NSFC-61261130588

ANR-12-IS02-002-01

Lindicle

Linked data interlinking in a cross-lingual environment
跨语言环境中语义链接关键技术研究
Liage des données dans un environnement interlingue

D4.1 Language-independent link key-based data interlinking

Coordinator: Jérôme David
With contributions from: Jérôme David, Jérôme Euzenat, Manuel Atencia

Quality reviewer:	Jérôme Euzenat
Reference:	Lindicle/D4.1/v5
Project:	Lindicle ANR-NSFC Joint project
Date:	July 6, 2015
Version:	5
State:	final
Destination:	public

Executive Summary

The goal of the Lindicle project is data interlinking. This problem can be solve using two main types of techniques: (1) similarity-based techniques computing a similarity between entities and considering as equal entities with high similarity; (2) key based-techniques finding aligned combinations of properties identifying unique entities and generating links from them. This deliverable is concerned with the latter approach.

We define link keys which extend the notion of a key to the case of different data sets. They are made of a set of pairs of properties belonging to two different classes.

We develop an approach to extract weak link keys from a pair of data sets. A weak link key holds between two classes if any resources having common values for all of these properties are the same resources. There are many potential weak link keys between two data sets. Hence we propose an algorithm that efficiently generates a small set of candidate link keys.

The quality of such candidates link key must be evaluated in order to identify those with the best properties. Depending on whether some of the, valid or invalid, links are known, we define supervised and non supervised measures for selecting the appropriate link keys. The supervised measures approximate precision and recall, while the non supervised measures are the ratio of pairs of entities a link key covers (coverage), and the ratio of entities from the same data set it identifies (discrimination).

We have experimented these techniques on two data sets, showing the accuracy and robustness of both approaches.

This deliverable is a slightly extended version of [Atencia et al. 2014].

DOCUMENT INFORMATION

Project number	ANR-NSFC Joint project	**Acronym**	Lindicle
Full Title	跨语言环境中语义链接关键技术研究 Linked data interlinking in a cross-lingual environment Liage des données dans un environnement interlingue		
Project URL	`http://lindicle.inrialpes.fr/`		
Document URL			

Deliverable	**Number**	4.1	**Title**	Language-independent link key-based data interlinking
Work Package	**Number**	4	**Title**	Cross-lingual data interlinking

Date of Delivery	**Contractual**	M18	**Actual**	2015-03-20
Status	final		final ☒	
Nature	prototype ☐ report ☒ dissemination ☐			
Dissemination level	public ☒ consortium ☐			

Authors (Partner)	Jérôme David, Jérôme Euzenat, Manuel Atencia			
Resp. Author	**Name**	Jérôme David	**E-mail**	Jerome.David@inria.fr
	Partner	INRIA		

Abstract (for dissemination)	Links are important for the publication of RDF data on the web. Yet, establishing links between data sets is not an easy task. We develop an approach for that purpose which extracts weak link keys. Link keys extend the notion of a key to the case of different data sets. They are made of a set of pairs of properties belonging to two different classes. A weak link key holds between two classes if any resources having common values for all of these properties are the same resources. An algorithm is proposed to generate a small set of candidate link keys. Depending on whether some of the, valid or invalid, links are known, we define supervised and non supervised measures for selecting the appropriate link keys. The supervised measures approximate precision and recall, while the non supervised measures are the ratio of pairs of entities a link key covers (coverage), and the ratio of entities from the same data set it identifies (discrimination). We have experimented these techniques on two data sets, showing the accuracy and robustness of both approaches.
Keywords	data interlinking, linked data, link key, candidate link key, coverage, dissimilarity

Version Log			
Issue Date	**Rev No.**	**Author**	**Change**
20/09/2011	1	J. Euzenat	A simple entry example
25/03/2015	2	J. David	Put the ECAI paper in the template
25/03/2015	3	J. David	Added reference to the published version
02/07/2015	4	J. Euzenat	Set final
06/07/2015	5	J. Euzenat	Changed numbering

Table of Contents

1. Data interlinking

Linked (open) data is the publication of data by using semantic web technologies [Heath and Bizer 2011]: data is expressed in RDF, eventually described by an ontology and linked to other data sets through statements identifying equivalent resources. Usually, such statements are asserted through triples between equivalent elements using the owl:sameAs predicate. We call them sameAs links, or simply links. They are a very important part of linked data.

It is thus critical to be able to generate relevant links between data sources, what is called data interlinking. We consider the setting in which users want to interlink data sets. They are able to identify equivalent or overlapping classes of objects (this can also be provided through an ontology alignment) and they may be able to provide some examples of correct and incorrect links. Hence, we design algorithms which, from a pair of classes in two data sets and optionally two sample sets of owl:sameAs and owl:differentFrom links, are able to generate a set of owl:sameAs links.

Among the possible ways to produce links is the identification of keys: sets of properties whose values characterize unique individuals. We consider here link keys, i.e., keys that span across two data sets and which identify unique individuals only for the available data. A link key between a pair of classes is characterized by pairs of corresponding properties $\{\langle p_1, q_1\rangle, \ldots \langle p_n, q_n\rangle\}$ which together identify unique entities. Weak link keys are required to be keys only on the identified entities. A valid link key allows straightforwardly to generate links since entities bearing common values for these properties are the same individual.

