# Mining Rules to Align Knowledge Bases

Luis Galárraga[1], Nicoleta Preda[2], Fabian M. Suchanek[1]
[1]Max-Planck Institute for Informatics, Saarbrücken, Germany
[2]University of Versailles, PRiSM-CNRS, Versailles, France
[1]{lgalarra, suchanek}@mpi-inf.mpg.de, [2]{preda}@prism.uvsq.fr

## ABSTRACT

The Semantic Web has made huge progress in the last decade, and now comprises hundreds of knowledge bases (KBs). The Linked Open Data cloud connects the KBs in this Web of data. However, the links between the KBs are mostly concerned with the instances, not with the schema. Aligning the schemas is not easy, because the KBs may differ not just in their names for relations and classes, but also in their inherent structure. Therefore, we argue in this paper that advanced schema alignment is needed to tie the Semantic Web together. We put forward a particularly simple approach to illustrate how that might look.

## Categories and Subject Descriptors

H.m [**Information Systems**]: Miscellanous

## General Terms

Design, Standarization

## Keywords

Rule Mining, Linked Data, Ontology Alignment

## 1. INTRODUCTION

Recent years have seen a rise of large knowledge bases (KBs). Endeavors such as YAGO [20], DBpedia [1], NELL [5], and KnowItAll [9] have constructed huge KBs of millions of facts. These projects are complemented by commercial projects such as Freebase [22], Google's knowledge graph, Evi.com (bought by Amazon), and Facebook's Graph Search. Thanks to the Semantic Web, RDF [24] has established itself as a widely accepted standard of knowledge representation. Many of the KBs overlap in their instances. All of the above KBs, e.g., contain cities and places. All KBs that feed from Wikipedia, such as YAGO, DBpedia, Freebase, and Google's knowledge graph, share a large part of their entities. Consequently, many of the (public) KBs have been connected in the Linked Open Data Cloud [14] (LOD). The LOD cloud provides *sameAs* links between equivalent entity identifiers across KBs. Altogether, the LOD cloud provides hundreds of millions of such links, thus interconnecting billions of statements.

These links, however, concern mostly instances. The schema of the KBs, i.e., the class hierarchy and the relations of the KBs, have not yet been mapped at large scale. This entails that, although the instances are aligned, the data is not interoperable. A query formulated in the schema of one KB will not have answers in another KB – even if the desired information is there, and even if the instances have been linked between the two resources. Hence, despite the great advances of linked open data, the KBs are still to a large degree disconnected databases.

A standard data integration solution based on manually defined mappings would not scale-up to the hundreds of KBs on the Semantic Web. Recent work [19] has allowed finding relation and class alignments across KBs at large scale. However, this approach requires the KBs to have the same structure: One relation in one KB has to correspond to one relation in the other. Real-data examples show us that this is not always the case. For example, if the user asks for the country of birth of Elvis, then one KB may express this by the relationship *wasBornInCountry*. Another KB, in contrast, may require a join of the relationship *wasBornInCity* with the relationship *locatedInCountry*. This problem is particularly prevalent in cases where one KB distinguishes between the entity and its label, while the other one does not. For example, one KB could say *sang(Elvis, "All Shook Up")*, while the other one could say *sang(Elvis, allShookUp)*, *label(allShookUp, "All Shook Up")*. Such a structural mismatch would derail any alignment system that assumes isomorphic KBs. Consequently, any query evaluation algorithm that uses them would miss relevant query plans.

Another important issue faced by a data integration systems is the translation of literal values from one KB to another. For example, different KBs may use different labels for the same entities. MusicBrainz uses abbreviated names for countries, stating e.g. *livedIn(Elvis, "DE")*. YAGO, on the other hand, uses complete names for countries. If the data of MusicBrainz were integrated directly into YAGO, the result would be inconsistencies.

This leaves us to conclude that, even though KBs talk about the same entities, they often talk about them in different languages. Even though they say the same things, or even complement each other, and even though they use the same knowledge representation format, they are not able to


http://dx.doi.org/10.1145/2509558.2509566 ...$15.00.

