Selection of representative subsets of link key candidates

Nacira Abbas, Alexandre Bazin, Jérôme David, Amedeo Napoli

Laboratoire d'Informatique de Grenoble
Montbonnot, France
Firstname.Lastname@inria.fr
https://moex.inria.fr

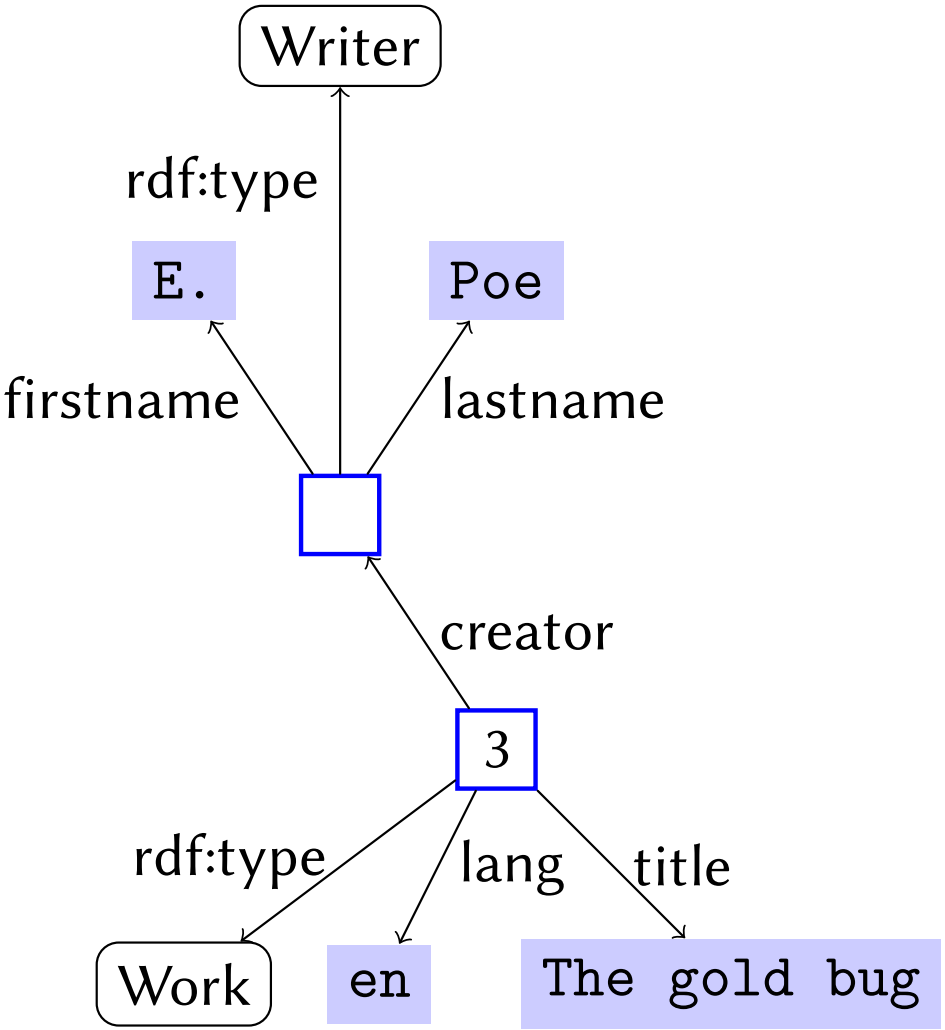
Atelier Decade, le 28/06/2022, St Etienne

Partly funded by Elker ANR project (ANR-17-CE23-0007-01)
Most of this work is from the PhD of Nacira Abbas (Loria, Nancy)