Our method first relies on generating all candidate link keys, i.e., maximal sets of property pairs for which there is at least two instances sharing a value. Since there are several candidate link keys, it is necessary to evaluate them and select the most promising ones. For that purpose, we define measures of discriminability and coverage for non supervised link key extraction and approximation of precision and recall for the supervised case. We show through experiments that they are good approximations of precision and recall and that they are robust to data alteration.

So, after defining some notation (§2) and discussing prior art (§3), we define more precisely the notion of a weak link key and provide an algorithm for generating candidate link keys (§4). Such an algorithm is able to drastically reduce the number of candidate link keys. Then we provide measures for assessing their quality (§5). We evaluate these measures and their robustness through an experiment based on actual data (§6).

2. Notation and problem statement

Consider that we want to link two data sets D and D' complying to specific ontologies O and O', respectively. We assume that the ontologies are description logic TBoxes and the data sets are ABoxes containing only $c(a)$ and $p(a, a')$ axioms. The structure $\mathcal{O} = \langle O, D \rangle$ will be called an ontology.

Let us assume that the vocabularies of O and O' are disjoint. We use the letters c, p, and a, with sub- or super-scripts, to denote class and property expressions, and individuals names of $\mathcal{O}$, respectively, and we retain the letters d, q, b for those of $\mathcal{O}'$.

The general task carried out by data interlinking is, given two data sets D and D', to find one set of relations between individuals of D and D'. We restrict ourselves to finding equality statements between named individuals a and b from each data sets denoted by $\langle a, \mathsf{owl{:}sameAs}, b \rangle$ or the pair $\langle a, b \rangle$. A set of such pairs is called a link set and denoted by L.

We consider the subproblem of finding a set of links L between instances of c and d from O and O', given a set of links L_0 between D and D' which does not contain links between c and d. L_0 is used for comparing property values of instances of c and d. Links may be generated in an iterative way: first links are generated for classes having only owl:DatatypeProperties, then the generated links may be used for generating links based on owl:ObjectProperties involving these classes. In the following, $p(a) \cap q(b)$ means $\{x | \mathcal{O}, L_0 \models p(a, x) \text{ and } \mathcal{O}', L_0 \models q(b, x)\}$.

3. Related works

There has been a lot of work recently on data interlinking [Ferrara et al. 2011] in part inspired by the work on record linkage in databases [Elmagarmid et al. 2007].

Usually, one defines a similarity between resources based on their property values and declares an owl:sameAs link between those which are highly similar [Ngonga Ngomo and Auer 2011]. The difficult part is to define the similarity and what is “highly”. So, some works use machine learning in order to set similarity parameters and thresholds from sample links [Ngonga Ngomo and Lyko 2012; Isele and Bizer 2013]. Similarities do not attempt at defining what makes identity, but rather require that as many features as possible be close enough. There is no explicit assertion of what makes identity.

Keys in databases are sets of attributes (columns in a table) such that two different individuals cannot have the same values for these attributes. These are sufficient conditions for being the same. Hence, interlinking may be based on keys.

In database, the extraction of keys has been mainly studied through the discovery of functional dependencies. According to [Yao and Hamilton 2008] there are three kinds of methods for finding functional dependencies in data: the candidate generate-and-test methods [Huhtala et al. 1999; Yao and Hamilton 2008; Novelli and Cicchetti 2001], minimal cover methods [Flach and Savnik 1999; Wyss et al. 2001; Sismanis et al. 2006], and formal concept analysis methods [Lopes et al. 2002; Baixeries 2004]. Methods following the first approach use level-wise search for testing combinations of attributes. For each set of attributes, these methods generate a partition of the instances into equivalence classes in which all instances share the same values for the attributes. The search space is reduced using pruning rules with differ according to the methods.

Two methods have been proposed for discovering keys in RDF data sets. KD2R [Pernelle et al. 2013] is a method based on the Gordian algorithm [Sismanis et al. 2006] which derives keys from the maximal non keys.

The pseudo-key extraction method proposed by [Atencia et al. 2012] follows the candidate generate-and-test approach. Since it has been designed for RDF data, it differs from the database methods, considering that properties in the RDF model are not total functions like attributes in the relational model. This makes optimizations and pruning rules proposed by [Huhtala et al. 1999] and [Yao and Hamilton 2008] not valid for RDF data.

So far, keys were extracted in each data set independently without considering their interactions.

4. Extracting candidate link keys

The approach presented here extracts directly what we call link keys. Link keys are adaptations of keys across different data sets. These link keys are used for generating links, because, like keys, they find equivalent objects. In principle, there are many candidate link keys. Hence we present algorithms for exploring them efficiently.

4.1 Weak link keys and candidate link keys

Like alignments, link keys [Euzenat and Shvaiko 2013] are assertions across ontologies and are not part of a single ontology. They are sets of corresponding properties from both ontologies which, for a pair of corresponding classes, identify equivalent individuals. Various sorts of link keys may be defined by requiring that they be keys on some parts of the datasets. Weak link keys do only have to be keys for the set of linked entities. Unlike keys, they do not guarantee that the properties used for linking them identify a unique entity per initial class. Hence, they may generate multiple links between entities.

Definition 1 (Weak link key) *A* weak link key *between two classes c and d of ontologies $\mathcal{O}$ and $\mathcal{O}'$, respectively, is a set of property pairs*

$$\{\langle p_1, q_1\rangle, \ldots, \langle p_k, q_k\rangle\}$$

such that $p_1, \ldots, p_k$ are properties in $\mathcal{O}$ and $q_1, \ldots q_k$ are properties in $\mathcal{O}'$, and $\forall a; \mathcal{O} \models c(a)$, $\forall b; \mathcal{O}' \models d(b)$, if $\forall i \in 1, \ldots, k$, $p_i(a) \cap q_i(b) \neq \varnothing$, then $\langle a, \text{owl:sameAs}, b\rangle$ holds.

