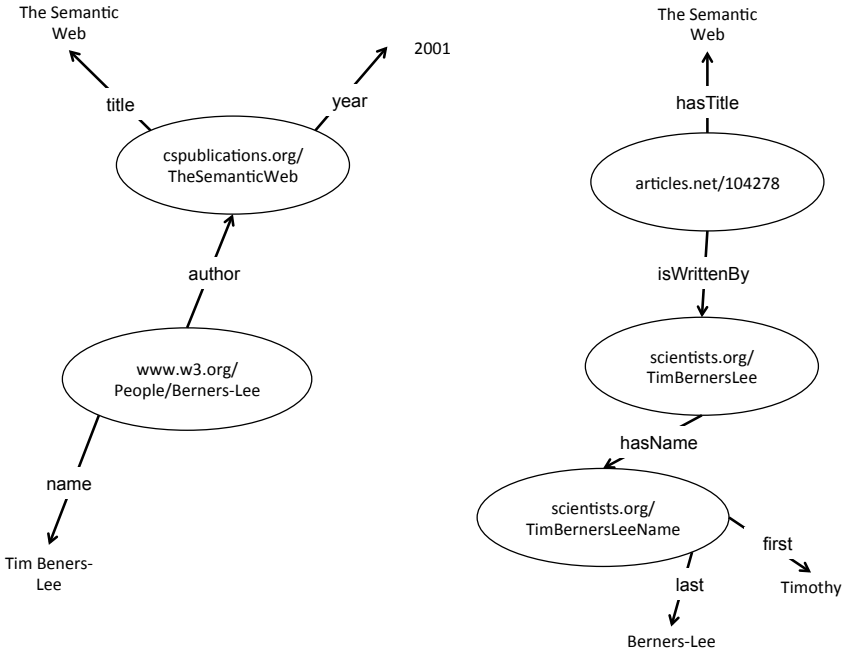
If the filtering step has decided that two instances are close enough to warrant further scrutiny, the algorithm will compare them based on a selection of features. In most current coreference resolution systems, these features are either property values alone or property values together with property names. There is also a question of how deep within an instance's semantic neighborhood to go when extracting features. For an example, in Figure 4 there are two instances, from different datasets. A coreference resolution algorithm could either compare the values only (e.g. compare "The Semantic Web" and "Tim Berners-Lee" from the instance on the left with "The Semantic Web", "Timothy" and "Berners-Lee" from the instance on the right), or it could compare both the values and the property names. In the latter case, for example, rather than an exact match on the title, the similarity would be slightly less than perfect because the property for the title of an article is called "title" in the left instance and "hasTitle" in the right.

Regardless of what features are compared, the most common method for comparing them is via string similarity metrics. This is because even when a property is a non-string datatype, such as a date or URL, it is often expressed as a string in datasets. Different string metrics are employed, primarily depending on the length of the strings to be compared. A survey of string metrics commonly used by these systems is provided in [11]. Note that global metrics, which based decision on characteristics of the overall distribution of values in the dataset, are not generally feasible due to the size of the A-box, but they may be employed based on a random selection of the A-box. A decision must also be made on how much to weight each feature. Various methods have been proposed for this, including both supervised [87] and unsupervised [73] machine learning approaches.

Finally, the coreference resolution system must take the outcome of a comparison of two instances and make a decision on whether or not those instances are equivalent. This is often done by specifying thresholds and other parameters of the algorithm. This is a somewhat neglected area of research – it is common for researchers to report that these values were "determined empirically" for the particular datasets being aligned. Among the small amount of work on this topic is an explo-

[3] http://oaei.ontologymatching.org

Fig. 4 A potential coreference



ration by Paulheim and his colleagues of using interactive techniques to configure the threshold by asking a user targeted questions regarding the validity of potential matches with confidence values on either side of the current threshold and updating it accordingly [81].

2 Current State-of-the-Art

In the face of the performance plateau on current alignment benchmarks, many researchers have created innovative new alignment techniques that focus on various aspects or subproblems under the general umbrella of semantic data integration. This section explores a selection of this current work.

2.1 Interactive and Collaborative Approaches

While the performance of automated alignment systems is becoming quite good for certain types of mapping tasks, in practice no existing system generates alignments that are completely correct. Alignments tend to either lack some correct mappings,

contain some incorrect mappings, or both. As a result, there is significant ongoing research on alignment systems that allow users to contribute their knowledge and expertise to the mapping process. These systems exist on a spectrum ranging from entirely manual approaches to semi-automated techniques methods that ask humans to chime in only when the automated system is unable to make a definitive decision. Because entirely manual alignment is feasible only for small datasets, most current research in this area focuses on semi-automated approaches. In contrast to fully manual approaches, semi-automated systems interact with the user(s) only intermittently, and then attempt to leverage this human-supplied knowledge to improve the scope and quality of the alignment. Interactive systems of this type differ in terms of what questions they ask users and how they make use of the responses. In addition to being judged on precision (how many of the mappings they generate are correct) and recall (how many of the correct mappings they generate), these systems are also generally gauged based on how much effort they require from the humans interacting with the system, in terms of the number of questions they must answer and the difficulty inherent in coming up with each answer.

An obvious approach to leveraging user input in an alignment system is to first use an automated approach to generate an alignment and then ask the user to verify (a subset of) the matches that were created. Invalid matches can then be pruned from the final alignment. ServOMBI implements this approach [52]. Clearly, this approach is capable of improving precision, particularly if the user is asked about matches that the automated system is most in doubt about, perhaps evidenced by confidence values near the threshold. However, since the user involvement comes at the end of the alignment process, this method cannot improve recall over what the automatic component achieves. On the other hand, this approach is suitable for adding an interactive component to *any* matching system, because it only require the end product of the tool.

A variation of this technique is to move the interactive questioning to within the matching process rather than conducting it at after the fact. This can have a very large impact on both precision and recall because most alignment system will only match an entity to one other entity, so any match that is incorrect may be doubly bad by causing the correct match to be missed. Furthermore, when a match is found, many algorithms use a technique called similarity flooding [63] to thoroughly explore the neighborhoods of both of the entities involved, sometimes with relaxed match criteria on the theory that things related to equivalent entities are more likely to also be equivalent. The general idea behind similarity flooding is that two entities that are connected to similar things are most likely similar themselves. For example, assume there is a class in one ontology called Man that is a subclass of Human and the domain of a property called hasAge, and there is class in a second ontology called Male that is a subclass of Person and the domain of a property called hasYears. If Human and Person and hasAge and hasYears have already been found to be highly similar, similarity flooding will increase the similarity value between Man and Male. When using similarity flooding, an incorrect decision during the matching process can cascade to cause a host of other incorrect decisions.

Several interactive systems attempt to ask the user for guidance at critical decision points (and only these points) during the mapping process in order to maximize their accuracy. One such system is LogMap 2, which arranges all potential mappings that it is unsure about in partial order based on the value of similarity metrics employed by the system. Starting at the beginning of this list, the system asks the user whether each potential mapping is valid, until the end of the list is reached or the user halts the process. When a user approves a match for an entity, any other potential matches for that entity are discarded. Any matches suggested by the algorithm that are logically inconsistent with the user-approved match are also discarded. Experimental results indicate that this interactive technique improves performance as long as the user responds accurately at least 70% of the time [50].

The AgreementMaker alignment system takes a different approach to integrating user feedback into the alignment process. Rather than choosing which mappings to ask the user about based on a single or aggregate similarity score, AgreementMaker asks about potential mappings on which its constituent matchers disagree. Specifically, the system uses four syntactic matchers, a structural matcher, and a semantic matcher (according to the terminology presented in Section 1.3). If the matchers are divided on a particular mapping, the user is asked to provide a decision. This decision is then used to update the certainty values on other potential mappings with the same pattern of matcher agreement/disagreement, and this update is considered when deciding what question to ask the user next. In this manner the system is able to significantly improve the alignment accuracy while asking relatively few questions overall [16].

The OAEI established an interactive matching track in 2013. Participating systems can make a programmatic call to an "oracle" that consists of a pair of URIs and a relation (currently limited to either equivalence or subsumption) and receive a true or false reply indicating whether or not the relation holds between the two entities[4]. Beginning in 2015, the track included tests with an imperfect oracle, i.e. the oracle was either correct all of the time, correct 90% of the time, 80% of the time, or 70% of the time. Also in 2015, the alignment tasks were expanded from ontologies related to conference organization to other tasks, including mapping larger biomedical ontologies. Four alignment systems have participated in this track each year (though not always the same four), and the results have improved annually. In 2013, the average performance of the system when interactions were possible was actually 3% *worse* in terms of f-measure than in a fully automatic setting. The best system performed 8% better, for an f-measure of 0.72. Two years later, the average performance was 20% better with interactions, and the best system performed 11% better, for an f-measure of 0.818. The number of requests to the oracle required to achieve these results has also decreased markedly since the first year [34, 19].

While the introduction of the interactive alignment track to the OAEI has clearly been productive in terms of encouraging research in this area and driving the improvement of interactive alignment systems, this is not a perfect approach to evaluating such systems. In particular, the type of queries that systems can pose to the