"understand" each other. The reason is a structural mismatch between the schemas of the KBs. Our vision is that this structural mismatch be overcome. We would like to see a Semantic Web where the KBs are not just linked by their instances, but also by their relations and classes – irrespective of structural differences, and across different schemas. This would allow users to make full use of the data and links of the Semantic Web.

To illustrate how such an alignment could look, we resort to a simple, yet very effective technique. We assume that some of the instances of two KBs have already been aligned. Then we coalesce the two KBs to a single KB, and apply rule mining [10]. This yields what we call ROSA rules: Logical rules that reveal complex relationships between properties and concepts of the KBs. This idea subsumes state-of-the-art ontology matching [19], in that it can find equivalences of relationships, equivalence to the inverse of a relationship, and the subsumption of relationships. However, it goes further by finding that one "hop" in one KB can correspond to a several hops in the other. It can find that one literal in one KB corresponds to a different constant in the other KB, or that one class in one KB corresponds to a relationship in the other. We hope that, by this illustration, we can open up the door for a new, and exciting, area of research.

# 2. RELATED WORK

**Scope.** Several aspects of KB alignment have been addressed in recent work: the alignment of classes [12, 11], the alignment of classes together with relations (T-Box) [2, 6, 18], and the alignment of instances (A-Box) [14, 13, 3]. Holistic approaches have been investigated in [19, 23].

In the present paper, we agree that the alignment of instances has been solved to a large degree. However, we are concerned that state-of-the-art solutions for the schema alignment of KBs are not yet mature enough.

**Instance-Based Approaches.** Some approaches [23, 19] can align the instances and the schema of two KBs at the same time. ILIADS [23] combines a clustering algorithm based on lexical, graph-structure and data instance statistics with a logical inference to check the consistency of the mappings. Most large KBs, however, do not come with the OWL axioms on which ILIADS relies. PARIS [19] develops a probabilistic model for graph alignment. However, both ILIADS and PARIS assume that one relation or class in one KB corresponds to one relation or class in the other KB. In practice, this is often not the case.

**Schema Alignment.** Schema matching solutions such as CLIO [16] typically rely on database constraints, which are often unavailable for KBs. Some schema matching systems are able to align KBs [2, 6, 18]. [2] expects as input OWL descriptions and ignores data instances. [6] develops a user-assisted approach. [18] proposes a tool collection for information integration. In our vision, the alignment of KBs would happen fully automatically and without the need for database constraints.

**Data Translation.** The approach of [4] argued for the introduction of more complex mapping rules, which are able to express data translations. However, such formalisms have not yet been widely adopted due to scalability issues. In this work, we propose to focus on mining a small, yet expressive set of mappings patterns, which capture many real-data cases. The actual translation of data from one KB to another KB shares resemblance to the query discovery problem [16]. Query discovery and schema matching are seen as complementary and independent problems [15]. Query discovery solutions are not directly applicable to KBs as they are data model dependent or rely on database constraints. Furthermore, the data transformations that we envision go beyond data restructuring.

**Association Rule Mining for Ontology Alignment.** To our knowledge, there are only few works that mine rules for KB alignment [7, 21]. These works focus exclusively on the alignment of hierarchies of entities. Our vision goes beyond this goal. We would like to express more complex mappings and actual data transformations.

# 3. PRELIMINARIES

**RDF KBs.** This paper assumes that the KBs are represented in RDF [1]. An RDF KB is a set of triples of the form $<x,r,y>$, called *facts*, where $x$ is the subject, $r$ the relation and $y$ the object. This paper uses the logical notation *r(x,y)*. We assume that, for every relation $r$, there is also the inverse relation $r^{-1}$, i.e., if the KB contains $r(x,y)$, then it also implicitly contains $r^{-1}(y,x)$. Facts state relationships between entities of a KB (e.g. *married(Elvis, Priscilla)*). Facts may also define the schema of the KB, i.e., its class hierarchy, and its relations with domains and ranges.