Link keys are defined here with respect to the sharing of a value for a property. They may also rely on the equality between property values. The two notions are equivalent for functional properties. Equality of property values can be seen as too restrictive, especially across datasets. However, this problem can be partially solved by using methods such as value clustering or normalization.

Because they are sufficient conditions for two instances to denote the same individual, they can be used for generating links: any pairs of instances from the two classes which satisfy the condition must be linked. We denote by $L_{D,D'}(r)$ the set of links that are generated by a (candidate) link key r between data sets D and D'.

We present here a method to extract a superset of weak link keys instantiated on the current data. Then, we show how to select the relevant ones by assessing their quality through several selection criteria.

The approach generates all candidate link keys. We call candidate link key a set of property pairs which is maximal for at least one link it would generate if used as a link key.

Definition 2 (Candidate link key) *Given two ontologies $\mathcal{O}$ and $\mathcal{O}'$, $\{\langle p_1, q_1\rangle, \ldots, \langle p_k, q_k\rangle\}$ is a candidate link key for the pair of classes $\langle c, d\rangle$ iff $\exists a, b$ such that*

- *$\forall i \in 1 \ldots k, p_i(a) \cap q_i(b) \neq \varnothing$, and*
- *$\forall \langle p, q\rangle \notin \{\langle p_1, q_1\rangle, \ldots, \langle p_k, q_k\rangle\}$, $p(a) \cap q(b) = \varnothing$.*

This simply means that we only consider as candidates sets of pairs of properties that would generate at least one link that would not be generated by any larger set.

Table 4.1 shows an example of candidate link keys that hold between data sets D and D'. For instance, the set $\{\langle p_2, q_2\rangle\}$ that would generate links $\langle a_1, b_1\rangle$, $\langle a_1, b_2\rangle$ and $\langle a_2, b_2\rangle$ is not a

D		D'		Candidate link keys
$\langle a_1, p_1, v_1\rangle$	$\langle a_2, p_2, v_4\rangle$	$\langle b_1, q_1, v_1\rangle$	$\langle b_2, q_2, v_2\rangle$	$\{\langle p_1, q_1\rangle, \langle p_2, q_2\rangle\}$
$\langle a_1, p_2, v_2\rangle$	$\langle a_2, p_3, v_5\rangle$	$\langle b_1, q_2, v_2\rangle$	$\langle b_2, q_2, v_4\rangle$	$\{\langle p_2, q_2\rangle, \langle p_3, q_3\rangle\}$
	$\langle a_2, p_1, v_3\rangle$	$\langle b_2, q_1, v_1\rangle$	$\langle b_2, q_3, v_5\rangle$	

Table 4.1: Two sets of triples and the corresponding candidate link keys.

indexDataset(D):	indexDataset(D'):	PropertyAgreement
$v_1 : \{\langle a_1, p_1\rangle\}$	$v_1 : \{\langle b_1, q_1\rangle, \langle b_2, q_1\rangle\}$	$\langle a_1, b_1\rangle \to \{\langle p_1, q_1\rangle, \langle p_2, q_2\rangle\}$
$v_2 : \{\langle a_1, p_2\rangle\}$	$v_2 : \{\langle b_1, q_2\rangle, \langle b_2, q_2\rangle\}$	$\langle a_1, b_2\rangle \to \{\langle p_1, q_1\rangle, \langle p_2, q_2\rangle\}$
$v_3 : \{\langle a_2, p_1\rangle\}$		
$v_4 : \{\langle a_2, p_2\rangle\}$	$v_4 : \{\langle b_2, q_2\rangle\}$	$\langle a_2, b_2\rangle \to \{\langle p_2, q_2\rangle, \langle p_3, q_3\rangle\}$
$v_5 : \{\langle a_2, p_3\rangle\}$	$v_5 : \{\langle b_2, q_3\rangle\}$	

Table 4.2: Indexes computed by Algorithms 1 and 2 on the example of Table 4.1.

candidate link key because these links can also be generated by supersets $\{\langle p_1, q_1\rangle, \langle p_2, q_2\rangle\}$ and $\{\langle p_2, q_2\rangle, \langle p_3, q_3\rangle\}$. Instead of the $2^{3\times 3} = 512$ possible sets of property pairs, there are only 2 candidate link keys.

Generating and checking all combinations of sets of property pairs is not suitable due the exponential size of search space. In order to extract them efficiently, we rely on several indexation steps.

4.2 Extraction algorithms

The extraction procedure is given by Algorithm 2. It first indexes, for each data set, the set of subject-property pairs sharing at least one value. Then it calls Algorithm 1 which iterates over these indexes in order to generate another index associating each pair of subjects to the maximal sets of properties on which they agree. From the values contained in this last index, we can easily derive the set of candidate link keys and count their occurrence.

In the worst case, if all subjects have the same predicate-object pairs, time complexity is $O(n^2)$. In any case, we have to browse the whole datasets which is in $O(n)$. The practical complexity depends on the number of subject-predicate pairs per object. Space complexity is $O(n)$, i.e., the sum of the triples in both datasets.

Algorithm 1 Maximal property pairs agreement.