[4] http://oaei.ontologymatching.org/2013/interactive/index.html

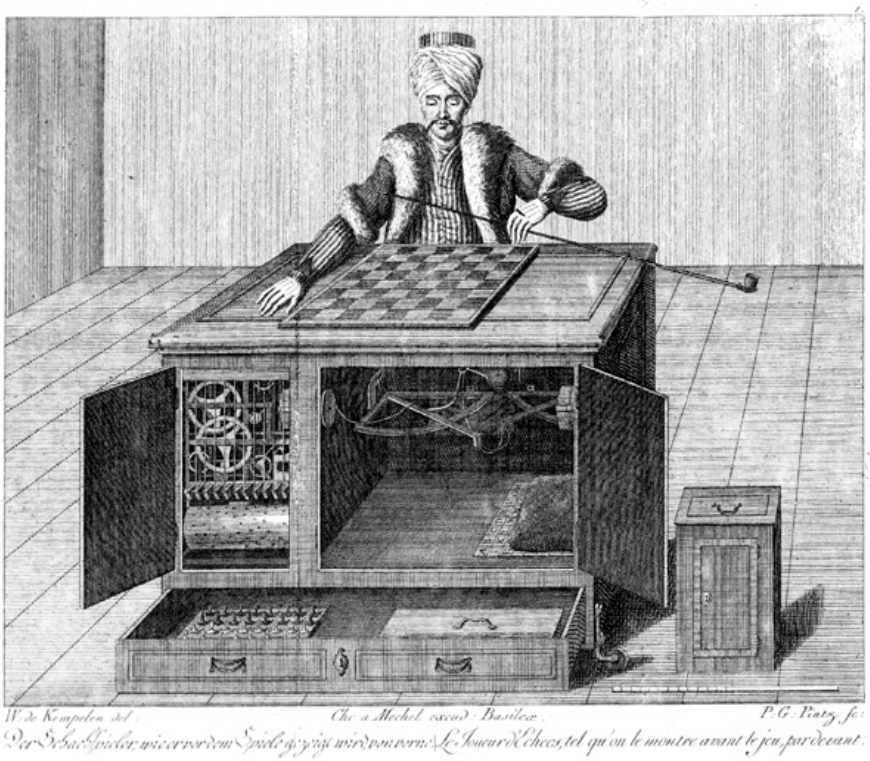
oracle is limited to asking yes or no questions regarding a particular match. One can easily think of many other types of questions that would be worthwhile, such as how certain a user is that a particular match is correct or how a user arrived at their decision on the correctness of a match. As Paulheim and his colleagues point out, the questions that are asked of a user and the way in which they are asked impact the size of the burden placed on the user. For instance, asking a user *what* relationship holds between two entities (or what other entity is equivalent to a given entity) is likely a more difficult question for a user to answer than *whether* a particular relationship holds [81]. Others have conducted more extensive evaluations of interactive matching systems for the bioinformatics domain, including usability, time requirements, and user satisfaction [55]. This type of user study is time consuming however, and it has not been performed in a standardized way for a large number of general matching systems.

Of course, the issue with the above methods is that ontology engineers and domain experts are generally very busy people, and they may not have much time to devote to manual or semi-automated data integration projects. As a result, some ontology alignment researchers have turned to generic large-scale crowdsourcing platforms, such as Amazon's Mechanical Turk.

Amazon publicly released Mechanical Turk in 2005. It is named for a famous chess-playing "automaton" from the 1700s. The automaton actually concealed a person inside who manipulated magnets to move the chess pieces. Similarly, Amazons Mechanical Turk is based on the idea that some tasks remain very difficult for computers but are easily solved by humans. Mechanical Turk therefore provides a way to submit these types of problems, either through a web interface or programmatically using a variety of programming languages, to Amazons servers, where anyone with an account can solve the problem. In general, this person is compensated with a small sum of money, often just a cent or two. The solution can then be easily retrieved for further processing, again either manually or programmatically. While there are few restrictions on the type of problems that can be submitted to Mechanical Turk, they tend towards relatively simple tasks such as identifying the subject of an image, retrieving the contents of receipts, business cards, old books, or other documents that are challenging for OCR software, transcribing the contents of audio recordings, etc. As of 2010, 47% of Mechanical Turk workers, called Turkers, were from the United States while 34% were from India. Most are relatively young (born after 1980), female, and have a Bachelors degree [44]. It is possible for individuals asking questions via Mechanical Turk (called Requesters) to impose qualifications on the Turkers who answer them. For instance, Requesters can specify that a person lives in a particular geographic area, has answered a given number of previous questions, has had a given percentage of their previous answers judged to be of high quality, or pass a test provided by the Requester. In addition, Requesters have the option to refuse to pay a Turker if they judge the Turkers answers to be of poor quality.

A group of researchers from Stanford University has recently published several papers on using Mechanical Turk to verify relationships within biomedical ontologies [67, 68, 66, 76]. Their results show that general purpose crowdsourcing plat-

Fig. 5 An engraving of the original Mechanical Turk by Karl Gottlieb von Windisch. The Mechanical Turk was a famous chess-playing "automoton" from the 1700s that was actually operated by a human nestled inside the cabinet. It is the namesake of Amazon's Mechanical Turk platform, which allows developers to harness a plethora of human workers to solve tasks that remain difficult for computers.



forms can be used to answer questions about the relationships between ontology entities, even if the domain modeled by the ontology is quite scientific. Mechanical Turk has also been used to validate existing alignments [12, 13]. Additionally, there is an interactive alignment system called CrowdMap that uses Mechanical Turk to generate alignments between two ontologies [92].

All of these systems reported good results, though some were hampered by scammers that answered questions randomly or with some other time-saving strategy in order to maximize their profit-to-effort ratio [92, 76]. Additionally, there is some indication that the way in which questions about potential mappings are asked may have a large impact on the utility of the general crowdsourcing approach. In the work described in [13], questions about potential equivalent properties were presented in the following form: "Does property label A mean the same thing as property label B?" Respondents were instructed to choose one of four options: they mean the same thing, one is a more general or more specific term than the other, they are related

in some other way, or there is no relation. In order to provide some context, the questions provided information about the domain and range of each property and up to five examples of instances with values for each property. The initial results showed that the general response on these nuanced verification questions were not very reliable. In all cases, the researchers responded by qualifying Turkers based on their performance on a small simple set of questions regarding possible mappings in order to gain access to the full range of tasks. This strategy proved reasonably effective.

Another strategy for dealing with scammers is to take money out of the equation. Instead of paying individuals to contribute to an alignment, the work can be packaged as a game that the user plays in order to earn a good score. This is the approach taken by the game SpotTheLink [109]. The game involves teaming up two random players, presents them both with an entity from the source ontology along with a description and image of the entity, if available, and asking them to collaboratively find an entity in the target ontology that is related and how it is related (equivalent, subclass, or superclass). Players only get points when they both agree. This game was built on top of OntoGame, which is a Java plug-in framework that provides services such as user login, randomly pairing users, and keeping score [99].

Another possible approach for avoiding scammers when crowdsourcing data alignments is to require people who wish to make sure of that data resource to first answer questions that contribute to its growth and quality (or to improve a separate, related data resource). This is the approach suggested by [60]. McCann and his colleagues point out that for this technique to work, the data resource must either not be available with no strings attached elsewhere, or it must be of a higher quality than alternative sources for the information. Additionally, the users must only be asked a limited number of questions, and they must have some control over when they will answer those questions.

2.2 Visualizing the Data Integration Process

Involving humans in the data integration process, as described in the previous section, requires some type of interface to enable those individuals to understand what questions are being asked of them, be aware of the context necessary to accurately answer those questions, and understand the implications of their answers. These needs lead to a set of requirements for data alignment interfaces. Several researchers have worked to enumerate these requirements. One of the first steps towards this was work by Falconer [25], which was then built upon by several others. There are several recurring points of emphasis in this work, which are addressed throughout the remainder of this section.

2.2.1 Presentation of Candidate Mappings

Users need a way to quickly see the mappings suggested by the automated mapping component, why they have been selected, and which mappings have been validated, refused, or remain to be considered. Because the number of candidate mappings may be quite large, there needs to be some way to manage them, for instance by clustering or by filter-based searching [33]. Logically organizing these potential mappings is key, because often validating one mapping enables a user to validate many additional mappings that have a similar underlying rationale [97].

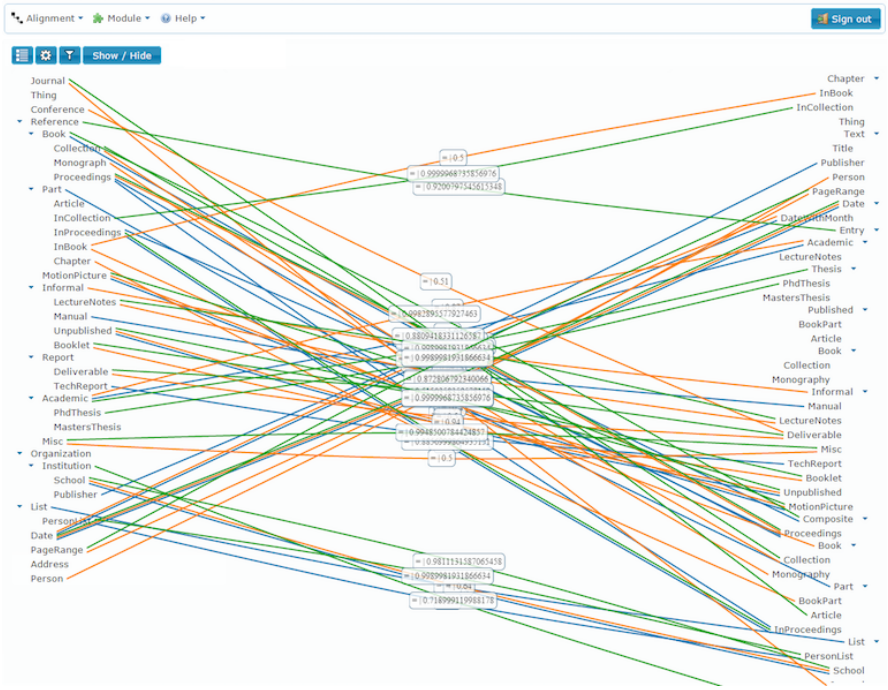
VOAR is an application for working with ontology alignments that illustrates several of these concepts [96]. VOAR does not have a built-in automated alignment algorithm, but rather can call any such algorithm that implements a standard interface. Users can also load in multiple existing alignments and merge them or compute the intersection (i.e. only mappings that occur in all alignments are kept). When the user is validating and/or creating mappings, the class hierarchies are shown on either side of the interface, and a table of mappings, including the associated confidence value, is in the middle. Clicking on a mapping will highlight the relevant entities in the trees. VOAR also has another mode that allows users to visualize the alignment as a whole (Figure 6). This shows which entities are involved in mappings, indicating areas within each ontology that are densely or sparsely related to the other. This view lists all entities from each ontology along the sides of the interface and connects related entities with a line. To assist the user in following these correspondences when there are many mappings, related entities and the line connecting them are color-coded.

BioMixer, a tool designed to visualize mappings among more than two biomedical ontologies at once, takes a different approach to showing an overview of all mappings within an alignment [111]. This tool provides several different ways to view mappings, including a matrix view in which the terms from each ontology are listed in alphabetical order along the top and left side of the matrix, and a colored square within the matrix indicates a mapping. This highlights clusters of mappings for similarly-named terms (which often serve as anchor points upon which to build more complete alignments). A different view enables the user to drill down into the part of the ontology surrounding a particular entity. This part of the tool uses displays the entity, its neighbors in the ontology, and its relationship to entities in the other ontology as a graph. The user is able to understand an entity's context and potentially identify missing mappings.