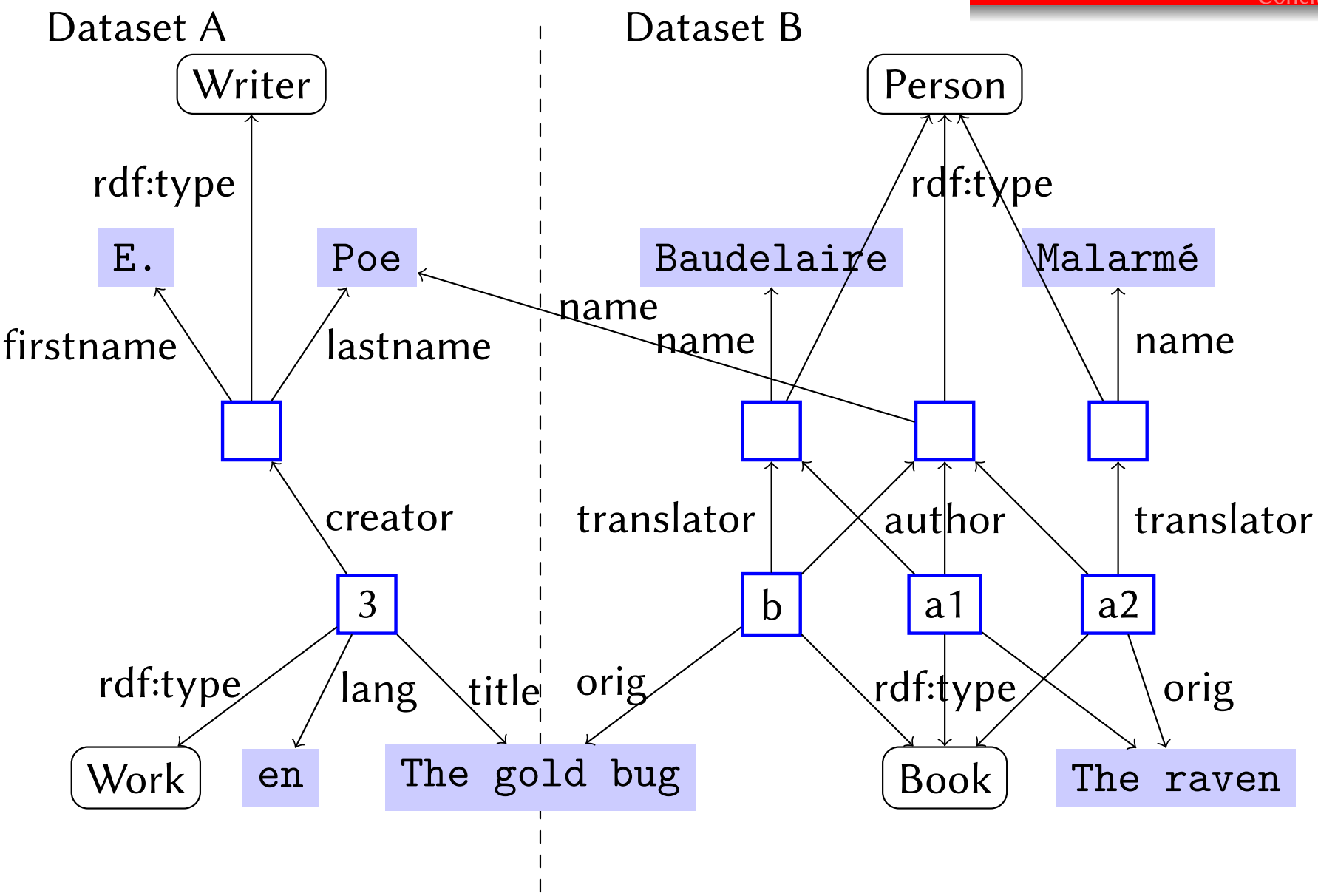
Outline

Outline

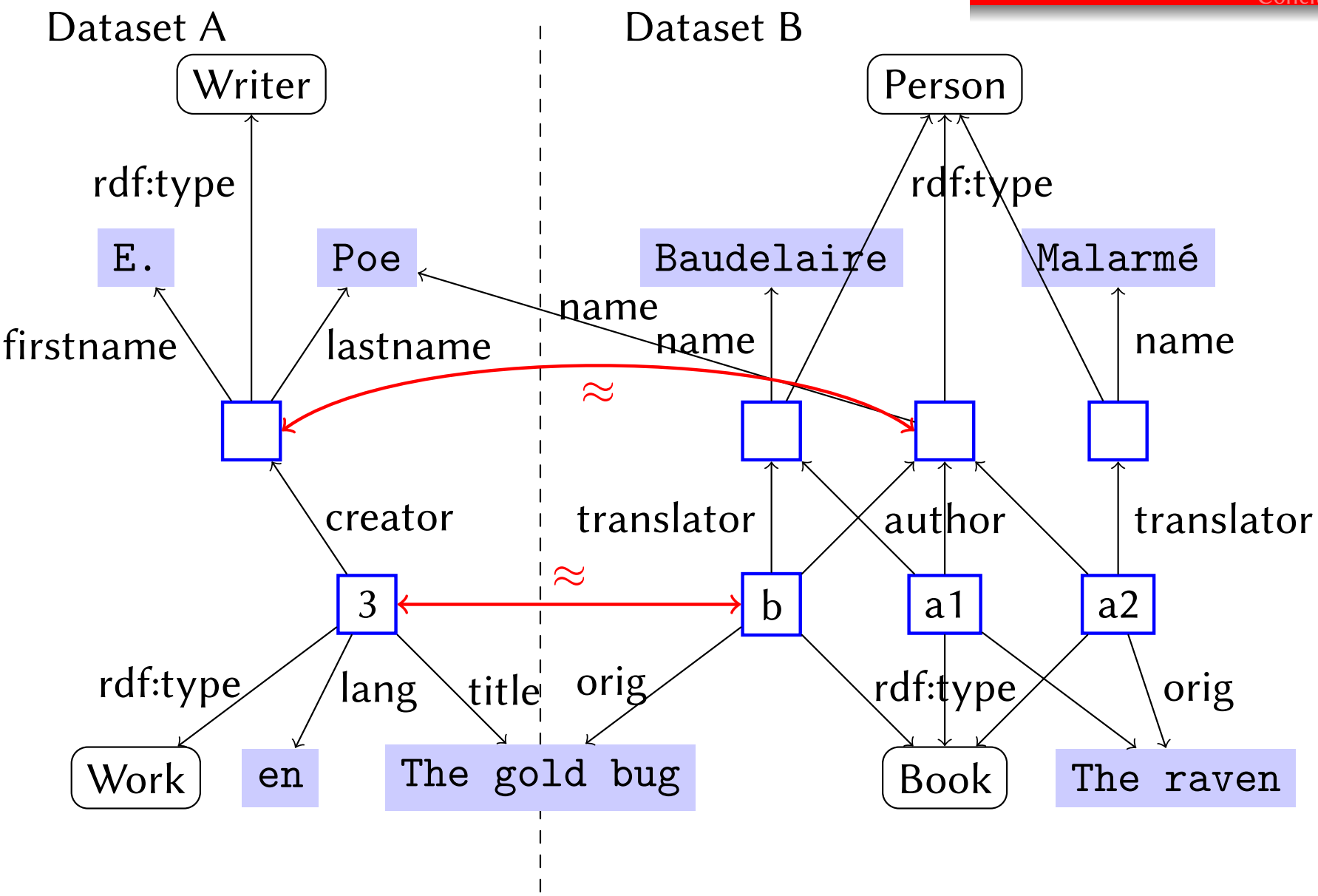
The problem: RDF data interlinking



The problem: RDF data interlinking



The problem: RDF data interlinking



Approaches to RDF data interlinking

- Numerical specifications (Link Specifications)
  - Express or learn a similarity from RDF data
  - Generate links using frameworks such as SILK or LIMES
- NLP/IR based approaches
  - Change representation: from RDF space to VSM, or embedding spaces
  - Compute or learn a similarity on this new space
- Logical link specifications
  - Key-based: combine keys and alignments for deducing links
  - Link keys: cross dataset, generalization of keys without requiring alignment between properties or concepts