**Horn rules.** The main idea of this paper is to reconciliate schemas by finding correlations across ontologies. For this purpose, we will use Horn rules. These are based on atoms. An atom is a fact that contains at least one variable as argument. A Horn rule consists of a head and a body where the head is a single atom and the body is a conjunction of atoms. We denote Horn rules by an implication

$$B_1, B_2 \dots B_n \Rightarrow r(x,y), \text{written as} \quad \vec{B} \Rightarrow r(x,y)$$

where each $B_i$ is an atom. As example, consider the rule:

$$hasChild(x,y), isCitizenOf(x,z) \Rightarrow isCitizenOf(y,z)$$

We say that two atoms are *connected* if they share at least one variable. A rule is connected if every atom is transitively connected to every other atom of the rule. Non-connected rules are normally not interesting. A Horn rule is *closed* if every variable in the rule appears at least twice (as in our example). Closed Horn rules are particularly interesting because if the variables of the body are substituted by constants so that they match facts in a KB, we can infer concrete facts. We consider only closed rules.

**Rule Mining.** Learning logical rules from datasets is the central topic of research in the field of Inductive Logic Programming (ILP). Conceptually, ILP learns (Horn) rules from a set of positive and negative examples, where the goal is to find hypotheses that cover all the positive examples and none of the negative examples. Since KBs usually do not contain negative data, different ILP approaches resort to different strategies to generate negative examples.

A vanilla association rule mining approach [8] could simply regard all absent data as counter-examples. By using absence of evidence as evidence of absence, however, this measure violates the Open World Assumption that most KBs make. To address this issue, Muggleton has developed a positive-only learning evaluation score for ILP without counter-examples [17]. This approach generates counter-examples randomly. The AMIE approach [10] uses yet an-

[1] http://www.w3.org/TR/rdf-primer/

other strategy to generate counter examples: the Partial Completeness Assumption (PCA). The PCA assumes that a KB knows either *all* or *none* of the $r$-attributes of some $x$. This is true for functional relations. It is also true for well documented KBs or for KBs that are extracted from well documented sources. Under this assumption, we count as counter-examples for a rule $\vec{B} \Rightarrow r(x,y)$ only those instances $x$ that have $r$ relations, but not $r(x,y)$. This yields the following confidence measure:

$$pcaconf(\vec{B} \Rightarrow r(x,y)) := \frac{\#(x,y) : \exists z_1,...,z_m : \vec{B} \wedge r(x,y)}{\#(x,y) : \exists z_1,...,z_m, y' : \vec{B} \wedge r(x,y')}$$

Here, $z_1,...,z_m$ are the free variables of the body, and $\#(x,y) : \mathcal{A}$ is the number of pairs $(x,y)$ that fulfill $\mathcal{A}$. Thereby, the PCA confidence takes into account that KBs can be incomplete. [10] shows that the PCA confidence consistently identifies the rules that make the largest number of correct predictions. Therefore, we use the PCA confidence also in our setting.

**Coalescing KBs.** We are interested in discovering complex schema relationships between two given KBs $\mathcal{K}_1$ and $\mathcal{K}_2$ in RDF format. We assume that the instances of the KBs have already been aligned, at least in part. This can be done by a (partial) substitution $\sigma$, which maps the instances of $\mathcal{K}_1$ to the *sameAs* counterparts from $\mathcal{K}_2$ if any, or to themselves otherwise. It leaves literals unchanged. As pointed out in [19], different KBs may use the same relation (as given by a URI) in different ways. Therefore, we use a substitution $t$ that substitutes all relation names in $\mathcal{K}_1$ so as to make sure they are different from the relation names in $\mathcal{K}_2$. With this in mind, we coalesce the two KBs as follows:

$$\mathcal{K} = \{r'(\sigma(x), \sigma(y)) \mid r(x,y) \in \mathcal{K}_1 \wedge r' = t(r)\} \cup \mathcal{K}_2$$

This definition entails that we have two sets of relations: $rel(\mathcal{K}_1) = t(\pi_{relation}(\mathcal{K}_1))$ and $rel(\mathcal{K}_2) = \pi_{relation}(\mathcal{K}_2)$. Our coalesced KB subsumes both KBs. We could restrict ourselves to the part that is common to both KBs, but then many alignments are lost because of missing data. We leave the detailed study of different coalescing techniques for future work.