2.2.2 Presentation of the Ontologies

Presentation of the ontologies is also important. Most individuals will confirm the validity of a mapping based upon the neighbors surrounding the entities in both ontologies, and once they validate one mapping they are often able to confirm several others involving related entities [25]. For this reason, it is helpful to enable quick navigation between the list of potential mappings and the relevant entities in both

Fig. 6 A screenshot of the VOAR ontology alignment visualizaton tool. The entities from each ontology that are involved in mappings are listed along the edges, and the lines between them indicate equivalence relations.



ontologies. Further, by showing the entities in both ontologies that are involved in one or more potential mapping, the user can focus on these areas first and thereby improve their efficiency. Users also need to be able to add mappings that were missing from the list of suggestions. This necessitates the ability to navigate across both ontologies at varying levels of abstraction, including drilling down to view the details of any entity, and filtering on a wide range of criteria [33]. This is the area of visualization research that has received the most attention, as we will see later in this section.

A tree-based presentation of an ontology is only capable of displaying hierarchical information, such as the class hierarchy within an ontology. Other types of information contained in an ontology's axioms, such as property domain and range and cardinality constraints, are lost in a tree representation. This is particularly problematic for aligning the properties within ontologies [13]. To avoid this, many ontology visualization applications use a graph to display the ontologies. Kow and his colleagues take this approach in [53]. In this tool, candidate mappings are shown in a list that allows the user to accept or reject them. As with BioMixer, selecting a mapping in the list displays the relevant entities and their neighbors within the ontologies in a detailed graph view. While limiting the graph to the nearby neighbors

of the entities in question keeps them from becoming cluttered, the overall context of the ontology is lost. A way to navigate across the ontologies at a high level of abstraction is needed. Kow's application enables this through a global view they call an "information landscape". This view shows all of the entities from both ontologies (color-coded red or green according to which ontology they belong), with similar concepts placed near one another. Clumps of entities are labeled with terms describing the group. The user can select areas of interest that seem likely to contain related entities, which automatically filters the mappings shown in the list. This method of filtering allows users to systematically explore an ontology at a high level of detail without losing track of the big picture.

2.2.3 Demonstration of Mapping Implications

The individual mappings that together make up an alignment are not independent of one another; there are some cases in which only one of two mappings can possibly be true. Sometimes mappings will result in a class being *unsatisfiable*, meaning that it is not possible for an instance to meet all of the requirements to be a member of that class. In other cases, one or more mappings, when taken together, may lead to an unintended and undesired inference. Visualization systems need to convey the implications of a potential mapping to users. Potential ways to achieve this include highlighting the relationships between mappings, allowing the user to temporarily add a mapping and observing its impact [45], and providing a mechanism for the user to indicate that a particular mapping is uncertain or subjective [25].

ContentMap is one application that attempts to provide details about the implications of mappings to users [49]. ContentMap uses several existing ontology alignment algorithms to generate a set of candidate mappings, which users can either accept or reject. The system then computes the *logical difference* between the entailments before and after the mappings are applied. Entailments that ContentMap suspects may be unintended (because they hold in the merged ontology but not in the individual ontologies) are presented to the user, who can indicate which ones are in fact undesired. The system then runs a mapping repair algorithm that attempts to remove the minimum number of axioms to alleviate the unintended entailments while preserving the entailments the user indicated were valid. Because computing the logical difference is quite difficult over expressive ontologies (there is no algorithm to do this for OWL 2 or OWLDL), ContentMap focuses only on alignments consisting entirely of subclass, equivalence, and disjointness relations.

Another data integration tool, MappingAssistant, takes a different approach to providing feedback to the user regarding the implications of a mapping [103]. MappingAssistant is based on the intuition that domain experts, who are not necessarily familiar with formal modeling constructs like ontologies or with logical entailments, nevertheless have a detailed understanding of the instance data. The system therefore conveys the implication of schema-level mappings to the user by selecting (using a clustering algorithm) a set of instances affected by the mapping rule and displaying them. Users can then indicate any instances that have been incor-

rectly classified, and the application will present a series of questions (expressed in natural language) to the user in order to determine which mapping has led to the incorrect classification.

2.2.4 Scalable to Large Ontologies

Assisting user in aligning ontologies with a large number of entities or many axioms constraining the relationships between entities can be particularly challenging for a visual interface. A myriad of issues come into play. For instance, many alignment systems display the class hierarchies as tree structures on either side of an interface, with lines between the trees indicating potential mappings. If the number of potential mappings is very large, users can quickly be overwhelmed by such a presentation [25]. Other interfaces display the ontologies and potential mappings in a graph, but again, the size and complexity of the graph grows with that of the ontologies, and the user can find this representation unwieldy [57]. Ivanova's work on requirements for large-scale ontology alignment make it clear that whatever strategy is used to represent the ontologies and potential mappings, it must not only "scale" visually, but also computationally – users cannot be made to wait after each interaction for the interface to update [45]. Users also cannot always be expected to align large ontologies by themselves in one setting. Consequently, tools should allow users to save their progress and to divide up the alignment task among multiple contributors [33].

One approach to dealing with the overwhelming complexity of a graph-based global view of an ontology is through employing a clustering algorithm to raise the level of abstraction at which the ontology is shown in the graph. Even though AlViz, developed in 2006, is older than many of the other visualization applications discussed in this section, we use it as an example of this approach since new work in the field still frequently cites it as a source of inspiration [57]. AlViz shows each ontology in its own graph and uses color coding of nodes to indicate areas of similarity and difference between the two. A slider on the side of each graph controls the level of abstraction. The size of a node gives an indication of the number of entities aggregated within it. Nodes are aggregated based on the similarity metric of an integrated alignment algorithm. Small world graphs such as those used by AlViz typically use a spring layout, which is known to have a cubic time complexity. Still, the original AlViz system was capable of displaying ontologies with 1000 entities and respond to user interactions without a lag.

The alignment system AML employs a different strategy to handle the complexity of graph-based views [84]. AML combines both ontologies and mappings in a single graph. However, instead of showing the full ontologies and alignment, it shows only a subgraph centered on a selected mapping, for which the neighborhood of classes and mappings between them can be shown up to five edges distance. This allows for a better understanding of neighboring mappings than typical tree-based visualizations, and is particularly relevant in the visualization of biomedical ontologies where multiple inheritance and the existence of different kinds of rela-

tions between classes is common. Users can then navigate the list of mappings to visualize the different subgraphs, mark mappings as correct or incorrect, and add new mappings.

The alignment system SAMBO implements several features to assist users in aligning large ontologies [56]. In particular, the tool allows users to cease calculations of mapping suggestions at any time and begin to approve or reject any of the mappings that have been suggested at that point. The user can also save their work and resume it later. Each saved session contains information about how many mappings have been validated, how many remain, and the last date the user worked on the alignment. Users also have the option to preprocess data in between sessions, to save time when they resume their work. The preprocessing step uses the class hierarchies of both ontologies to partition the ontologies into "mappable parts" such that the set of entities from the first ontology that are in a partition are highly likely to be mapped to an element from the second ontology that is also in the same partition. As a result, similarity metrics do not have to be applied between all pairs of entites, but only between those in the same partition. Furthermore, users can focus their attention on one partition rather than being overwhelmed by the entire ontologies. While the authors do not state this, these partitions might also be a way to divide the mapping validation task among multiple people.

Several researchers who have considered the requirements for a visualization system intended to facilitate data integration have also mentioned the need to allow users to annotate a particular mapping with its rationale and additional metadata as required for the particular use case, and a mechanism to debate or vote on mappings [33, 45]. Unfortunately, this information is not collected in as anything other than free text by most data integration tools. This issue is the subject of Section 3.3.

2.3 Integrating Geospatial Data

Many data sets, from user reviews of hotels and restaurants, to oceanographic measurements, to economic data, have a spatial component. Integrating data based on location can lead to important cross-domain insights relevant to a particular region. However, as mentioned in the introduction, spatial data is particularly difficult to align. There are many reasons for this. Of course, spatial data sets have all of the normal issues related to schema. For instance, one data set may refer to a building's location using the property "Address" while another one may use two properties: "City" and "State." There are also challenges specifically related to spatial data because of the many ways to express it. For example, location can be specified with an address, with latitude and longitude, or in reference to a nearby point of interest. There are also many ways to express a spatial region. For example, spatial regions can be represented by geopolitical entities (whose borders change over time), by polygons whose points are given via latitude and longitude, or by a point and a radius. Another issue is that spatial data is collected at widely different scales and

with different resolution and coverage, which raise quality concerns when integrating several data resources. Furthermore, for both technical and social reasons, many spatial data sets are stored in relational databases, as images, or as vector data. These different representation formats necessitate different approaches to integration. This section surveys some of the current research related to integrating geospatial data.

2.3.1 Representing Geospatial Data

Many geospatial datasets have been published on the Semantic Web. Two of the largest and most well-known are GeoNames and LinkedGeoData. GeoNames has information about over 8 million geographic entities from around the world, including place name, coordinates, elevation and population. Much of the data was originally imported from official public sources, but it can now be edited by individuals. GeoNames is organized according to a relatively simple ontology involving nine features and 645 feature codes.[5] LinkedGeoData is based on the data from the OpenStreetMap project. OpenStreetMap's goal is to build a geographic knowledge base from the ground up, by allowing contributors to use aerial imagery and GPS devices to create and verify information.[6] GeoNames and LinkedGeoData are interlinked with one another and with DBPedia. Other geospatial datasets are region-specific. For instance, the UK Ordnance Survey, Great Britain's national mapping agency, has published gazetteer and administrative boundary information as linked data as part of the "Making Public Data Public" initiative within that country.[7] Publishing geospatial data according to the linked open data principles allows it to be integrated more easily. Consequently, useful applications that levereage linked geospatial data have begun to emerge, including for disaster management [80] and wildlife monitoring [54].

Much of the geospatial data that is currently available is represented using the Geography Markup Language (GML).[8] GML was created by the Open Geospatial Consortium (OGC) and has become an ISO standard. The schema is centered on the class *Feature*. A *Feature* can have a *Geometry*, such as point, line, polygon, curve or surface. GML also supports specifying a *Feature*'s location, using a coordinate reference system. Another way to represent geospatial data is using GeoSPARQL, which is an RDF vocabulary and a set of extensions to SPARQL to support spatial queries.[9] The GeoSPARQL vocabulary currently leverages many elements of GML, together with well-known text (WKT), to represent vector geometry objects on a map; simple feature, which contains spatial relationships such as intersects and within; region connected calculus (RCC8), to represent relationships between

[5] http://www.geonames.org

[6] linkedgeodata.org

[7] http://data.ordnancesurvey.co.uk

[8] http://www.opengeospatial.org/standards/gml

[9] http://www.opengeospatial.org/standards/geosparql

two regions such as tangential or partially overlapping; and DE-9IM, to represent topology.