Input: Two o→{sp} indexes, idx and idx'
Output: An $\langle s, s' \rangle \rightarrow \{\langle p, p' \rangle\}$ index, i.e., the maximal agreeing property pairs for each subject pair

```
function propertyAgreement(idx, idx')
    residx ← ∅
    for all k belonging to both idx and idx' keys do
        for all ⟨s, p⟩ ∈ idx[k] do
            for all ⟨s', p'⟩ ∈ idx'[k] do
                residx[⟨s, s'⟩] = residx[⟨s, s'⟩] ∪ {⟨p, p'⟩}
            end for
        end for
    end for
    return residx
end function
```

Algorithm 2 Candidate link key extraction.

Input: Two data sets D and D'
Output: The set of candidate link keys instanciated between D and D' and their occurency

```
function candidateLinkkeyExtraction(D, D')
    idx ← indexDataset(D)
    idx' ← indexDataset(D')
    agreementIdx ← propertyAgreement(idx, idx')
    for all {⟨p1, p'1⟩, ..., ⟨pn, p'n⟩} ∈ agreementIdx values do
        linkkeys[{⟨p1, p'1⟩, ..., ⟨pn, p'n⟩}] + +
    end for
    return linkkeys
end function
```

5. Weak link key selection measures

Algorithm 2 extracts candidate link keys. These candidates are not necessarily valid link keys. In order to compare candidates, we propose measures for assessing their quality. Two important and classical quality criteria are the correctness and the completeness of the links that a candidate link key generates.

A good measure for assessing correctness a priori should approximate the ranking of candidate link keys given a posteriori by its precision. In the same manner, a good measure for completeness should approximate that of candidate link keys given by recall.

In the following, we propose measures that assess these two criteria according to two scenarios: supervised and non supervised.

5.1 Measures for supervised selection: exploiting owl:sameAs and owl:differentFrom links

In the supervised case, it is possible to directly approximate precision and recall on the examples. In some cases, some owl:sameAs and/or owl:differentFrom links across the two data sets are available. This can be used to compute estimations of precision and recall. Let be L^+, a set of owl:sameAs links (positive examples) and L^-, a set of owl:differentFrom links (negative examples), the set $L^+ \cup L^-$ can be considered as a sample. Hence, it is possible to evaluate the behavior of $L_{D,D'}(r)$ on this sample, i.e., compute the precision and recall of $L_{D,D'}(r) \cap (L^+ \cup L^-)$ with respect to L^+.

The quality of a candidate link key r can be evaluated by the two classical correctness and completeness measures restricted to the sample. They are defined as follows:

Definition 3 (Relative precision and recall)

$$\widehat{precision}(r, L^+, L^-) = \frac{|L^+ \cap L_{D,D'}(r)|}{|(L^+ \cup L^-) \cap L_{D,D'}(r)|}$$

$$\widehat{recall}(r, L^+) = \frac{|L^+ \cap L_{D,D'}(r)|}{|L^+|}$$

When the sample only consists of owl:sameAs links, i.e., $L^- = \varnothing$, $\widehat{\text{precision}}$ is not relevant. In that situation, we can artificially generate owl:differentFrom links by partially closing the owl:sameAs links. To that extent the following rule can be used: for each $\langle a, b\rangle \in L^+$, we assume $\langle a, x\rangle \in L^-$ for all x such that $\langle a, x\rangle \notin L^+$ and $\mathcal{O}' \not\models \langle b, \text{owl:sameAs}, x\rangle$, and $\langle y, b\rangle \in L^-$ for all y such that $\langle y, b\rangle \notin L^+$ and $\mathcal{O} \not\models \langle a, \text{owl:sameAs}, y\rangle$.

Given precision and recall, F-measure may be computed in the usual way ($F = \frac{2PR}{P+R}$).

5.2 Measures for unsupervised selection

In case no sameAs link across data sets is available, we can only rely on local knowledge for assessing the correctness of potentially generated links.

Assuming that, in each data set, instances are distinct, then there should not be more than one link involving one instance. So, a first measure of quality is the capability of discriminating between instances, i.e., that the link set is one-to-one. We then propose to measure the correctness of a candidate link key by its discriminability which measures how close the links generated by a candidate link key are to a one-to-one mapping.

Definition 4 (Discriminability)

$$disc(r) = \frac{min(|\{a|\langle a,b\rangle \in L_{D,D'}(r)\}|, |\{b|\langle a,b\rangle \in L_{D,D'}(r)\}\}|)}{|L_{D,D'}(r)|}$$

It is equal to 1, when links are a perfect one-to-one mapping and is lower-bounded by $(|\{a|\langle a,b\rangle \in L_{D,D'}(r)\} \times \{b|\langle a,b\rangle \in L_{D,D'}(r)\}|$.

For assessing the completeness of a candidate link key, we rely on the intuition that the more instances linked by a candidate link key, the more complete the candidate link key is. Then, the coverage of a candidate link key is defined as the proportion of instances of both classes that could be linked.

Definition 5 (Coverage)

$$cov(r, D, D') = \frac{|\{a|\langle a,b\rangle \in L_{D,D'}(r)\} \cup \{b|\langle a,b\rangle \in L_{D,D'}(r)\}|}{|\{a|c(a) \in D\} \cup \{b|d(b) \in D'\}|}$$

The coverage measure always favors the most general link key candidates: if $r' \subseteq r$, then $L_{D,D'}(r) \subseteq L_{D,D'}(r')$, so $cov(r', D, D') \geq cov(r, D, D')$. In particular, when we have two candidates link keys r and r' such that $r' \subseteq r$, r has a greater coverage value than r', even if there is only few links specifically generated by r but not by r'.