**ROSA rules.** On the coalesced KB $\mathcal{K}$, we will mine rules. We are particularly interested in rules that express KB alignments. We call them ROSA rules:[2]

Definition 1. *A ROSA rule from a KB $\mathcal{K}_1$ to a KB $\mathcal{K}_2$ is a rule mined on the coalesced KB $\mathcal{K}$, such that the relations of the body belong to $rel(\mathcal{K}_1)$, and the relation of the head belongs to $rel(\mathcal{K}_2)$.*

This definition is asymmetric in the sense that we can mine ROSA rules from $\mathcal{K}_1$ to $\mathcal{K}_2$ and from $\mathcal{K}_2$ to $\mathcal{K}_1$. ROSA rules express one type of cross-schema alignments.

**KBs.** For our experiments, we mined ROSA rules on the following KBs:

**YAGO 2s**: We used the facts about instances contained in the datasets *yagoFacts* and *yagoLiteralFacts*, with 2.9M entities and 22.8M facts in total. We also used, though separately, the instance data contained in *yagoSimpleTypes*, which comprises 5.4M *rdf:type* statements.

[2]ROSA = Rule for Ontology Schema Alignment. For convenience, and with a slight redundancy, we speak of "ROSA rules".

| | |
|---|---|
| $r(x,y) \Rightarrow r'(x,y)$ | (R-subsumption) |
| $r(x,y) \Leftrightarrow r'(x,y)$ | (R-equivalence) |
| $type(x,C) \Rightarrow type'(x,C')$ | (C-subsumption) |
| $r_1(x,y), r_2(y,z) \Rightarrow r'(x,z)$ | (2-Hops alignment) |
| $r(z,x), r(z,y) \Rightarrow r'(x,y)$ | (Triangle alignment) |
| $r_1(x,y), r_2(x,V) \Rightarrow r'(x,y)$ | (Specific R-subsumption) |
| $r(y,V) \Rightarrow r'(x,V')$ | (Attr-Value translation) |
| $r_1(x,V_1), r_2(x,V_2) \Rightarrow r'(x,V')$ | (2-Value translation) |

**Figure 1: ROSA Rules ($r \in rel(\mathcal{K}_1), r' \in rel(\mathcal{K}_2)$).**

**DBpedia 3.8**: We used the *person data* and *raw infobox properties* datasets, which together contain 11M facts about 2.1M entities. We also used the *ontology infoboxes* dataset, which has 13.2M *rdf:type* statements about 2.3M entities.

**Freebase**: We used information about people, which comprises 19M facts about 2.7M subjects. In addition, we used the instance facts, which comprise 140M *rdf:type* statements about 74M entities.

**IMDb**: We used a custom crawl of IMDb, similar to the one in [19]. It comprises 722K entities and 12M facts.

We use the namespace prefixes $Y$, $D$, $F$, and $I$ for these KBs, respectively. ROSA rules require pre-existing instance alignments. For the first three KBs, we used the instance alignments from the Linked Data cloud. For IMDb, we used the gold standard instance alignments provided by [19].

For the actual mining of rules, we use the AMIE system [10]. AMIE can mine rules on large KBs in a few minutes.

# 4. ALIGNMENT PATTERNS

**Goal.** Our goal is to illustrate the complex schema alignments that are necessary to make two KBs interoperable. Some of the interesting alignments are given by ROSA rules. Given two KBs, we mine rules from the first KB to the second, and from the second to the first. In the following, assume that we mine from a KB $\mathcal{K}_1$ to a KB $\mathcal{K}_2$ on their coalesced KB $\mathcal{K}$.