Other OGC standards are closely related, including Keyhole Markup Language (KML) to specify how display geographic information on a map or other visualization.[10] Several ontologies have also been developed to represent higher-level concepts with a strong spatial element, such as a "Semantic Trajectory" to describe movement through space [43] and "Stimulus-Sensor-Observation" to model observations of phenomena collected at a particular time and place [47].

2.3.2 Querying Geospatial Data

One way to query geographical data is by using the Web Feature Service (WFS).[11] This OGC standard can return results in GML or as shapefiles, a vector format dictated by the Environmental Systems Research Institute (Esri) and used by the popular GIS software platform ArcGIS. Examples of WFS queries are: "return the name of all towns that are along this line" and "return the name of all mountains within this bounding box." GeoSPARQL supports these types of queries as well, but over the full RDF vocabulary expressed in the GeoSPARQL standard.

Many datasets with a geospatial component are stored in relational databases rather than published as linked data. There are several reasons for this: databases are often an established part of a scientist's workflow, existing data analysis tools may require the data to be stored in a database, or collection systems may automatically publish to a database. However, there is still a need to incorporate semantics into queries of this data. One approach to achieving this is to allow users of the data to query it based on an ontology, and then to translate those queries into the language required by the database. This area of research is sometimes known as ontology-based data access (OBDA). An example of this is the work of Zhao and his colleages, described in [114]. Their system uses an RDF ontology to enable semantic queries on a standard relational database containing geospatial data. Their approach is to translate queries based on their RDF ontology to WFS queries on the underlying database. There are some limitations to this approach. In particular, the ontology needs to be created manually for each dataset and application domain, and a table in the database can only map to one class in the ontology. Also, the ontology is specified in RDF rather than OWL in order to simplify the query rewriting. Later work supports more complex queries while extending this semantic querying capability to multiple geospatial datasets stored as GML [110].

[10] http://www.opengeospatial.org/standards/kml

[11] http://www.opengeospatial.org/standards/wfs

2.3.3 Coreference Resolution and Alignment

Regardless of how it is stored, coreference resolution of geospatial data is particularly challenging due to noise and difference in coverage and resolution. Even extracting appropriate features on which to base the data integration task can be difficult, though recent work in that area by projects such as Brainwash show promise [1]. Once features have been collected, it has been common to use machine learning approaches to weight features of geospatial datasets such as location name, location type (e.g. mountain, desert, island) and coordinates, which are then used in standard classifiers such as SVMs and linear regression models [95]. More recent work takes a very similar approach. For instance, McKenzie and his colleagues integrate data from FourSquare and Yelp using a weighted combination of location name, location category (e.g. seafood restaurant, casual dining), geographic location, and an unstructured textual description. Their results were impressive, with 98% of places of interests correctly aligned. An interesting element of their work was that a system based only on geographic coordinates was only 57% accurate. They indicate that this may be due to inaccuracies of mobile devices using GPS or wireless to calculate position [61]. Similarly, Li et. al. merged point of interest data from Baidu (a search engine) and Sina (a social networking site) based on name, category and location. Their weighting method was based on the entropy of the various attributes. This method was chosen because the attribute values exhibited a non-linear similarity metric characteristic [58].

Aligning the schema of geospatial datasets can actually be somewhat easier than in the more general case. While geospatial datasets often have different labels for the same properties (e.g. “state” versus “administrative district”), labels of geospatial properties are selected from a smaller domain than are general properties. Furthermore, geospatial datasets typically have a large A-box, making extensional matching techniques useful when aligning the T-Box. For example, if one dataset has a property called “CensusCount” and another has a property called “Population,” values for particular cities contained in both datasets allow an automated alignment system to conclude that these properties are likely equivalent.

2.3.4 Assessing Quality

When integrating data from multiple sources, quality becomes an important concern. This is particularly important for geospatial datasets. Whenever any continuous entity is measured, there will be inaccuracies inherent in the measured values due to limitations of coverage and resolution. Typical quality indicators include lineage, positional accuracy, attribute accuracy, logical consistency and completeness [7]. Additionally, interviews with consumers of geospatial data indicate the importance of metadata when assessing quality, such as the reputation of the data provider and the number of citations. Unfortunately, the majority of geospatial datasets do not have *any* quality information associated with them [59].

There have been some efforts to automatically derive quality information for geospatial datasets that lack it. For instance, work by Thakkar et. al. is targeted toward situations in which many geographical datasets are being integrated, and only some of them have associated quality metrics. Quality is based upon completeness and positional accuracy. Completeness is the percentage of features that the source contains information on. Thakkar gives the following example: if there are 100 hospitals in an area and a source contains 25 of them, then the source is 25% complete. Positional accuracy is determined based on the number of features within a given bound, i.e. the location of 40% of the hospitals is accurate to within 10 meters. They automatically assess the quality of an unknown data source by identifying a source with known quality that provides the same attribute and has at least some instances in common. The quality of the new source is then based upon comparison of a sample of the common subset. Once that source's quality has been evaluated, it can then be checked for overlap with any other sources within the system whose quality was previously unknown and used to assess their quality [108].

There has also been considerable research on assessing the quality of volunteered geographic information. For example, in 2010 Mooney and his colleagues evaluated OpenStreetMap data from 11 European countries based on sampling density and metadata tagging and their utility in correctly representing the shape of features such as lakes and forests (and found the quality to be quite low overall) [65]. Ballatore and Zipf take a higher level approach and consider the quality of the schemas used to organize the geospatial data. They argue that maintaining conceptual quality is straightforward when data producers and consumers are all colleagues, but that quality suffers when producers and consumers don't know one another, as is the case with volunteered geographic information. Their framework includes accuracy (existence of entities, categories and attributes necessary to accurately describe the geospatial features of interest), granularity (ability to describe the features at the desired level of abstraction), completeness (ability to describe all the features of interest), consistency (similar features are described with similar classes and properties), compliance (agreement of this schema with another one), and richness (number and variety of dimensions with which to describe a feature) [4].

2.4 Integrating Biomedical Data

Massive amounts of multimodal and diverse data are currently being generated by researchers, hospitals and mobile devices around the world, and their combined analysis presents unique opportunities for healthcare, science and society. The data can range from molecular to phenotypic, behavioral to clinical, individual to population, genetic to environmental. Maximizing the potential of this data through its meaningful integration can enable new directions for research, for instance discovering new drugs or determining the factors causing human disease.

Biomedical Big Data goes well beyond the recognized challenges in handling large volumes of data or large numbers of data sources, and presents specific chal-

lenges pertaining to the heterogeneity and complexity of data as well as to the complexity of its subsequent analysis. The availability of over 500 open biomedical ontologies in BioPortal [77] and dozens of biomedical datasets as Linked Open Data represents a unique opportunity to integrate clinical and biomedical data.

A first step in supporting the semantic integration of biomedical data is by making it available as Linked Data and having the entities and relationships referred to in the Linked Data defined according to ontologies. Exposing datasets as Linked Data enables the interconnection of distinct data items across providers, facilitating the integration of high volume and heterogeneous data sources (i.e. experimental data, libraries, databases) and also provides an aggregated view of biomedical data in a way that is machine interpretable and reusable, as well as semantically-enriched via links to ontologies. These links support the classification of data according to the concepts defined by a given ontology, which provides a perspective on the data.

The same data can be described under different ontologies, which provide different perspectives. For instance, patient data described under the Disease Ontology [94] will provide a view of the diseases and disorders affecting the patient, while the same patient data described under the Symptom Ontology [3], will provide information about signs and symptoms but not the underlying causes. However, even when focusing on a single perspective, lets say diseases, the multitude of ontologies and controlled vocabularies currently in use to describe them impedes the seamless integration of data. Multiple ontologies for the same or closely related domains can and do exist, due to several reasons ranging from disconnected development, to development focusing on particular applications. This is especially true in the biomedical domain where there are for instance nine ontologies that describe neurological disease, ranging from highly specific ontologies covering a single disease (e.g. epilepsy, Alzheimers) to ontologies covering all kinds of human disease, such as the Disease Ontology. This results in several ontologies describing the same concepts under slightly different models.

These challenges are being addressed at several levels by the application of Semantic Web technologies.

2.4.1 Linked Biomedical Data

There have been several efforts to expose biomedical data as Linked Data, with the aim of providing structured and integrated access to the massive amounts of biomedical data distributed in numerous repositories [8, 88, 89, 64, 112]. This is a challenging endeavor since each biomedical dataset has a unique structure and vocabulary.

The Bio2RDF project [8] defines a set of simple conventions that allow the creation of a knowledge space of RDF documents as Linked Data. It uses a mashup approach that leverages normalized URIs and a common ontology, integrating publicly available data from some of the most popular databases in bioinformatics. However, few biomedical repositories expose their data as RDF, so the project built a toolbox to generate RDF files from locally stored databases or directly from HTML docu-

ments accessed via http requests. Although Bio2RDF facilitates the integration of heterogeneous datasets, achieving a complete syntactic and semantic normalization is not yet a reality. One of the reasons behind this, is that Linked Data serialized as RDF does not support the complex formal semantics that allow the inference of relationships between data items from heterogeneous datasets. This prevents a fuller integration, namely at the level of relations or types.

Another related project is Neurocommons [88], which is dedicated to creating an open source knowledge management platform for biological research and is specifically working on an open knowledge base of annotations to biomedical abstracts (in RDF) and the integration of major neuroscience databases into the annotation graph. Neurocommons is grounded in Semantic Web technologies, integrating OWL ontologies, RDF and SPARQL endpoints.

BioPortal also publishes their ontologies as RDF [89]. This dataset contains over 190 million triples, representing both metadata and content of the ontologies. It also publishes over 10 million mappings between ontologies, generated via both manual and automatic methods.

2.4.2 Cross-references and mappings

To promote and facilitate integration, some biomedical ontologies already provide cross-references to equivalent or related concepts in other ontologies. These can be used not only for integrating ontologies, but also for the integration of data items described with the ontologies.