Using both coverage and discriminability strikes a balance between the completeness and generality of candidate link keys. They can be aggregated by harmonic means just like F-measure does for precision and recall.

6. Experimental evaluation

The accuracy and robustness of the proposed quality measures have been experimentally evaluated[1]. Our goal is to assess that proposed measures help to select the best candidate link keys in term of precision and recall. To that extent, we performed two series of experiments evaluating discriminability and coverage on the one hand, and partial precision and recall on the other hand. Both series of experiments use on the same data sets.

6.1 Data sets

We have experimented with geographical data from INSEE and GeoNames data sets[2]. INSEE comprehends data about French geography, economy and society, whereas GeoNames is a world-wide geographical database. We have concentrated on the fragment of INSEE which corresponds to geographical data (available as an RDF dump), and the fraction of GeoNames corresponding to French geographical data (retrieved by querying in the whole data set individuals with FR as value for the property countryCode),[3] for which there exist owl:sameAs links to INSEE. The INSEE data set covers 36700 instances; GeoNames contains 36552 instances. The reference link set maps each instance of commune in GeoNames to one and only one commune in INSEE. So, 448 INSEE instances are not linked.

Our objective is to extract candidate link keys between classes representing the French municipalities of these two data sets and evaluate them according to the different selection criteria.

Our objective is to automatically reproduce these links by exploiting keys and an alignment between INSEE's and GeoNames' ontologies computed manually.

In both data sets, these instances are also described as part of broader administrative regions which are themselves described within each data set. In the experiments, links between these administrative regions are part of L_0.

6.2 Experimental protocol

Two series of test are performed respectively for the unsupervised and supervised selection measures.

[1]All the material allowing to reproduce experiments is available at `http://melinda.inrialpes.fr/linkkey/`

[2]`http://www.insee.fr/`, `http://www.geonames.org/`

[3] This restriction was made due to time constraints and are not expected to change the results. They enter in our setting where users identify the two classes to be compared. We omit to use prefixes as the two data sets are written in distinct languages (French and English).

Entities	**INSEE$_{\textbf{GEO}}$**	**GeoNames$_{\textbf{FR}}$**
Datatype properties	3	9
Object properties	5	14
Instances	36,700	36,552

Table 6.1: Number of instances and properties of the fragments of INSEE and GeoNames considered in our experiments.

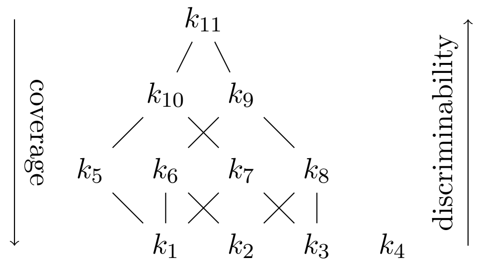

Figure 6.1: The set of candidate link keys. This is a subset of the powerset lattice $\langle 2^{P \times P'}, \subseteq \rangle$, but not a lattice.

For the the first series, candidate link keys between the two data sets are extracted with the given algorithm and the ranking given by discriminability and coverage are compared to those given by precision and recall.

Then, a set of derivative tests simulating perturbed interlinking scenarios are performed. They extract and evaluate candidates on altered versions of the data sets. Different kinds of alterations are performed: (1) triples removal: we randomly suppress some triples; (2) values scrambling: we randomly scramble the object of some triples; (3) instance removal: instances are randomly removed by suppressing all triples involving them. For each series of tests, the probability of degradation varies from 0 to 0.9 by step of 0.1.

The second series of tests evaluates the behavior of supervised selection measures when the size of the positive examples varies. To that extent, the probability that a owl:sameAs link from the reference be in L^+ varied from 0 to 0.9 by step of 0.1. L^- is generated from owl:sameAs links according to Section 5.1.

For both series, 10 runs are performed and their results averaged.

6.3 Results

Unsupervised selection measures There are 7 property pairs that have been found in candidate link keys. They are:

$P_5 = \langle \text{codeINSEE}, \text{population} \rangle$ $\quad P_1 = \langle \text{nom}, \text{name} \rangle$

$P_6 = \langle \text{codeCommune}, \text{population} \rangle$ $\quad P_2 = \langle \text{nom}, \text{alternateName} \rangle$

$P_3 = \langle \text{subdivisionDe}, \text{parentFeature} \rangle$ $\quad P_7 = \langle \text{nom}, \text{officialName} \rangle$

$P_4 = \langle \text{subdivisionDe}, \text{parentADM3} \rangle$

The algorithms extracted eleven candidate link keys which are detailed in Table 6.2. Their relations are provided in Figure 6.1

Among the 11 candidate link keys, 8 have a precision greater or equals to 0.8. These candidates are k_1 and its specializations and k_8. Three have good recall, for all the others recall is very low, i.e., less than 0.3%. Only k_1 and k_7 have a good F-measure, with a clear superiority of the last one. The first candidate does not have a perfect precision because there are different communes in France with the same name, but these communes can be distinguished by the arrondissement they belong to. As an example, Bully may refer to three communes: Bully in Dieppe, Bully in Lyon, and Bully in Roanne[4].