**Rule Patterns.** Figure 1 groups some useful ROSA rules of up to 3 atoms into patterns. Arguments in lower case denote variables, whereas uppercase letters refer to constant values. Since we assume that every relation $r$ is also present in its inverse, $r^{-1}$, the patterns also comprise rules that involve a swap of arguments. We will now discuss each of these patterns in detail. For each pattern, we show some mined rules together with their PCA confidence.

**R-subsumption.** We say $r'$ *subsumes* $r$, if $(\forall)$ $r(x,y) \in \mathcal{K}$,

$$r(x,y) \Rightarrow r'(x,y)$$

This rule holds when two ontologies contain semantically close relations with different levels of specificity. In this respect, the rule can capture *subPropertyOf* relationships, as defined in the RDF Schema. For instance, in YAGO the relationship between an author and his œuvre is labelled generically with *Y:created*, while DBpedia defines more specific relations such as *D:writer* for authors or *D:musicalArtist* for singers. To show this, we ran AMIE on a coalasced KB built from YAGO and DBpedia. AMIE found 360 R-subsumption alignments. We show the top 3 [3] with their PCA confidences:

$$D{:}musicalArtist(x,y) \Rightarrow Y{:}created(y,x) \quad (90\%)$$
$$D{:}musicalBand(x,y) \Rightarrow Y{:}created(y,x) \quad (90\%)$$
$$D{:}mainInterest(x,y) \Rightarrow Y{:}isInterestedIn(x,y) \quad (88\%)$$

**R-equivalence.** If two relations $r$ and $r'$ subsume each other, then they are *semantically equivalent*. In our framework, this translates to two ROSA implications. We define the confidence of an equivalence rule as the minimum of the PCA confidences of the two implications. On DBpedia and YAGO, we mined a total of 88 R-equivalence rules. The top 3 by PCA confidence are:

$$Y{:}directed \Leftrightarrow D{:}director \quad (98\%)$$
$$Y{:}wroteMusicFor \Leftrightarrow D{:}musicBy \quad (97\%)$$
$$Y{:}isCitizenOf \Leftrightarrow D{:}nationality \quad (96\%)$$

**C-subsumption.** A C-subsumption is a rule of the form

$$type(x,C) \Rightarrow type'(x,C')$$

where *type* is the *rdf:type* relationship in one KB, and *type'* is the *rdf:type* relationship in the other KB. If all instances of class $C$ are also instances of class $C'$, then $C'$ subsumes $C$, i.e. *rdfs:subClassOf*$(C, C')$. Class alignment is a crucial task in ontology integration, because instance information frequently constitutes a significant part of the contents of any ontology. As we did for R-subsumption rules, we used AMIE to align the instance data of YAGO and DBpedia. We show the top 3 alignments (from a total of 59) where the inverse implication holds with low confidence.

$$Y{:}type(x, Y{:}Site) \Rightarrow D{:}type(x, D{:}PopulatedPlace) \quad (97\%)$$
$$Y{:}type(x, Y{:}Site) \Rightarrow D{:}type(x, D{:}Settlement) \quad (95\%)$$
$$D{:}type(x, D{:}Athlete) \Rightarrow Y{:}type(x, Y{:}Person) \quad (91\%)$$

C-equivalence patterns follow immediately by combining C-subsumption rules with their corresponding inverse implications.

**More complex patterns.** Rules with constants or with more than 2 atoms increase the size of the search space drastically. As a consequence, rule mining finds many more rules. Some of the rules are *soft rules*. These do not express an a priori alignment, but rather a correlation that happens to be in the data. The rules can still be of use for reasoning [4], but are of more contingent nature. The following patterns all produce some soft rules with high PCA confidence. Therefore, we show as examples some handpicked interesting rules from the results. Our goal is to show that some quite complex alignments do exist between the KBs, even if we cannot identify them automatically yet.