One notable effort in increasing the interoperability of biomedical ontologies has been the creation of logical definitions [71]. This is an initiative of the Open Biomedical Ontologies Foundry [101], a collective of ontology developers whose mission is to develop a family of interoperable ontologies that are both logically well-formed and scientifically accurate. One issue of biomedical ontologies is that although almost all classes have a textual definition, which can be interpreted by a human user, this is not accessible to a computer without sophisticated natural language processing. Therefore, efforts have been made to transform these definitions into a computable form as a set of logical definitions. Such logical definitions facilitate automated access to an ontology and complement text definitions. They could also potentially be used to reason over an ontology or to automatically derive relationships between classes, thus contributing to the integration of different ontologies. Developing and maintaining these computable definitions requires a lot of manual labor, leading to the development of strategies to partially automate the process [70]. More recently, the definition of composite relations as class expressions has also been explored through the alignment of classes in biomedical ontologies with foundational classes in a top-level ontology [42].

Another relevant resource is the UMLS [6], which provides a mapping structure among over 100 controlled vocabularies in the biomedical sciences, covering over 1 million biomedical concepts and 5 million concept names. UMLS is not originally available as RDF, but BioPortal through its UMLS2RDF project [89] has trans-

formed the UMLS MySQL release into RDF triples. BioPortal also provides a set of 3.1 million mappings between the terms in UMLS vocabularies.

While these projects are contributing to the utility of biomedical data on the Semantic Web by establishing links between datasets, in many cases such mappings are unavailable, giving rise to the need to derive them automatically using ontology matching techniques.

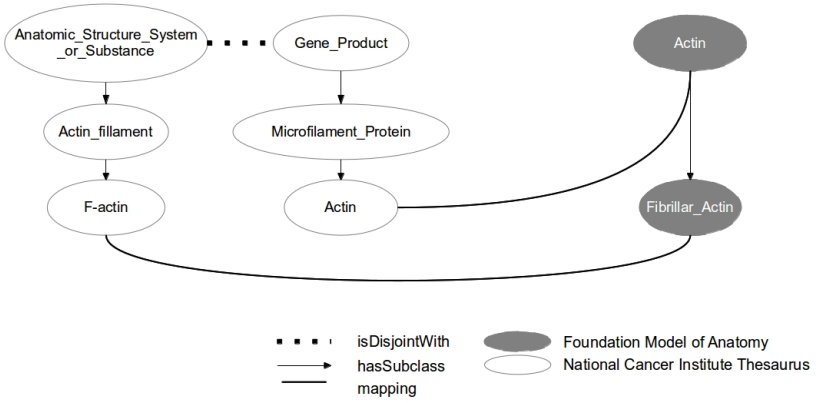
2.4.3 Ontology Matching for Biomedical Ontologies

The specific characteristics of biomedical ontologies need to be taken into account when developing tools and techniques to explore them:

- large size: biomedical ontologies commonly have thousands of classes, which can represent both a computational and a visualization challenge. In Bioportal there are over fifty ontologies with more than ten thousand classes.
- complex vocabulary: biomedical ontologies typically encode several names for the same class, including one main label and several synonyms of different kinds (e.g., narrow synonym, broad synonym). This represents a challenge for lexical matchers, which need to be able to handle multiple labels and at different closeness degrees.
- multiple related domains with different points of view: it is fairly common to have the same biomedical domain being described according to different models. This can cause logical incoherences when two ontologies with different models are integrated. For instance, Figure 7 illustrates a logical incoherence caused by two mappings between the National Cancer Institute Thesaurus Ontology and the Foundation Model of Anatomy Ontology. The logical incoherence arises because upon integration, *Fibrilar_Actin* becomes a subclass of both *Anatomic_Structure_System_or_Substance* and *Gene_Product*, which are disjoint classes. Solving these incoherences is far from trivial [83]
- rich axioms: biomedical ontologies have been evolving towards greater semantic richness establishing different kinds of relations between classes (e.g., regulates, adjacent to, participate in) and complex axioms (e.g., 'human patient and (has Age some float $[>= 8]$) participant in 'WHO standard treatment for human brucellosis in adults and children eight years of age and older). Typically, ontology matching systems either just focus on taxonomic relations, or do not differentiate between different types of relations. This is especially relevant for structural matchers.

The relevance of ontology matching for biomedical ontologies has been recognized by the community, and the Ontology Alignment Evaluation Initiative [10] currently contains two tracks dedicated to biomedical ontologies: the anatomy track, and the large biomedical ontologies track. Both tracks illustrate the above mentioned challenges, and in the last few years some ontology matching systems have been quite successful in addressing the challenges of matching biomedical ontologies.

Fig. 7 Alignment between portions of the National Cancer Institute Thesaurus Ontology and the Foundation Model of Anatomy Ontology illustrating a logical incoherence.



Regarding size and scalability, a well studied challenge [51], systems have had to evolve from the traditional encoding of the matching problem as a matrix of similarities to more efficient data structures. For instance, AgreementMaker [15], a system successfully used for matching biomedical ontologies [17] struggled with more than a few thousand classes, which inspired the development of AgreementMakerLight (AML) [29], based on more scalable data structures that can handle over 100 thousand classes in an ontology.

The complexity of the vocabulary in biomedical ontologies has also been addressed by several systems, which combine several matchers capable of exploiting different string and lexical similarities [15, 29, 72]. Some systems take this further by leveraging external resources as a source of synonyms. For instance, AML includes specifically tailored approaches to exploiting the rich synonyms of biomedical ontologies [85], that can use cross-references to extend synonyms and apply lexical techniques to derive new synonyms. It also contains high performing strategies to automatically select and utilize external ontologies as background knowledge [28]. Other systems use pre-defined external resources. For instance LogMap [48] makes use of external lexicons to derive spelling variants, and GOMMA [39] can explore external ontologies for synonyms.

Regarding the ability to handle logical incoherence, few ontology systems currently support it, and even fewer at a scale conforming to biomedical ontologies' typical size. The most basic approach is filters out any mappings that violate a series of semantic rules (e.g. [72]). More sophisticated approaches rely on automated procedures which are able to identify the mappings involved in the logical incoherence and select which ones to remove to achieve coherence. Both AML [91] and LogMap support this, and their application to the mappings in BioPortal has proven successful [27].

Finally, the ability to process rich axioms is still not a focus of current ontology matching systems. Despite the growing complexity of biomedical ontologies, systems are still lacking in this respect. A recent effort in this area has been the development of a compound matching approach [79], that is able to capture equivalence mappings between one class from one ontology and an expression involving two classes from two other ontologies, forming a ternary mapping. For example, HP : *aorticStenosis* is equivalent to an FMA : *aorta* that is $PATO$: *constricted* This novel matching paradigm needs to be able to handle the much larger search space (three ontologies instead of two) and be able to not only identify the equivalence mapping but compose the expression as well.

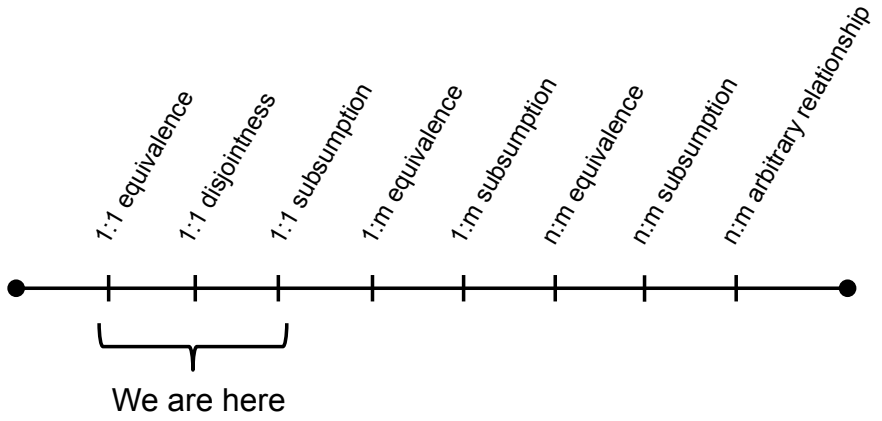
3 The Path Forward

Work such as that described in Section 2 has already begun to pay dividends. Techniques for semantic data integration have reached a level of maturity that has allowed them to be incorporated into commercial and open source tools from organizations such as Oracle, Apache and Microsoft. For example, Oracle 11g provides support for storing data as RDF, querying data from disparate sources seamlessly via SPARQL, and performing reasoning via SWRL-like rules [113]. Various aspects of the performance of these industry systems is being evaluated by the academic community [98] as well as utilized for domain-specific research [100, 26]. Additionally, these systems are being used by other commercial enterprises for applications ranging from entertainment media management to national intelligence [113]. However, many challenges clearly stand in the way of accurate and efficient data integration in the general case. This section considers some research threads that could potentially lead to future breakthroughs in semantic data integration.

3.1 Moving Beyond 1-to-1 Equivalence Mappings

Ideally, alignment systems should be able to uncover any entity relationships across two ontologies that can exist within a single ontology. Such relationships have a wide range of complexity, as shown in Figure 8. The simplest type of relationship is 1-to-1 equivalence or disjointness of two entities (i.e. all instances of A are also instances of B or an instance of A is definitely not an instance of B). Assume that we have two ontologies, ont1 and ont2,that model a university. The relation ont1:Course = ont2:Class is an example of a 1-to-1 equivalence match, while ont1:registeredFor disjoint ont2:Teaching (i.e. someone cannot both register to take a course and teach it) is an example of a 1-to-1 disjointness relationship. The next complexity level is subsumption relationships, i.e. that an entity in one ontology is a subclass or superclass of an entity in another ontology. ont1:Faculty $\subset$ ont2:Employee is an example of this. Even harder to find are 1-to-many equivalence or subsumption re-

Fig. 8 Complexity range of entity relationships between ontologies.



lationships between entities, such as the union of ont2:AsstProf, ont2:AssocProf, and ont2:FullProf is equivalent to ont1:Professor. This causes a complexity problem. To find 1-to-1 relationships, an exhaustive search needs to compare every entity in the first ontology to every entity in the second ontology, which may be feasible for small ontologies. To find 1-to-m relationships an exhaustive approach would need to compare each entity in the first ontology to all possible combinations of m entities in the second ontology, which is not generally possible. Finding arbitrary n-to-m relationships is the most complex alignment task. By "arbitrary," we mean any type of relationship, not restricted to equivalence, disjointness, or subsumption. An example of this might be that a ont1:Professor with an ont1:hasRank value of "Assistant" is equivalent to an ont2:AsstProf. Such complex relationships would need to be expressed as logical rules or axioms.