Approaches to RDF data interlinking

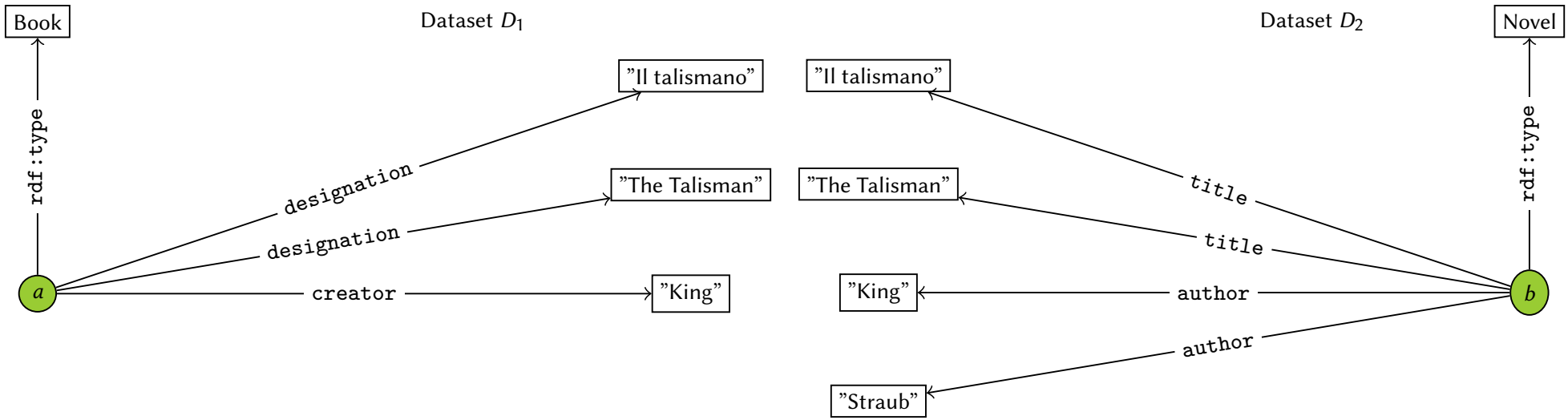
- Numerical specifications (Link Specifications)
  - Express or learn a similarity from RDF data
  - Generate links using frameworks such as SILK or LIMES
- NLP/IR based approaches
  - Change representation: from RDF space to VSM, or embedding spaces
  - Compute or learn a similarity on this new space
- Logical link specifications
  - Key-based: combine keys and alignments for deducing links
  - Link keys: cross dataset, generalization of keys without requiring alignment between properties or concepts

Outline

What is a link key?

two sets of property pairs and a pair of classes like

$$(\underbrace{\{(\texttt{designation},\texttt{title})\}}_{\forall}, \underbrace{\{(\texttt{designation},\texttt{title}),(\texttt{creator},\texttt{author})\}}_{\exists}, \underbrace{(\texttt{Book},\texttt{Novel})}_{\text{Classes}})$$



On this example, the link key will generate the link $(a, \texttt{owl:sameAs}, b)$

A more complex example

They may be several expressions having the form of link keys

D (Employés)					D' (Staff)				
id	prenom	datenaiss	poste	bât.	firstname	birthdate	position	building	id
i_2	Paul	1967	Dir.	B2	Paul		Dir.	B2	z_2
i_3	Mary	1963	Dir.	B1	Mary		Dir.	B1	z_3
i_4	John	1963	Pr.	B1	John		Pr.	B1	z_4
i_6	Bill	1980	Pr.	B1	William	1980	Pr.		z_6
i_7	Ana	1947	Dir.	B2	Ana	1947	Dir.		z_7
i_8	John	1967	Pr.	B2	John	1967	Pr.		z_8

Example of link key expressions:

- $k = \langle\{\}, \{\langle \text{datenaiss}, \text{birthdate}\rangle\}, \langle \text{Employe}, \text{Staff}\rangle\rangle$
- $h = \langle\{\langle \text{datenaiss}, \text{birthdate}\rangle\}, \{\langle \text{poste}, \text{position}\rangle\}\langle \text{Employe}, \text{Staff}\rangle\rangle$
- $l = \langle\{\langle \text{datenaiss}, \text{birthdate}\rangle, \langle \text{poste}, \text{position}\rangle\}, \{\langle \text{poste}, \text{position}\rangle\}, \langle \text{Employe}, \text{Staff}\rangle\rangle$

And generated links (if used as link keys):

- $L_k^{D,D'} = \{\langle i_7, z_7\rangle, \langle i_8, z_8\rangle, \langle i_6, z_6\rangle, \langle i_2, z_8\rangle\}$
- $L_l^{D,D'} = L_h^{D,D'} = \{\langle i_7, z_7\rangle, \langle i_8, z_8\rangle, \langle i_6, z_6\rangle\}$

A more complex example

They may be several expressions having the form of link keys

D (Employés)

id	prenom	datenaiss	poste	bât.
i_2	Paul	1967	Dir.	B2
i_3	Mary	1963	Dir.	B1
i_4	John	1963	Pr.	B1
i_6	Bill	1980	Pr.	B1
i_7	Ana	1947	Dir.	B2
i_8	John	1967	Pr.	B2