**2-Hops subsumption.** A KB can express a relationship between two entities by passing through an intermediate entity (a blank node for example), while another KB expresses the same relationship by a single fact. This structural difference between two KBs can be captured by 2-hops subsumption rules of the form

$$r_1(x,y), r_2(y,z) \Rightarrow r'(x,z)$$

[3]We look only at R-subsumptions $A \Rightarrow B$ whose inverse $B \Rightarrow A$ has a low PCA confidence, because otherwise the subsumption is an R-equivalence.

For instance, in IMBb, the attribute *country-of-birth* points to the name of the country, while in YAGO it points to the country entity, which has a *label* attribute. By running AMIE on YAGO and IMDb, we mined the ROSA rule

$$Y{:}wasBornIn(x,y),\ Y{:}label(y,z) \Rightarrow I{:}bornIn(x,z) \quad (37\%)$$

Some of the top 2-hops subsumption alignments found by AMIE between YAGO and Freebase are soft rules:

$$Y{:}married(x,y),\ Y{:}child(y,z) \Rightarrow F{:}children(x,z) \quad (73\%)$$

**Triangle Alignments.** The case where the body atoms of a 2-hops subsumption have the same relation is particularly interesting, because it can capture sibling relationships, co-author links and other relationships that denote co-participation. We call such rules *triangle alignments*. As an example, consider the following rule that we mined on YAGO and Freebase:

$$Y{:}child(x,y),\ Y{:}child(x,z) \Rightarrow F{:}sibling(y,z) \quad (37\%)$$

**Specific R-subsumption.** An R-subsumption may become more specific by adding an attribute-value constraint to one of the arguments in the body of the rule. The following example was extracted from YAGO and Freebase:

$$Y{:}graduated(x,y),\ Y{:}type(y,\ Univ.) \Rightarrow F{:}institution(x,y) \quad (98\%)$$

The corresponding R-subsumption, $Y{:}graduated(x,y) \Rightarrow F{:}institution(x,y)$, has a PCA confidence of only 88%. In this example, the type constraint strengthened the precision of the mapping by also aligning the ranges of the relations.

**Attribute-Value Translation.** Different conventions for entity labels, as well as missing links between the literal values across KBs, require ROSA rules of the form

$$r(x,V) \Rightarrow r'(x,V')$$

We call these rules attribute-value translations, because they provide a bridge between the semantics of predicate-object combinations across KBs. By allowing constants in the arguments of the rules, AMIE mined translations between genders in YAGO and IMDb (99% confidence), as they are represented as URIs in YAGO and as literals in the IMDb crawl.

We also found translations between places in YAGO and their corresponding timezones in DBpedia, such as:

$$Y{:}locatedIn(x, Italy) \Rightarrow D{:}timeZone(x, CET) \quad (100\%)$$
$$Y{:}locatedIn(x, California) \Rightarrow D{:}timeZone(x, PST) \quad (100\%)$$

**2-Value Translation.** Differences in the verbosity of two ontologies can be captured by ROSA rules of the form

$$r_1(x,V_1), r_2(x,V_2) \Rightarrow r'(x,V')$$

This rule says that a fact about an entity in $\mathcal{K}_1$ is mapped to two different facts about the same entity in $\mathcal{K}_2$. An example mined from YAGO and Freebase (with 55% confidence) is:

$$F{:}type(x,Royal), F{:}gender(x,female) \Rightarrow Y{:}type(y,Princess)$$

We envision that adding functions to the translation rules could address even more schema incompatibilities. For instance, the following rule could translate between a KB that concatenates first name and last name, and a KB that does not:

$$firstName(x,y), lastName(x,z) \Rightarrow name(x, concatenate(y,z))$$

## 5. CONCLUSION

In this paper, we have argued that complex schema alignments are needed if we want to knit the Semantic Web together. To illustrate this, we have defined one class of such alignments, the ROSA rules. We have shown that these are easy to mine, and that they already comprise some interesting types of alignments. Preliminary experiments on real large-scale KBs have shown that there exist quite some alignments that go beyond simple one-to-one mappings.