Nearly all existing alignment systems fall at the simplest end of this scale. A few systems, including BLOOMS [46] and PARIS [104], are capable of finding subsumption relationships across ontologies. CSR [102] and TaxoMap [38] attempt to find 1-to-m equivalence and subsumption relationships. There has also been some preliminary explorations into identifying ternary compound mappings across biomedical ontologies [79]. In general though, most research activity in the field of ontology alignment remains focused on finding 1-to-1 equivalence relations between ontologies.

As mentioned previously, the performance of current alignment systems on tasks that focus on the identification of 1-to-1 equivalence relations has become quite good. However, alignment research may be in danger of becoming stuck in a "local maximum", and it might be time to make a concerted push towards discovering more complex semantic relationships. The computational complexity of this task makes it very unlikely that existing approaches to mapping discovery can be used to discover complex relationships. It is possible that existing algorithms from the fields of data mining and machine learning might be applied for this purpose, but

significant effort will likely be required to identify appropriate techniques and tailor them for this application.

3.2 Advancing Alignment Evaluation

The Ontology Alignment Evaluation Initiative (OAEI) is now over a decade old, and it has been extremely successful by many different measures: participation, accuracy, and the variety of problems handled by alignment systems have all increased, while runtimes have decreased [23]. The OAEI benchmarks have become *the* standard for evaluating general-purpose (and in some cases domain-specific or problem-specific) alignment systems. In fact, you would be hard-pressed to find a publication on an ontology alignment system in the last ten years that did *not* use these benchmarks. They allow researchers to measure their systems performance on different types of matching problems in a way that is considered valid by most reviewers for publication. They also enable comparison of a new systems performance to that of other alignment systems without the need to obtain and run the other systems. This is a huge boon for ontology alignment research.

Of course, benchmarks need to evolve over time in order to remain relevant. The OAEI suite of benchmarks contains eight tracks that test alignment systems in a range of contexts in which they might be used, but currently none of these tracks contain any complex relationships. In addition, the details of some of the test sets have led to the incorporation of behaviors in some alignment systems that may not be optimal. For instance, in several OAEI tracks an entity can be involved in at most one match, which may not be realistic for some real-world datasets. Similarly, entities are only matched to other entities of the same type in some tracks, e.g. classes to classes, instances to instances, etc. This is not realistic in all cases, particularly when the decision of when to represent something as an instance versus a class is not always clear cut.

As a specific example of the limitations of current alignment benchmarks, consider the case of property matching. Performance of current alignment systems on matching classes is on average three times better than on matching properties [13]. Researchers have suggested various reasons for this, including that the parts-of-speech used in property names differs from that used for class names [106], that taxonomies of properties are much less common that those of classes [106, 75], and that properties are reified in different cases than are classes [75]. Perhaps uncoincidentally, only one of the eight OAEI tracks involves any matches between properties, and those matches make up a small percentage of the total. This is a big cause for concern because many influential real-world linked datasets, such as DBPedia (the linked data version of Wikipedia) and YAGO, are strongly property-centric.

The OAEI is a community-driven effort, and its organizers are very willing to incorporate new benchmarks into the evaluation. Establishing new benchmarks is far from easy, however. Some of the existing OAEI testsets are synthetic, which means that the reference alignments are completely accurate. Synthetic benchmarks may

not accurately reflect the type of challenges alignment systems face "in the wild" though. On the other hand, several of the OAEI testsets are based on real-world ontologies. The reference alignments were developed primarily by three graduate students, with feedback from "Consensus Workshops" held after each OAEI for several years. This method of benchmark creation is very resource-intensive and is therefore only feasible for small ontologies. In order to create the large reference alignments comprised of complex mappings needed to drive the field forward, more scalable methods of benchmark construction need to be explored.

As mentioned in Section 2.1, some researchers have turned to crowdsourcing platforms such as Amazon's Mechanical Turk to facilitate scalable ontology alignment. It may be possible to use such platforms to generate alignment benchmarks as well. However, there is some well-founded skepticism regarding the trustworthiness of crowdsourced alignment benchmarks. In particular, there is concern that the results may be very sensitive to how the question is asked. For instance, how much context from each ontology are users provided with? Are they able to rush through the work, or does some mechanism force or encourage them to give due consideration to each potential match? Does the best method for question presentation depend on the characteristics of the ontologies being aligned? How does the amount of monetary payment and bonuses affect performance? These are all very important questions, and if researchers in the ontology alignment field are going to accept work on complex alignments evaluated via crowdsourcing or a crowdsourced benchmark as valid, they must be addressed.

Another obstacle is that when creating an ontology alignment benchmark, one has to start from somewhere. It is too resource intensive to try to verify every potential relationship across all entities in both ontologies, even in the 1-to-1 equivalence case. This is a complete non-starter for complex alignments. The standard approach to this problem is to employ an ensemble of existing high-performing alignment systems to align the ontologies and then manually refine the results to create the reference alignment for the benchmark [92]. Unfortunately, this approach is not feasible for the creation of some types of benchmarks due to the lack of current alignment systems that attempt to the type of relationships required. It is something of a chicken-and-egg problem. For instance, it is very difficult to create a benchmark containing complex relationships when there are no alignment systems capable of identifying such relationships that can be used to create the benchmark. Solving this problem is an open area of research in this field.

3.3 Contextualizing Alignments

Data and schema integration is done for some *purpose*, and the mappings that should be included in a particular alignment are a function of that purpose. For example, alignments can be done to support distributed querying, or they can be used for logical reasoning. The characteristics for each type of alignment are different. For querying, recall (i.e. returning the relevant results) is generally an important aspect

of the application using the alignment. This means that alignments to support query-centric applications arguably need to err on the side of expressing relationships that generally hold, even if some outliers lead those relationships to cause logical inconsistencies that confound reasoners. Conversely, applications that intend to employ a reasoner on the integrated data cannot generally make use of an alignment that contains any logical inconsistencies

Current alignment systems support these different use cases to some degree. For instance, AgreementMakerLight has the ability to detect and repair mapping inconsistencies, but it is also capable of leaving these in place [91]. However, there is currently no way to express in an alignment, after it has been created, which use case was targeted. It can be argued that the user of the alignment can simply check to see if it contains any inconsistencies, but this assumes that the user is employing the alignment, which may be only for the T-box of the ontologies, to the same A-box that was in place when the alignment was created, and that the A-box hasn't changed over time. Additionally, there are many applications that an alignment may have been created for beyond the simple query-versus-reasoning divide. A way to express the situations in which an alignment applies is needed.

In addition to a mechanism for expressing the applicability of an alignment, a standardized way to represent the rationale behind individual mappings within an alignment is also needed. For example, if an alignment asserts that Johnathan Smith who works at IBM is the same person as John Smith who organized the ABC Conference, it is helpful for the consumers of that alignment to know how this was determined. Perhaps in this case it is known that IBM was the primary sponsor of the ABC Conference and that John is a common nickname for Johnathan. Making this type of provenance available at the level of individual mappings is important for enabling consumers to make informed decisions about how to use the mappings within an alignment and how much confidence to place in them. While this need has been noted by both researchers and practitioners [74, 97], how best to represent this information is not currently clear.

By far the most common manner in which ontology alignment and coreference resolution systems represent their results is the Alignment API format. In this representation, each relationship between two ontologies (cell) is a "first-class citizen. In particular, each cell contains the URI of the entity that is the source of the relationship, the URI of the target entity, the relationship that holds between them (equality, subsumption, etc.), and the strength of that relationship (a decimal value between 0 and 1, inclusive). However, it also seems obvious that storing provenance information regarding who created a coreference and when would also be extremely useful. The creators of the Alignment API intended for such provenance information to be stored at the alignment level rather than at the level of individual cells. This is not well suited to projects in which coreferences may come from a variety of sources, including both people and automated algorithms, over a period of weeks, months, or years. Noy and her colleagues came to the same conclusion while collecting community-based mappings for the BioPortal ontology collection [74]. That work also reified coreferences, but it stored significantly more provenance information about the individual relations, including discussion and user comments, application

context (conditions under which the relationship holds), mapping dependency (to express that this mapping holds if and only if some other mapping holds), mapping algorithm, creation date, creator (the person who uploaded the mapping), and external references (e.g. relevant publications). Unfortunately, this information is currently encoded largely as free-text, which violates the underlying Semantic Web principle that information about data and how it relates should be accessible to both humans and machines. Establishing an appropriate method for representing provenance and contextual information for alignments and individual mappings remains an important challenge for the field of semantic data integration.

Acknowledgements This work was supported in part by the National Science Foundation award 1440202 GeoLink - Leveraging Semantics and Linked Data for Data Sharing and Discovery in the Geosciences. It was also partially supported by Fundaç ão para a Ciência e Tecnologia (PTDC/EEI-ESS/4633/2014).

References

1. M. R. Anderson, D. Antenucci, V. Bittorf, M. Burgess, M. J. Cafarella, A. Kumar, F. Niu, Y. Park, C. Ré, and C. Zhang. Brainwash: A data system for feature engineering. In *CIDR*, 2013.
2. M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.
3. K. Baclawski, C. J. Matheus, M. M. Kokar, J. Letkowski, and P. A. Kogut. Towards a symptom ontology for semantic web applications. In *The Semantic Web–ISWC 2004*, pages 650–667. Springer, 2004.
4. A. Ballatore and A. Zipf. A conceptual quality framework for volunteered geographic information. In *Spatial Information Theory*, pages 89–107. Springer, 2015.
5. T. Berners-Lee, J. Hendler, O. Lassila, et al. The semantic web. *Scientific american*, 284(5):28–37, 2001.
6. O. Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl 1):D267–D270, 2004.
7. A. T. Boin and G. J. Hunter. Do spatial data consumers really understand data quality information. In *7th International symposium on spatial accuracy assessment in natural resources and environmental sciences*, pages 215–224. Citeseer, 2006.
8. A. Callahan, J. Cruz-Toledo, P. Ansell, and M. Dumontier. Bio2rdf release 2: Improved coverage, interoperability and provenance of life science linked data. In *The semantic web: semantics and big data*, pages 200–212. Springer, 2013.
9. S. Castano, A. Ferrara, S. Montanelli, and G. Varese. Ontology and instance matching. In *Knowledge-driven multimedia information extraction and ontology evolution*, pages 167–195. Springer, 2011.
10. M. Cheatham, Z. Dragisic, J. Euzenat, D. Faria, A. Ferrara, G. Flouris, I. Fundulaki, R. Granada, V. Ivanova, E. Jiménez-Ruiz, et al. Results of the ontology alignment evaluation initiative 2015. In *10th ISWC workshop on ontology matching (OM)*, pages 60–115. No commercial editor., 2015.
11. M. Cheatham and P. Hitzler. String similarity metrics for ontology alignment. In *The Semantic Web–ISWC 2013*, pages 294–309. Springer, 2013.
12. M. Cheatham and P. Hitzler. Conference v2. 0: An uncertain version of the oaei conference benchmark. In *The Semantic Web–ISWC 2014*, pages 33–48. Springer, 2014.