D' (Staff)

firstname	birthdate	position	building	id
Paul		Dir.	B2	z_2
Mary		Dir.	B1	z_3
John		Pr.	B1	z_4
William	1980	Pr.		z_6
Ana	1947	Dir.		z_7
John	1967	Pr.		z_8

Example of link key expressions:

- $k = \langle\{\}, \{\langle \text{datenaiss}, \text{birthdate}\rangle\}, \langle \text{Employe}, \text{Staff}\rangle\rangle$
- $h = \langle\{\langle \text{datenaiss}, \text{birthdate}\rangle\}, \{\langle \text{poste}, \text{position}\rangle\}\langle \text{Employe}, \text{Staff}\rangle\rangle$
- $l = \langle\{\langle \text{datenaiss}, \text{birthdate}\rangle, \langle \text{poste}, \text{position}\rangle\}, \{\langle \text{poste}, \text{position}\rangle\}, \langle \text{Employe}, \text{Staff}\rangle\rangle$

And generated links (if used as link keys):

- $L_k^{D,D'} = \{\langle i_7, z_7\rangle, \langle i_8, z_8\rangle, \langle i_6, z_6\rangle, \langle i_2, z_8\rangle\}$
- $L_l^{D,D'} = L_h^{D,D'} = \{\langle i_7, z_7\rangle, \langle i_8, z_8\rangle, \langle i_6, z_6\rangle\}$

A more complex example

They may be several expressions having the form of link keys

D (Employés)						D' (Staff)				
id	prenom	datenaiss	poste	bât.		firstname	birthdate	position	building	id
i_2	Paul	1967	Dir.	B2		Paul		Dir.	B2	z_2
i_3	Mary	1963	Dir.	B1		Mary		Dir.	B1	z_3
i_4	John	1963	Pr.	B1		John		Pr.	B1	z_4
i_6	Bill	1980	Pr.	B1	⟷	William	1980	Pr.		z_6
i_7	Ana	1947	Dir.	B2	⟷	Ana	1947	Dir.		z_7
i_8	John	1967	Pr.	B2	⟷	John	1967	Pr.		z_8

Example of link key expressions:

- $k = \langle\{\}, \{\langle \text{datenaiss}, \text{birthdate}\rangle\}, \langle \text{Employe}, \text{Staff}\rangle\rangle$
- $h = \langle\{\langle \text{datenaiss}, \text{birthdate}\rangle\}, \{\langle \text{poste}, \text{position}\rangle\}\langle \text{Employe}, \text{Staff}\rangle\rangle$
- $l = \langle\{\langle \text{datenaiss}, \text{birthdate}\rangle, \langle \text{poste}, \text{position}\rangle\}, \{\langle \text{poste}, \text{position}\rangle\}, \langle \text{Employe}, \text{Staff}\rangle\rangle$

And generated links (if used as link keys):

- $L_k^{D,D'} = \{\langle i_7, z_7\rangle, \langle i_8, z_8\rangle, \langle i_6, z_6\rangle, \langle i_2, z_8\rangle\}$
- $L_l^{D,D'} = L_h^{D,D'} = \{\langle i_7, z_7\rangle, \langle i_8, z_8\rangle, \langle i_6, z_6\rangle\}$

Outline

Link key induction

Problem: How to induce link keys from data ?

The number of set of pairs of properties is exponential

Our approach:

- restrict on link key expressions that would generate at least one link between the datasets
- consider only closed expressions : those which are maximal for a set of links

we call such expressions “link key candidates” (LKC) and we extract them using Formal Concept Analysis

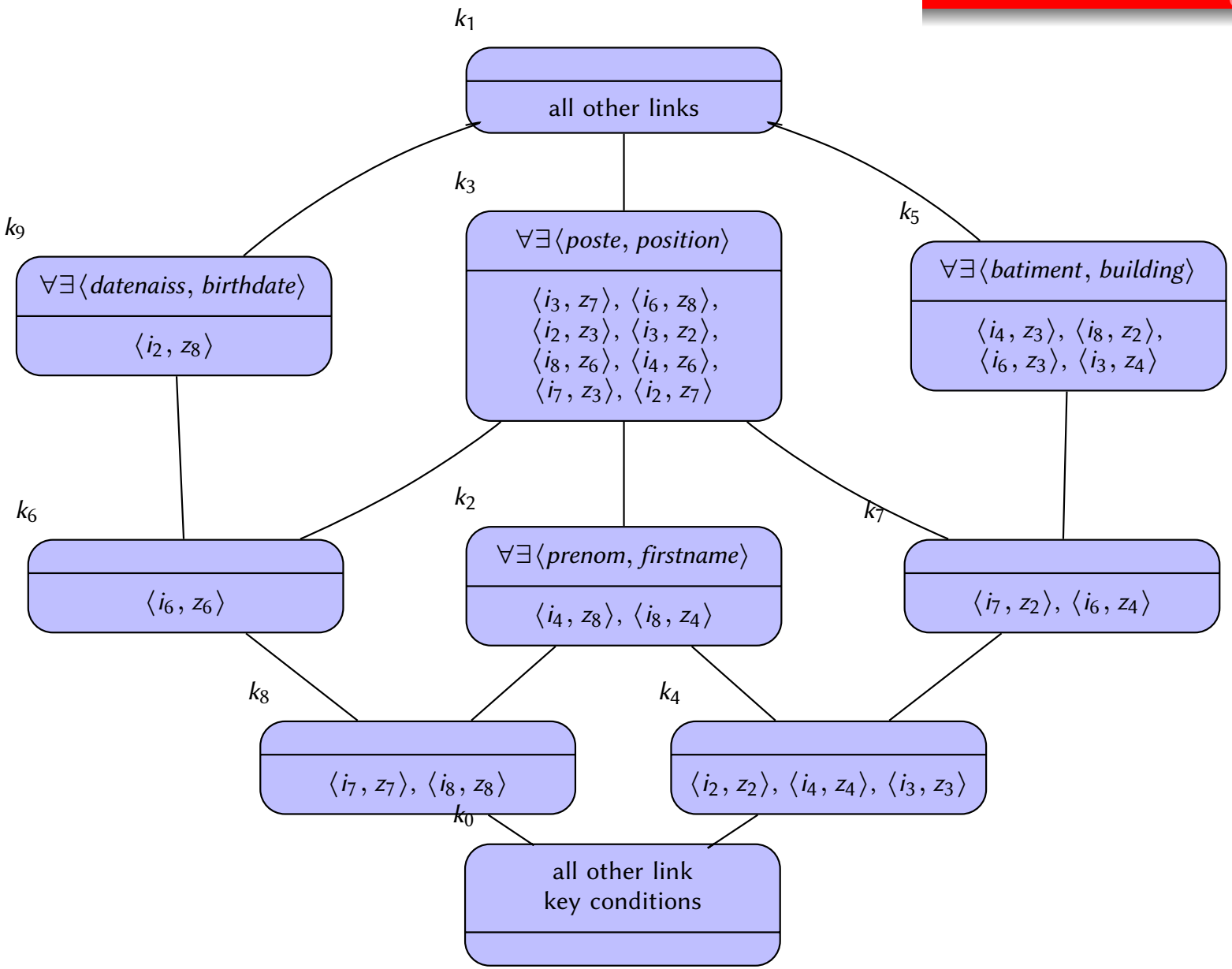
Formal context for candidate link key extraction