Coverage values are strongly correlated to those given by recall. This confirms our expectation. There is also a good correlation between discriminability and precision, except

[4]Here we refer to the arrondissements, and not the homonymous cities.

Candidate link keys			Quality estimators			Reference			10% of reference		
name	**pairs**	**# links**	**disc.**	**hmean**	**cov.**	**prec.**	**F-m.**	**rec.**	**prec.**	**F-m.**	**rec.**
k_1	$\{P_1\}$	45 654	0.801	0.889	0.998	0.8	0.889	1	0.68	0.809	0.999
k_2	$\{P_2\}$	19	0.79	0.002	0.001	0.579	0.002	0.001	0.434	0	0
k_3	$\{P_3, P_4\}$	5 331 816	0.007	0.014	0.975	0.007	0.014	0.977	0.004	0.008	0.978
k_4	$\{P_5, P_6\}$	354	0.984	0.02	0.01	0	0	0	0	0	0
k_5	$\{P_7, P_1\}$	44	0.887	0.004	0.002	0.887	0.004	0.002	0.918	0.002	0.001
k_6	$\{P_2, P_1\}$	11	0.819	0.002	0.001	0.819	0.002	0.001	0.778	0	0
k_7	$\{P_3, P_4, P_1\}$	35 689	1	0.987	0.975	1	0.988	0.976	1	0.988	0.977
k_8	$\{P_3, P_2, P_4\}$	11	1	0.002	0.001	1	0.002	0.001	1	0	0
k_9	$\{P_3, P_2, P_4, P_1\}$	9	1	0.002	0.001	1	0.002	0.001	1	0	0
k_{10}	$\{P_3, P_7, P_4, P_1\}$	39	1	0.004	0.002	1	0.004	0.002	1	0.002	0.001
k_{11}	$\{P_3, P_7, P_2, P_4, P_1\}$	2	1	0	0	1	0	0	1	0	0
Correlations to the reference											
Pearson ρ			0.645	1	1				0.99	0.999	1
Kendall τ_b (all p-values < 0.01)			0.778	0.695	0.723				1	0.911	0.911

Table 6.2: Candidate link keys and quality estimation in the non supervised case and the supervised case with 10% links.

for the candidate $k_4 = \{\langle \text{codeINSEE}, \text{population} \rangle, \langle \text{codeCommune}, \text{population} \rangle\}$. Indeed, codeINSEE and codeCommune are two equivalent identifiers of French communes. They are obviously not related to the population property which is the number of inhabitants, but 354 pairs of instances share the same values for this properties. This candidate link key has a good discriminability because its properties are themselves discriminant. This shows that the discriminability alone is not sufficient.

Thus, the best link key given by F-measure is not one of the most simple rule like k_1, but one with an intermediate position in the graph of Figure 6.1: k_7. This is correctly predicted by the harmonic means of coverage and discrimination. Here again, Pearson value correlation is optimal, while the Kendall rank correlation is hurt by k_4's high rank in discriminability. k_7 generates 35689 links out of the 36546 expected links and all these links are correct. The missing links are due to missing links between parent regions in L_0 and differences in spelling, e.g., Saint-Étienne-de-Tulmont vs. Saint-Etienne-de-Tulmont. This could be improved by using a priori normalization or less strict constraints than inclusion.

Robustness The number of generated link key candidates is stable when instances are removed or triples are scrambled[5] but it increases when triples are removed. It reaches a maximum of 33 candidates at 30% of triple removed, then it decreases. Indeed, when triples are removed some pairs of instances agree on less properties and then more general candidates are generated. The majority of these candidates still have a very low coverage (and recall).

Figure 6.2 shows that when alterations increase, the discriminability remains stable for the majority of link keys candidates. Candidates showing less smooth curves are candidate link keys generating few links, i.e., with low coverage. For candidates k_1 and k_3, two candidates having good recall but not perfect precision, we observe that discriminability increases more rapidly when removed triples or instances increase. These two candidates have more stable discriminability values when objects of triples are scrambled. For candidates having not a very low coverage, these tests show that discriminability is robust until at least 50% of alterations.

Coverage is less robust to alterations. When link key candidates generate one-to-one link sets, the coverages values decreases when alterations increase. On the instance removal test,

[5] In that last case, only one more candidate is generated.

we observe a linear decrease for candidate link keys which generates one-to-one mapping. For k_3 which tends to a many-to-many mapping, the coverage curve is stable. This is in line with Definition 5 (coverage). Indeed, if a link key is one-to-one, each time one instance is suppressed, one link will be suppressed. Hence the numerator is decreased of two units while the denominator is decreased by only one unit. In the case of the cartesian product, these two quantities will decrease at the same speed. In the case of triple removal or scrambling, the probability that an alteration removes a link is higher than that it removes an instance. Then, the coverage measure decreases even faster when the probability of alteration increases.

However, we observe that the order of link key candidates given by coverage is preserved in most of the cases. For instance, rule k_1 has always better coverage than k_7. This behaviour shows that coverage is a good estimator of the link key candidates ranking given by recall.

Supervised selection measures When the amount of reference owl:sameAs links varies, the precision value is constant for the majority of link key candidates (7/11) (see Figure 6.3). These candidates are those having extreme precision value, i.e., either 1 or 0. For the other four candidates, the precision slowly and linearly decreases from 100% to 50% of owl:sameAs. Under 50% of reference links, three of these candidates do not have a stable trend anymore. This is caused by the low number of links they generate. The last candidate, k_1, which generates much more links, has a more stable precision. The recall values are perfectly robust to the variation of sample links.