13. M. Cheatham and P. Hitzler. The properties of property alignment. In *Proceedings of the 9th International Conference on Ontology Matching-Volume 1317*, pages 13–24. CEUR-WS. org, 2014.
14. R. Cornet and N. de Keizer. Forty years of snomed: a literature review. *BMC medical informatics and decision making*, 8(Suppl 1):S2, 2008.
15. I. F. Cruz, F. P. Antonelli, and C. Stroe. Agreementmaker: efficient matching for large real-world schemas and ontologies. *Proceedings of the VLDB Endowment*, 2(2):1586–1589, 2009.
16. I. F. Cruz, C. Stroe, and M. Palmonari. Interactive user feedback in ontology matching using signature vectors. In *Data Engineering (ICDE), 2012 IEEE 28th International Conference on*, pages 1321–1324. IEEE, 2012.
17. I. F. Cruz, C. Stroe, C. Pesquita, F. M. Couto, and V. Cross. Biomedical ontology matching using the agreementmaker system. In *ICBO*, 2011.
18. B. Di Martino. Semantic web services discovery based on structural ontology matching. *International Journal of Web and Grid Services*, 5(1):46–65, 2009.
19. Z. Dragisic, K. Eckert, J. Euzenat, D. Faria, A. Ferrara, R. Granada, V. Ivanova, E. Jiménez-Ruiz, A. O. Kempf, P. Lambrix, et al. Results of the ontology alignment evaluation initiative 2014. In *Proceedings of the 9th International Conference on Ontology Matching-Volume 1317*, pages 61–104. CEUR-WS. org, 2014.
20. S. Duan, A. Fokoue, O. Hassanzadeh, A. Kementsietsidis, K. Srinivas, and M. J. Ward. Instance-based matching of large ontologies using locality-sensitive hashing. In *The Semantic Web–ISWC 2012*, pages 49–64. Springer, 2012.
21. A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios. Duplicate record detection: A survey. *Knowledge and Data Engineering, IEEE Transactions on*, 19(1):1–16, 2007.
22. J. Euzenat. Brief overview of t-tree: The tropes taxonomy building tool. *Advances in Classification Research Online*, 4(1):69–88, 1993.
23. J. Euzenat, C. Meilicke, H. Stuckenschmidt, P. Shvaiko, and C. Trojahn. Ontology alignment evaluation initiative: Six years of experience. In *Journal on Data Semantics XV*, pages 158–192. Springer, 2011.
24. J. Euzenat and P. Shvaiko. *Ontology matching*, volume 18. Springer Heidelberg, 2007.
25. S. M. Falconer and M.-A. Storey. *A cognitive support framework for ontology mapping*. Springer, 2007.
26. Z. Fan and S. Zlatanova. Exploring ontologies for semantic interoperability of data in emergency response. *Applied Geomatics*, 3(2):109–122, 2011.
27. D. Faria, E. Jiménez-Ruiz, C. Pesquita, E. Santos, and F. M. Couto. Towards annotating potential incoherences in bioportal mappings. In *The Semantic Web–ISWC 2014*, pages 17–32. Springer, 2014.
28. D. Faria, C. Pesquita, E. Santos, I. F. Cruz, and F. M. Couto. Automatic background knowledge selection for matching biomedical ontologies. *PloS one*, 9(11):e111226, 2014.
29. D. Faria, C. Pesquita, E. Santos, M. Palmonari, I. F. Cruz, and F. M. Couto. The agreementmakerlight ontology matching system. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pages 527–541. Springer, 2013.
30. I. P. Fellegi and A. B. Sunter. A theory for record linkage. *Journal of the American Statistical Association*, 64(328):1183–1210, 1969.
31. B. Gallagher. Matching structure and semantics: A survey on graph-based pattern matching. *AAAI FS*, 6:45–53, 2006.
32. A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. Sweetening ontologies with dolce. In *Knowledge engineering and knowledge management: Ontologies and the semantic Web*, pages 166–181. Springer, 2002.
33. M. Granitzer, V. Sabol, K. W. Onn, D. Lukose, and K. Tochtermann. Ontology alignmenta survey with focus on visually supported semi-automatic techniques. *Future Internet*, 2(3):238–258, 2010.
34. B. C. Grau, Z. Dragisic, K. Eckert, J. Euzenat, A. Ferrara, R. Granada, V. Ivanova, E. Jiménez-Ruiz, A. O. Kempf, P. Lambrix, et al. Results of the ontology alignment evaluation initiative 2013. In *Proceedings of the 8th International Conference on Ontology Matching-Volume 1111*, pages 61–100. CEUR-WS. org, 2013.

35. T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.
36. K. Gunaratna, K. Thirunarayan, P. Jain, A. Sheth, and S. Wijeratne. A statistical and schema independent approach to identify equivalent properties on linked data. In *Proceedings of the 9th International Conference on Semantic Systems*, pages 33–40. ACM, 2013.
37. H. Halpin and P. J. Hayes. When owl: sameas isn't the same: An analysis of identity links on the semantic web. In *LDOW*, 2010.
38. F. Hamdi, B. Safar, N. B. Niraula, and C. Reynaud. Taxomap alignment and refinement modules: Results for oaei 2010. In *Proceedings of the 5th International Workshop on Ontology Matching (OM-2010) Collocated with the 9th International Semantic Web Conference (ISWC-2010), CEUR-WS*, pages 212–220, 2010.
39. M. Hartung, A. Gross, T. Kirsten, and E. Rahm. Effective mapping composition for biomedical ontologies. In *Proc of Semantic Interoperability in Medical Informatics (SIMI-12), Workshop at ESWC*, volume 12, 2012.
40. M. Hartung, L. Kolb, A. Groß, and E. Rahm. Optimizing similarity computations for ontology matching-experiences from gomma. In *Data Integration in the Life Sciences*, pages 81–89. Springer, 2013.
41. P. Hitzler, M. Krotzsch, and S. Rudolph. *Foundations of semantic web technologies*. CRC Press, 2011.
42. R. Hoehndorf, M. Dumontier, A. Oellrich, D. Rebholz-Schuhmann, P. N. Schofield, and G. V. Gkoutos. Interoperability between biomedical ontologies through relation expansion, upper-level ontologies and automatic reasoning. *PloS one*, 6(7):e22006, 2011.
43. Y. Hu, K. Janowicz, D. Carral, S. Scheider, W. Kuhn, G. Berg-Cross, P. Hitzler, M. Dean, and D. Kolas. A geo-ontology design pattern for semantic trajectories. In *Spatial Information Theory*, pages 438–456. Springer, 2013.
44. P. G. Ipeirotis. Demographics of mechanical turk. 2010.
45. V. Ivanova, P. Lambrix, and J. Åberg. Requirements for and evaluation of user support for large-scale ontology alignment. In *The Semantic Web. Latest Advances and New Domains*, pages 3–20. Springer, 2015.
46. P. Jain, P. Hitzler, A. P. Sheth, K. Verma, and P. Z. Yeh. Ontology alignment for linked open data. In *The Semantic Web–ISWC 2010*, pages 402–417. Springer, 2010.
47. K. Janowicz and M. Compton. The stimulus-sensor-observation ontology design pattern and its integration into the semantic sensor network ontology. In *Proceedings of the 3rd International Conference on Semantic Sensor Networks-Volume 668*, pages 64–78. CEUR-WS. org, 2010.
48. E. Jiménez-Ruiz and B. C. Grau. Logmap: Logic-based and scalable ontology matching. In *The Semantic Web–ISWC 2011*, pages 273–288. Springer, 2011.
49. E. Jiménez-Ruiz, B. C. Grau, I. Horrocks, and R. B. Llavori. Logic-based ontology integration using contentmap. In *JISBD*, pages 316–319. Citeseer, 2009.
50. E. Jiménez-Ruiz, B. C. Grau, Y. Zhou, and I. Horrocks. Large-scale interactive ontology matching: Algorithms and implementation. In *ECAI*, volume 242, pages 444–449, 2012.
51. E. Jiménez-Ruiz, C. Meilicke, B. C. Grau, and I. Horrocks. Evaluating mapping repair systems with large biomedical ontologies.
52. N. Kheder and G. Diallo. Servombi at oaei 2015.
53. W. O. Kow, V. Sabol, M. Granitzer, W. Kienrich, and D. Lukose. A visual soa-based ontology alignment tool. In *Proceedings of the Sixth International Workshop on Ontology Matching (OM 2011)*, volume 10, 2011.
54. K. Kyzirakos, M. Karpathiotakis, G. Garbis, C. Nikolaou, K. Bereta, I. Papoutsis, T. Herekakis, D. Michail, M. Koubarakis, and C. Kontoes. Wildfire monitoring using satellite images, ontologies and linked geospatial data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 24:18–26, 2014.
55. P. Lambrix and A. Edberg. Evaluation of ontology merging tools in bioinformatics. In *Pacific Symposium on Biocomputing*, volume 8, pages 589–600, 2003.
56. P. Lambrixa and R. Kaliyaperumalb. A session-based ontology alignment approach for aligning large ontologies.