The *formal context for link key candidates* $\langle G, M, I \rangle$ is:

G \ M	…	$\exists\langle p_i, p'_j\rangle$	…	…	$\forall\langle p_i, p'_j\rangle$	…
⋮	⋱	⋮	⋱	⋱	⋮	⋱
$\langle o, o'\rangle$	…	1 iff $p^D(o) \cap p'^{D'}(o') \neq \emptyset$	…	…	1 iff $p^D(o) = p'^{D'}(o')$	…
⋮	⋱	⋮	⋱	⋱	⋮	⋱

- G : the set of pairs of objects from each dataset
- M : two sets of pairs of properties from each dataset
  - $\exists$: if the objects share at least one value ($\langle o, o'\rangle I \exists\langle p_i, p'_j\rangle$)
  - $\forall$: if the object have the same values ($\langle o, o'\rangle I \forall\langle p_i, p'_j\rangle$)

Lattice of extracted link key candidates



Implementation: Linkex

FCA link key extraction is implemented in Linkex
(`https://gitlab.inria.fr/moex/linkex`)

Characteristics/functionalities:

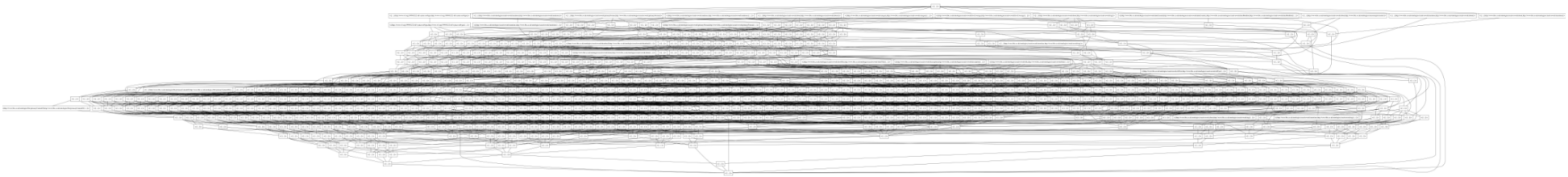
- Fully unsupervised: only two RDF datasets as input.
- Normalization of textual content: lowercase, remove diacritics, tokenization, sort
- Can compute inverse and composition of properties
- Compute the class expression of concepts (i.e. covering instances)
- Different output formats: Alignment Format, GraphViz (dot), tabular file, etc.
- Implementation of diverse quality measures (discriminability, coverage, etc)

Outline

The problem: link key selection

Many link key candidates can be extracted!

ex: OAEI Spimbench task, $> 2k$ candidates



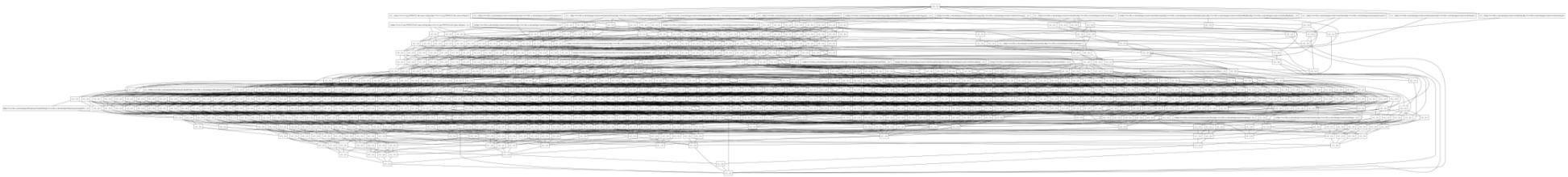
How to reduce the lattice and select the interesting/representative ones?

1. Reduce
   - Identify redundant LKC according to `owl:sameAs` semantics
   - Representative link key candidates based on clustering
2. Evaluate
   - Quality measures: discriminability and coverage
3. Combine: Disjunctions based on antichains
   - Explore the antichains of the LKC lattice

The problem: link key selection

Many link key candidates can be extracted!