The rankings given by precision and recall remain the same when the sample links decrease. It is thus possible to select good link key candidates when we have only a sample of reference owl:sameAs links (Table 6.2 provides the estimation with 10%). This behavior has also been shown in ontology matching [Ritze and Paulheim 2011]

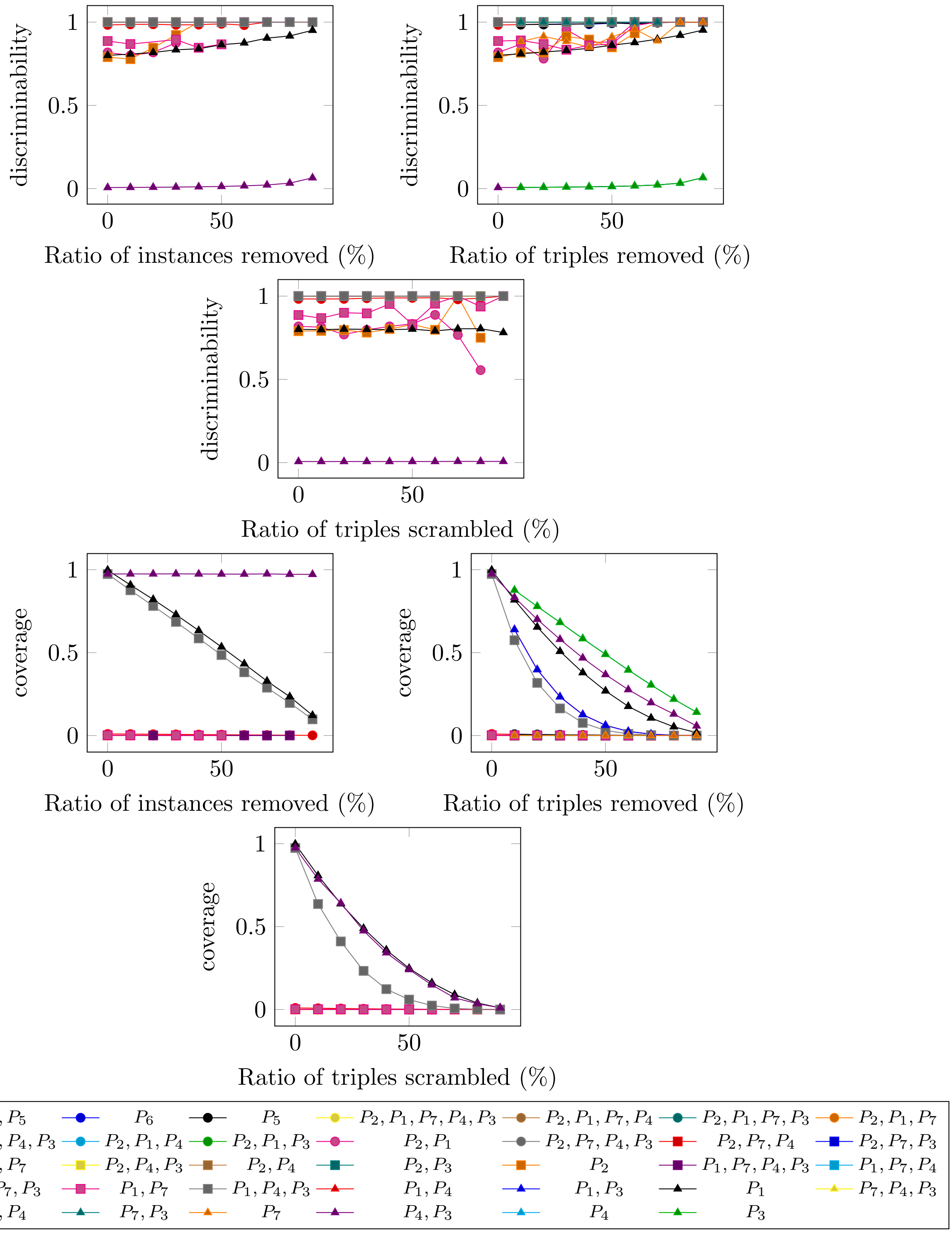

Figure 6.2: Evolution of discriminability and coverage measures in function of the degradation of data sets. A curve stops when the confidence is not computable, i.e., there is no owl:sameAs link generated by the candidate link key. When less instances are available, more candidates are generated.

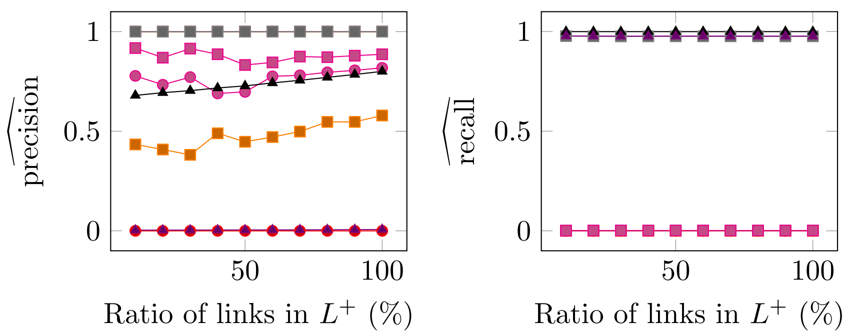

Figure 6.3: Evolution of precision and recall measures in function of the ratio of owl:sameAs links in L^+. A curve stops when confidence is not computable, i.e., there is no owl:sameAs link generated by the candidate link key (Legend as of Figure 6.2).