57. M. Lanzenberger and J. Sampson. Alviz-a tool for visual ontology alignment. In *Information Visualization, 2006. IV 2006. Tenth International Conference on*, pages 430–440. IEEE, 2006.
58. L. Li, X. Xing, H. Xia, and X. Huang. Entropy-weighted instance matching between different sourcing points of interest. *Entropy*, 18(2):45, 2016.
59. V. Lush, L. Bastin, and J. Lumsden. Geospatial data quality indicators. 2012.
60. R. McCann, W. Shen, and A. Doan. Matching schemas in online communities: A web 2.0 approach. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 110–119. IEEE, 2008.
61. G. McKenzie, K. Janowicz, and B. Adams. A weighted multi-attribute method for matching user-generated points of interest. *Cartography and Geographic Information Science*, 41(2):125–137, 2014.
62. C. Meilicke. *Alignment incoherence in ontology matching*. PhD thesis, Universitätsbibliothek Mannheim, 2011.
63. S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *Data Engineering, 2002. Proceedings. 18th International Conference on*, pages 117–128. IEEE, 2002.
64. V. Momtchev, D. Peychev, T. Primov, and G. Georgiev. Expanding the pathway and interaction knowledge in linked life data. *Proc. of International Semantic Web Challenge*, 2009.
65. P. Mooney, P. Corcoran, and A. C. Winstanley. Towards quality metrics for openstreetmap. In *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, pages 514–517. ACM, 2010.
66. J. M. Mortensen. Crowdsourcing ontology verification. In *The Semantic Web–ISWC 2013*, pages 448–455. Springer, 2013.
67. J. M. Mortensen, M. A. Musen, and N. F. Noy. Crowdsourcing the verification of relationships in biomedical ontologies. In *AMIA Annual Symposium (submitted, 2013)*, 2013.
68. J. M. Mortensen, M. A. Musen, and N. F. Noy. Ontology quality assurance with the crowd. In *First AAAI Conference on Human Computation and Crowdsourcing*, 2013.
69. B. Motik, P. F. Patel-Schneider, B. Parsia, C. Bock, A. Fokoue, P. Haase, R. Hoekstra, I. Horrocks, A. Ruttenberg, U. Sattler, et al. Owl 2 web ontology language: Structural specification and functional-style syntax. *W3C recommendation*, 27(65):159, 2009.
70. C. J. Mungall. Obol: integrating language and meaning in bio-ontologies. *Comparative and functional genomics*, 5(6-7):509–520, 2004.
71. C. J. Mungall, G. V. Gkoutos, C. L. Smith, M. A. Haendel, S. E. Lewis, and M. Ashburner. Integrating phenotype ontologies across multiple species. *Genome biology*, 11(1):R2, 2010.
72. D. Ngo and Z. Bellahsene. Yam++: A multi-strategy based approach for ontology matching task. In *Knowledge Engineering and Knowledge Management*, pages 421–425. Springer, 2012.
73. A. Nikolov, M. dAquin, and E. Motta. Unsupervised learning of link discovery configuration. In *The Semantic Web: Research and Applications*, pages 119–133. Springer, 2012.
74. N. F. Noy, N. Griffith, and M. A. Musen. *Collecting community-based mappings in an ontology repository*. Springer, 2008.
75. N. F. Noy and C. D. Hafner. The state of the art in ontology design: A survey and comparative review. *AI Magazine*, 18(3):53, 1997.
76. N. F. Noy, J. Mortensen, M. A. Musen, and P. R. Alexander. Mechanical turk as an ontology engineer?: using microtasks as a component of an ontology-engineering workflow. In *Proceedings of the 5th Annual ACM Web Science Conference*, pages 262–271. ACM, 2013.
77. N. F. Noy, N. H. Shah, P. L. Whetzel, B. Dai, M. Dorf, N. Griffith, C. Jonquet, D. L. Rubin, M.-A. Storey, C. G. Chute, et al. Bioportal: ontologies and integrated data resources at the click of a mouse. *Nucleic acids research*, page gkp440, 2009.
78. D. J. Odgers and M. Dumontier. Mining electronic health records using linked data. *AMIA Summits on Translational Science Proceedings*, 2015:217, 2015.
79. D. Oliveira and C. Pesquita. Compound matching of biomedical ontologies. In *Proceedings of the International Conference on Biomedical Ontology 2015*, pages 87–88, 2015.

80. J. Ortmann, M. Limbu, D. Wang, and T. Kauppinen. Crowdsourcing linked open data for disaster management. In *Proceedings of the Terra Cognita Workshop on Foundations, Technologies and Applications of the Geospatial Web in conjunction with the ISWC*, pages 11–22. Citeseer, 2011.
81. H. Paulheim, S. Hertling, and D. Ritze. Towards evaluating interactive ontology matching tools. In *The Semantic Web: Semantics and Big Data*, pages 31–45. Springer, 2013.
82. A. Pease, I. Niles, and J. Li. The suggested upper merged ontology: A large ontology for the semantic web and its applications. In *Working notes of the AAAI-2002 workshop on ontologies and the semantic web*, volume 28, 2002.
83. C. Pesquita, D. Faria, E. Santos, and F. M. Couto. To repair or not to repair: reconciling correctness and coherence in ontology reference alignments. In *OM*, pages 13–24, 2013.
84. C. Pesquita, D. Faria, E. Santos, J.-M. Neefs, and F. M. Couto. Towards visualizing the alignment of large biomedical ontologies. In *Data Integration in the Life Sciences*, pages 104–111. Springer, 2014.
85. C. Pesquita, D. Faria, C. Stroe, E. Santos, I. F. Cruz, and F. M. Couto. Whats in a nym? synonyms in biomedical ontology matching. In *The Semantic Web–ISWC 2013*, pages 526–541. Springer, 2013.
86. R. G. Raskin and M. J. Pan. Knowledge representation in the semantic web for earth and environmental terminology (sweet). *Computers & geosciences*, 31(9):1119–1125, 2005.
87. S. Rong, X. Niu, E. W. Xiang, H. Wang, Q. Yang, and Y. Yu. A machine learning approach for instance matching based on similarity metrics. In *The Semantic Web–ISWC 2012*, pages 460–475. Springer, 2012.
88. A. Ruttenberg, J. A. Rees, M. Samwald, and M. S. Marshall. Life sciences on the semantic web: the neurocommons and beyond. *Briefings in bioinformatics*, page bbp004, 2009.
89. M. Salvadores, P. R. Alexander, M. A. Musen, and N. F. Noy. Bioportal as a dataset of linked biomedical ontologies and terminologies in rdf. *Semantic web*, 4(3):277–284, 2013.
90. E. Santos, D. Faria, C. Pesquita, and F. Couto. Ontology alignment repair through modularization and confidence-based heuristics. *arXiv preprint arXiv:1307.5322*, 2013.
91. E. Santos, D. Faria, C. Pesquita, and F. M. Couto. Ontology alignment repair through modularization and confidence-based heuristics. *PloS one*, 10(12), 2015.
92. C. Sarasua, E. Simperl, and N. F. Noy. Crowdmap: Crowdsourcing ontology alignment with microtasks. In *The Semantic Web–ISWC 2012*, pages 525–541. Springer, 2012.
93. M. Schmachtenberg, C. Bizer, A. Jentzsch, and R. Cyganiak. Linking open data cloud diagram, 2014, 2014.
94. L. M. Schriml, C. Arze, S. Nadendla, Y.-W. W. Chang, M. Mazaitis, V. Felix, G. Feng, and W. A. Kibbe. Disease ontology: a backbone for disease semantic integration. *Nucleic acids research*, 40(D1):D940–D946, 2012.
95. V. Sehgal, L. Getoor, and P. D. Viechnicki. Entity resolution in geospatial data integration. In *Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*, pages 83–90. ACM, 2006.
96. B. Severo, C. Trojahn, and R. Vieira. Voar: A visual and integrated ontology alignment environment. 2014.
97. A. Shepherd, C. Chandler, R. Arko, Y. Chen, A. Krisnadhi, P. Hitzler, T. Narock, R. Groman, and S. Rauch. Semantic entity pairing for improved data validation and discovery. In *EGU General Assembly Conference Abstracts*, volume 16, page 2476, 2014.
98. H. Shi, K. Maly, S. Zeil, and M. Zubair. Comparison of ontology reasoning systems using custom rules. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, page 16. ACM, 2011.
99. K. Siorpaes and M. Hepp. *Ontogame: Weaving the semantic web by online games*. Springer, 2008.
100. S. Sizov. Geofolk: Latent spatial semantics in web 2.0 social media. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, pages 281–290. ACM, 2010.

101. B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L. J. Goldberg, K. Eilbeck, A. Ireland, C. J. Mungall, et al. The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251–1255, 2007.
102. V. Spiliopoulos, G. A. Vouros, and V. Karkaletsis. On the discovery of subsumption relations for the alignment of ontologies. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(1):69–88, 2010.
103. H. Stuckenschmidt, J. Noessner, and F. Fallahi. A study in user-centric data integration. In *ICEIS (3)*, pages 5–14, 2012.
104. F. M. Suchanek, S. Abiteboul, and P. Senellart. Paris: Probabilistic alignment of relations, instances, and schema. *Proceedings of the VLDB Endowment*, 5(3):157–168, 2011.
105. Y. Sure, S. Bloehdorn, P. Haase, J. Hartmann, and D. Oberle. The swrc ontology–semantic web for research communities. In *Progress in Artificial Intelligence*, pages 218–231. Springer, 2005.
106. V. Svátek, O. Šváb-Zamazal, and V. Presutti. Ontology naming pattern sauce for (human and computer) gourmets. In *Workshop on Ontology Patterns*, pages 171–178, 2009.
107. J. M. Taylor, D. Poliakov, and L. J. Mazlack. Domain-specific ontology merging for the semantic web. In *Fuzzy Information Processing Society, 2005. NAFIPS 2005. Annual Meeting of the North American*, pages 418–423. IEEE, 2005.
108. S. Thakkar, C. A. Knoblock, and J. L. Ambite. Quality-driven geospatial data integration. In *Proceedings of the 15th annual ACM international symposium on Advances in geographic information systems*, page 16. ACM, 2007.
109. S. Thaler, E. P. B. Simperl, and K. Siorpaes. Spotthelink: A game for ontology alignment. *Wissensmanagement*, 182:246–253, 2011.
110. S. Tschirner, A. Scherp, and S. Staab. Semantic access to inspire. In *Terra Cognita 2011 Workshop Foundations, Technologies and Applications of the Geospatial Web*, page 75. Citeseer, 2011.
111. E. Voyloshnikova, B. Fu, L. Grammel, and M.-A. D. Storey. Biomixer: Visualizing mappings of biomedical ontologies. In *ICBO*, 2012.
112. A. J. Williams, L. Harland, P. Groth, S. Pettifer, C. Chichester, E. L. Willighagen, C. T. Evelo, N. Blomberg, G. Ecker, C. Goble, et al. Open phacts: semantic interoperability for drug discovery. *Drug discovery today*, 17(21):1188–1198, 2012.
113. A. Wu and X. Lopez. Building enterprise applications with oracle database 11g semantic technologies. Presentation at Semantic Technologies Conference, San Jose, 2009.
114. T. Zhao, C. Zhang, M. Wei, and Z.-R. Peng. Ontology-based geospatial data query and integration. In *Geographic Information Science*, pages 370–392. Springer, 2008.