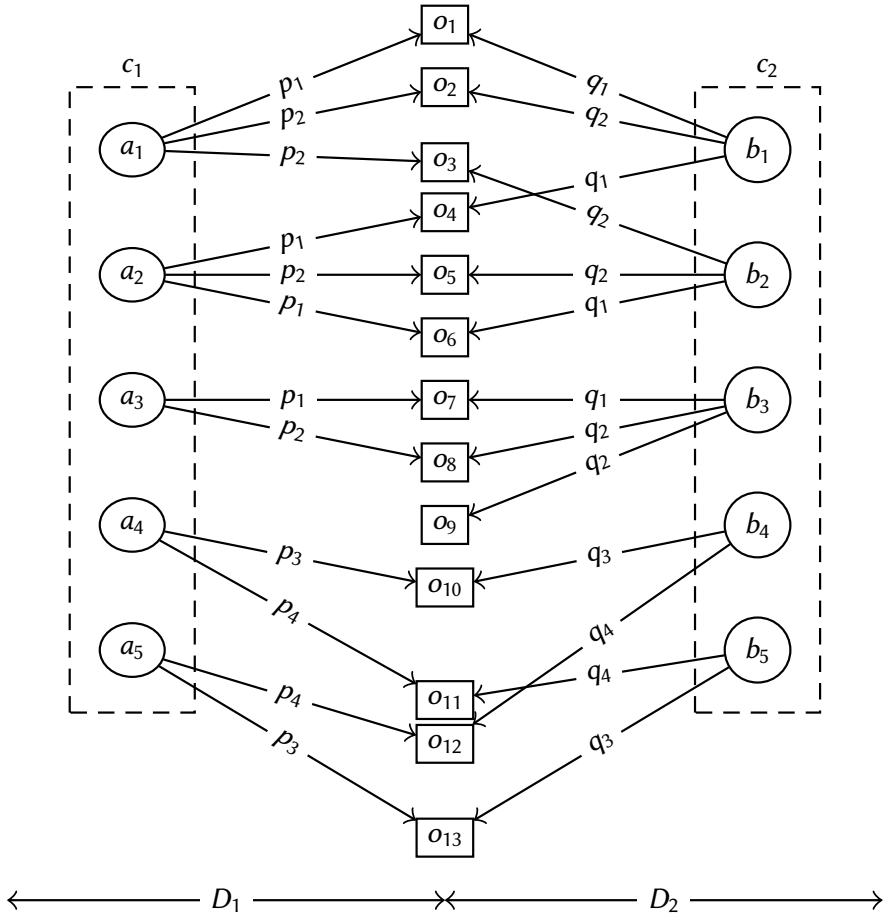
ex: OAEI Spimbench task, $> 2k$ candidates



How to reduce the lattice and select the interesting/representative ones?

1. Reduce
   - Identify redundant LKC according to `owl:sameAs` semantics
   - Representative link key candidates based on clustering
2. Evaluate
   - Quality measures: discriminability and coverage
3. Combine: Disjunctions based on antichains
   - Explore the antichains of the LKC lattice