7. Conclusions and perspectives

Link keys are sets of pairs of properties characterizing equivalence. They can be used for generating links across RDF data sets. We provided an algorithm for enumerating a restricted number of link key candidates and provided measures for evaluating the quality of these candidates. We experimentally observed that these measures select the best candidate in both the supervised and non supervised case. They are also robust to mistakes in the data sets and sample links.

Other measures, such as consistency, may be used in addition but they require expressive alignments which are not often available.

This setting is well suited for finding one-to-one linksets. Establishing similar measures for many-to-many correspondences is an open question.

Bibliography

Atencia, Manuel, Jérôme David, and François Scharffe (2012). "Keys and pseudo-keys detection for web datasets cleansing and interlinking". en. In: *Proc. 18th international conference on knowledge engineering and knowledge management (EKAW), Galway (IE)*, pp. 144–153 (cit. on p. 7).

Atencia, Manuel, Jérôme David, and Jérôme Euzenat (2014). "Data interlinking through robust linkkey extraction". In: *Proc. 21st european conference on artificial intelligence (ECAI), Praha (CZ)*, pp. 15–20 (cit. on p. 2).

Baixeries, Jaume (2004). "A formal concept analysis framework to mine functional dependencies". In: *Proceeding of the Workshop on Mathematical Methods for Learning*. Como, Italy (cit. on p. 7).

Elmagarmid, Ahmed, Panagiotis Ipeirotis, and Vassilios Verykios (2007). "Duplicate Record Detection: A Survey". In: *IEEE Transactions on knowledge and data engineering* 19.1, pp. 1–16 (cit. on p. 7).

Euzenat, Jérôme and Pavel Shvaiko (2013). *Ontology matching*. en. 2nd. Heidelberg (DE): Springer-Verlag. 520 pp. (cit. on p. 8).

Ferrara, Alfio, Andriy Nikolov, and François Scharffe (2011). "Data Linking for the Semantic Web". In: *International Journal of Semantic Web and Information Systems* 7.3, pp. 46–76 (cit. on p. 7).

Flach, Peter and Iztok Savnik (1999). "Database dependency discovery: a machine learning approach". In: *AI Communication* 12.3, pp. 139–160. ISSN: 0921-7126 (cit. on p. 7).

Heath, Tom and Christian Bizer (2011). *Linked Data: Evolving the Web into a Global Data Space*. Morgan & Claypool (cit. on p. 5).

Huhtala, Ykä, Juha Kärkkäinen, Pasi Porkka, and Hannu Toivonen (1999). "TANE: An Efficient Algorithm for Discovering Functional and Approximate Dependencies". In: *The Computer Journal* 42.2, pp. 100–111 (cit. on p. 7).

Isele, Robert and Christian Bizer (2013). "Active learning of expressive linkage rules using genetic programming". In: *Journal of web semantics* 23, pp. 2–15 (cit. on p. 7).

Lopes, Stéphane, Jean-Marc Petit, and Lotfi Lakhal (2002). "Functional and approximate dependency mining: database and FCA points of view". In: *Journal of Experimental & Theoretical Artificial Intelligence* 14.2-3, pp. 93–114 (cit. on p. 7).

Ngonga Ngomo, Axel-Cyrille and Sören Auer (2011). "LIMES: A Time-Efficient Approach for Large-Scale Link Discovery on the Web of Data". In: *Proc. 22nd International Joint Conference on Artificial Intelligence (IJCAI)*. Barcelona (ES), pp. 2312–2317 (cit. on p. 7).

Ngonga Ngomo, Axel-Cyrille and Klaus Lyko (2012). "EAGLE: Efficient Active Learning of Link Specifications Using Genetic Programming". In: *Proc. 9th ESWC, Heraklion (GR)*, pp. 149–163 (cit. on p. 7).

Novelli, Noel and Rosine Cicchetti (2001). "Functional and embedded dependency inference: a data mining point of view". In: *Information Systems* 26.7, pp. 477–506 (cit. on p. 7).

Pernelle, Nathalie, Fatiha Saïs, and Danai Symeounidou (2013). "An Automatic Key Discovery Approach for Data Linking". In: *Journal of Web Semantics* 23, pp. 16–30 (cit. on p. 7).

Ritze, Dominique and Heiko Paulheim (2011). "Towards an automatic parameterization of ontology matching tools based on example mappings". In: *Proc. 6th International Workshop on Ontology Matching* (cit. on p. 16).

Sismanis, Yannis, Paul Brown, Peter Haas, and Berthold Reinwald (2006). "GORDIAN: efficient and scalable discovery of composite keys". In: *Proc. 32nd international conference on very large databases (VLDB)*, pp. 691–702 (cit. on p. 7).

Wyss, Catharine, Chris Giannella, and Edward Robertson (2001). "FastFDs: A Heuristic-Driven, Depth-First Algorithm for Mining Functional Dependencies from Relation Instances - Extended Abstract". In: *Proc. 3rd International Conference on Data Warehousing and Knowledge Discovery*. London (UK), pp. 101–110. ISBN: 3-540-42553-5 (cit. on p. 7).

Yao, Hong and Howard Hamilton (2008). "Mining functional dependencies from data". In: *Data Mining Knowledge Discovery* 16.2, pp. 197–219. ISSN: 1384-5810 (cit. on p. 7).