However, many challenges remain: Incompleteness in the KBs means that it is challenging to make the distinction between subsumption semantics and equivalence semantics, or to distinguish between soft and hard ROSA rules. This is particularly true for complex patterns like 2-hop subsumption or value translations, where the large number of soft correlations means that we cannot yet identify meaningful mappings automatically. Moreover, ROSA rules are just one particular possibility of schema alignment. We envision that future work will describe and discover these (and other) alignments in a more systematic fashion, thus finally enabling a full interoperability between between the KBs of the Semantic Web.

## 6. REFERENCES


[1] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. G. Ives. DBpedia: A nucleus for a Web of open data. In *ISWC*, 2007.

[2] D. Aumueller, H. H. Do, S. Massmann, and E. Rahm. Schema and ontology matching with coma++. In *SIGMOD*, 2005.

[3] C. Böhm, G. de Melo, F. Naumann, and G. Weikum. Linda: distributed web-of-data-scale entity matching. In *CIKM*, 2012.

[4] A. Calì, T. Lukasiewicz, L. Predoiu, and H. Stuckenschmidt. Rule-based approaches for representing probabilistic ontology mappings. In *URSW (LNCS Vol.)*, 2008.

[5] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. H. Jr., and T. M. Mitchell. Toward an architecture for never-ending language learning. In *AAAI*, 2010.

[6] I. F. Cruz, F. P. Antonelli, and C. Stroe. Agreementmaker: Efficient matching for large real-world schemas and ontologies. *PVLDB*, 2(2), 2009.

[7] J. David, F. Guillet, and H. Briand. Association rule ontology matching approach. *Int. J. Semantic Web Inf. Syst.*, 3(2), 2007.

[8] L. Dehaspe and H. Toivonen. Discovery of frequent datalog patterns. *Data Min. Knowl. Discov.*, 3(1), 1999.

[9] O. Etzioni, M. J. Cafarella, D. Downey, S. Kok, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates. Web-scale information extraction in knowitall: (preliminary results). In *WWW*, 2004.

[10] L. Galárraga, C. Teflioudi, K. Hose, and F. M. Suchanek. Amie: association rule mining under incomplete evidence in ontological knowledge bases. In *WWW*, 2013.

[11] M. Hartung, A. Groß, and E. Rahm. Conto-diff: generation of complex evolution mappings for life science ontologies. *J. of Biomedical Informatics*, 46(1), 2013.

[12] P. Jain, P. Hitzler, A. P. Sheth, K. Verma, and P. Z. Yeh. Ontology alignment for linked open data. In *ISWC*, 2010.

[13] S. Lacoste-Julien, K. Palla, A. Davies, G. Kasneci, T. Graepel, and Z. Ghahramani. Sigma: Simple greedy matching for aligning large knowledge bases. In *KDD*, 2013.

[14] http://linkeddata.org/.

[15] J. Madhavan, P. A. Bernstein, and E. Rahm. Generic schema matching with cupid. In *VLDB*, 2001.

[16] R. J. Miller, L. M. Haas, and M. A. Hernández. Schema mapping as query discovery. In *VLDB*, 2000.

[17] S. Muggleton. Learning from positive data. In *Inductive Logic Programming Workshop*, 1996.

[18] L. Seligman, P. Mork, A. Y. Halevy, K. P. Smith, M. J. Carey, K. Chen, C. Wolf, J. Madhavan, A. Kannan, and D. Burdick. Openii: an open source information integration toolkit. In *SIGMOD*, 2010.

[19] F. M. Suchanek, S. Abiteboul, and P. Senellart. Paris: Probabilistic alignment of relations, instances, and schema. *PVLDB*, 5(3), 2011.

[20] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *WWW*, 2007.

[21] C. Tatsiopoulos and B. Boutsinas. Ontology mapping based on association rule mining. In *ICEIS (3)*, 2009.

[22] M. Technologies. The freebase project. `http://freebase.com`.

[23] O. Udrea, L. Getoor, and R. J. Miller. Leveraging data and structure in ontology integration. In *SIGMOD*, 2007.

[24] http://www.w3.org/RDF/.