Pattern structure for link key candidate discovery

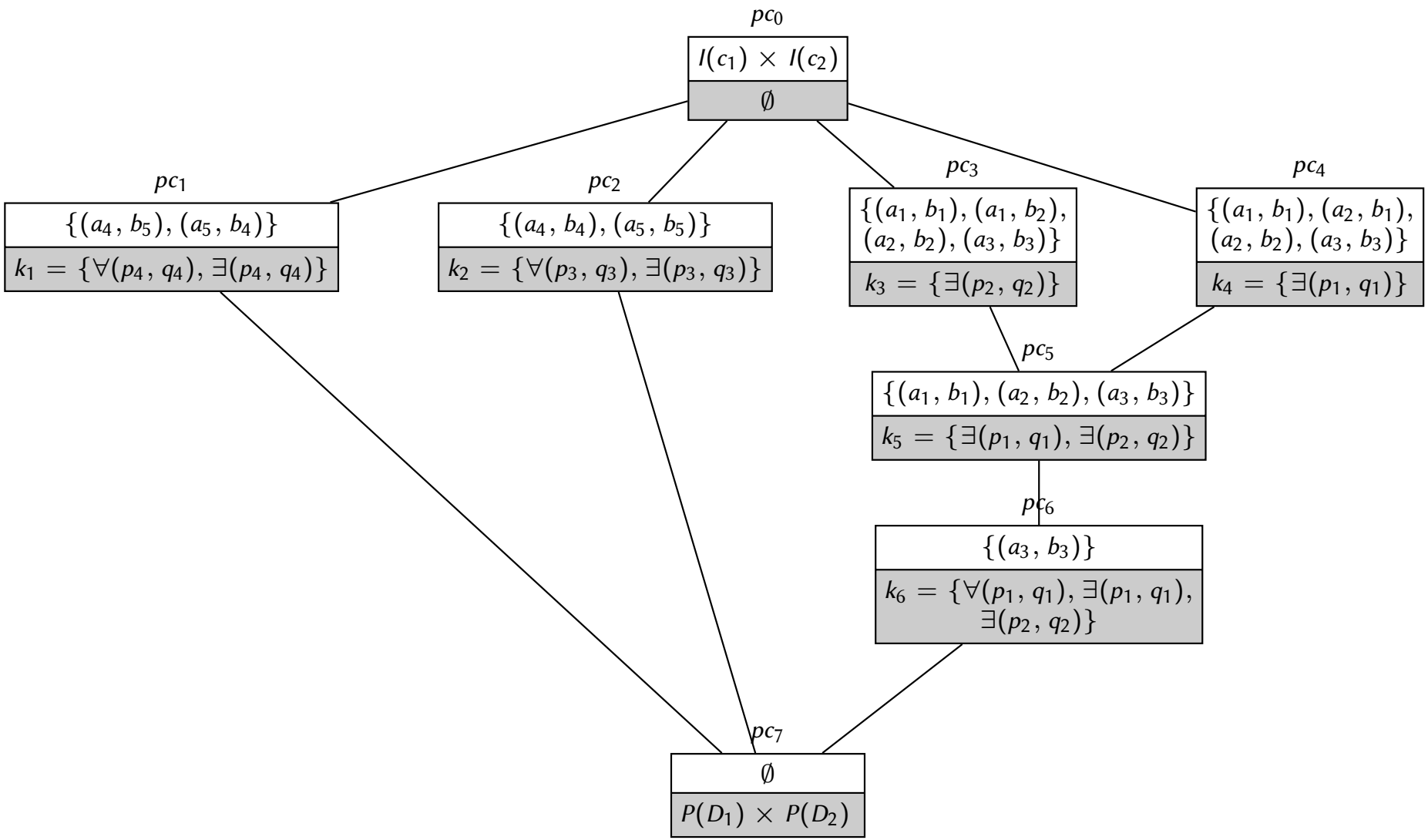


Pattern structure: $(G, (E, \sqcap), \delta)$

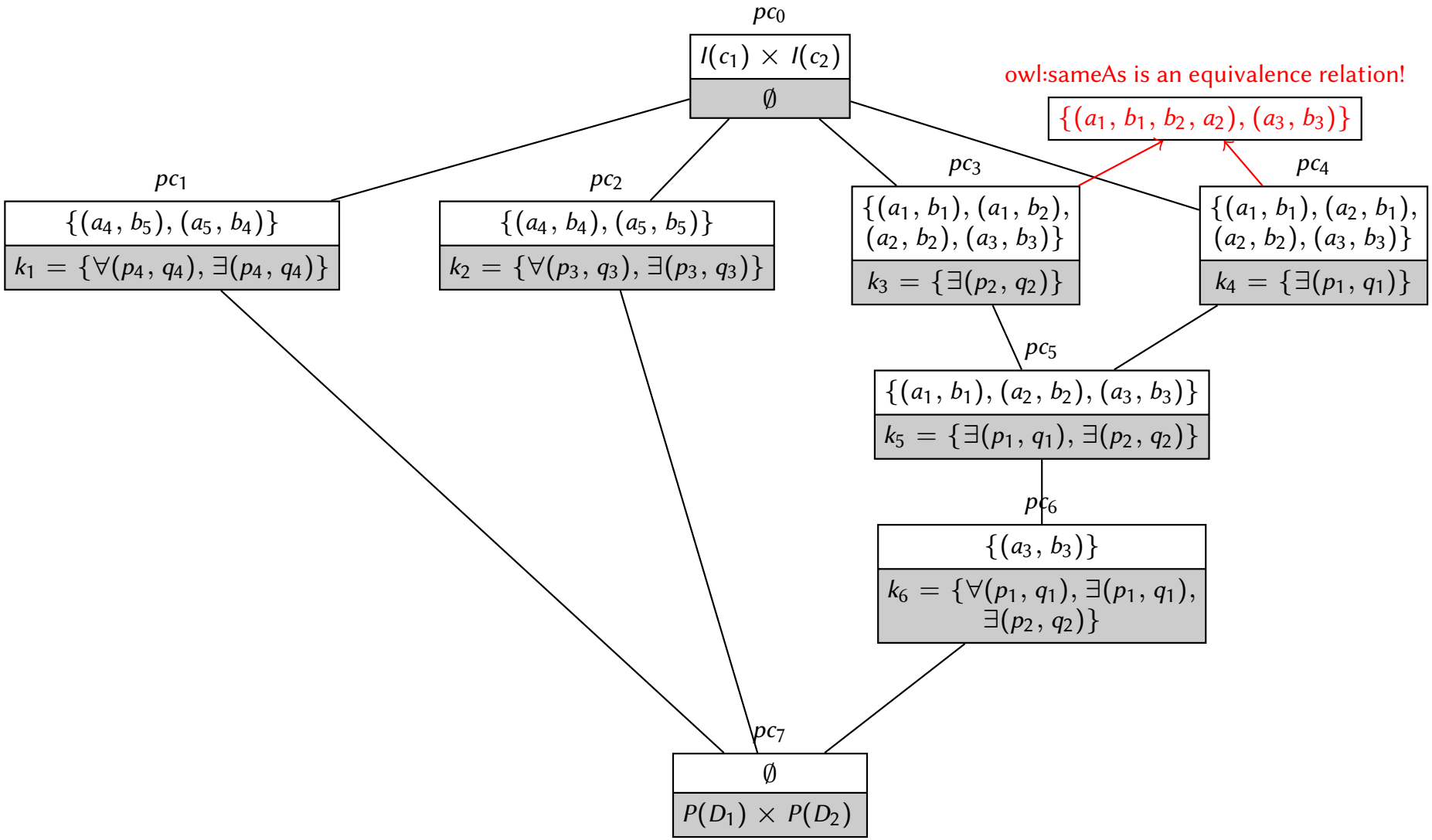
PS objects G	Descriptions (δ) E
(a_1, b_1)	$\{\exists(p_1, q_1), \exists(p_2, q_2)\}$
(a_1, b_2)	$\{\exists(p_2, q_2)\}$
(a_2, b_1)	$\{\exists(p_1, q_1)\}$
(a_2, b_2)	$\{\exists(p_1, q_1), \exists(p_2, q_2)\}$
(a_3, b_3)	$\{\forall(p_1, q_1), \exists(p_1, q_1), \exists(p_2, q_2)\}$
(a_4, b_4)	$\{\forall(p_3, q_3), \exists(p_3, q_3)\}$
(a_4, b_5)	$\{\forall(p_4, q_4), \exists(p_4, q_4)\}$
(a_5, b_4)	$\{\forall(p_4, q_4), \exists(p_4, q_4)\}$
(a_5, b_5)	$\{\forall(p_3, q_3), \exists(p_3, q_3)\}$

Nacira Abbas, Jérôme David, Amedeo Napoli: Discovery of Link Keys in RDF Data Based on Pattern Structures: Preliminary Steps. CLA 2020: 235-246

Pattern Structures lattice



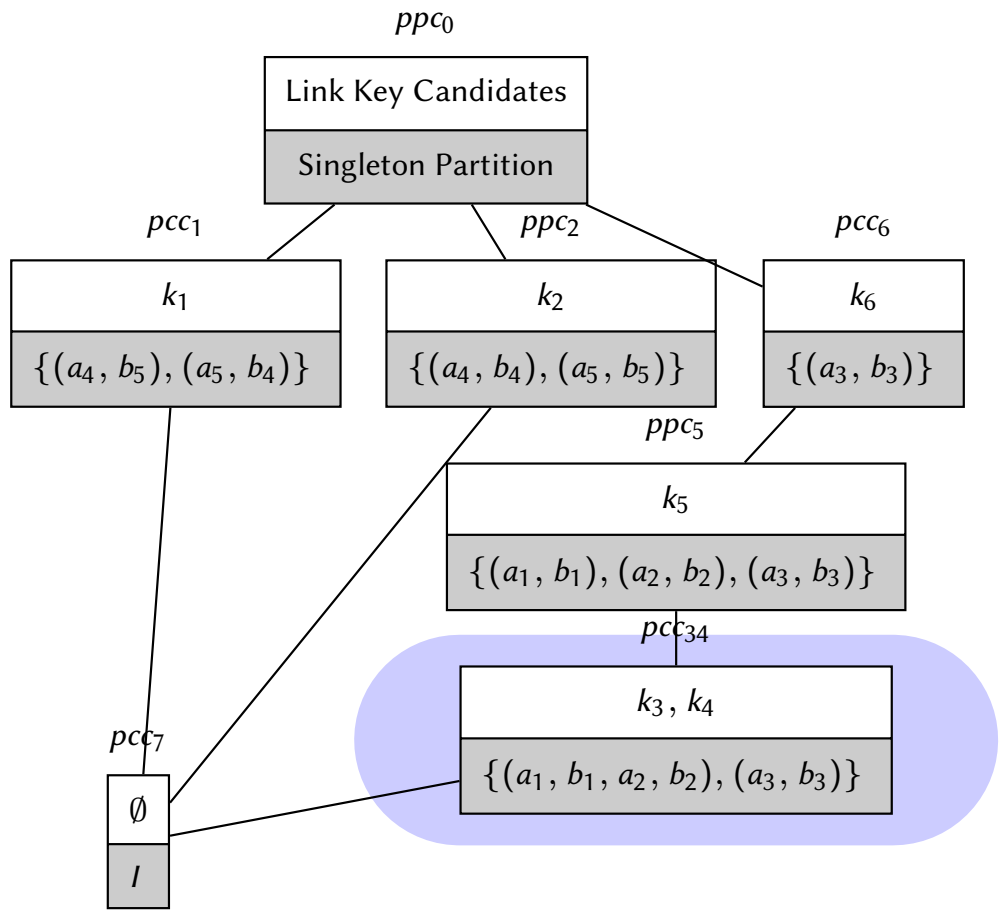
Pattern Structures lattice



Partition pattern Structures lattice

Idea: use Partition pattern structures to detect redundant LKC

Objects *LKC*	Descriptions *partitions induced by* `owl:sameAs`
k_1	$\{(a_4, b_5), (a_5, b_4)\}$
k_2	$\{(a_4, b_4), (a_5, b_5)\}$
k_3	$\{(a_1, b_1, a_2, b_2), (a_3, b_3)\}$
k_4	$\{(a_1, b_1, a_2, b_2), (a_3, b_3)\}$
k_5	$\{(a_1, b_1), (a_2, b_2), (a_3, b_3)\}$
k_6	$\{(a_3, b_3)\}$



Link key candidates having the same partition are in the same concept

Nacira Abbas, Alexandre Bazin, Jérôme David, Amedeo Napoli: A Study of the Discovery and Redundancy of Link Keys Between Two RDF Datasets Based on Partition Pattern Structures. CLA 2022: to appear

Limitations

This works but ... this is not so useful in practice :-(

Interlinking task	datasets	#triple	#subj	#prop	#LKC	#NRLKC
Actor	db:Actor yago:Actor	94 606 1 029 580	5 807 108 415	16 16	2 198	**2 177 (↓ 1%)**
Album	db:Album yago:Album	594 144 762 238	85 002 136 848	5 5	44	44
Book	db:Book yago:Book	247 372 185 032	29 846 41 849	7 7	82	82
Film	db:Film yago:Film	1 369 600 1 067 084	82 099 123 822	9 9	18 718	**17 643 (↓ 5%)**
Mountain	db:Mountain yago:Mountain	135 442 233 562	16 397 32 874	5 5	39	39
Museum	db:Museum yago:Museum	15 940 163 342	1 826 21 050	7 7	48	48
Organization	db:Organization yago:Organization	4 487 205 4 410 854	183 665 430 071	17 17	1 425	1 425
Scientist	db:Scientist yago:Scientist	128 360 671 266	18 409 92 828	10 18	862	862
University	db:University yago:University	241 838 263 624	10 352 23 334	9 9	213	213

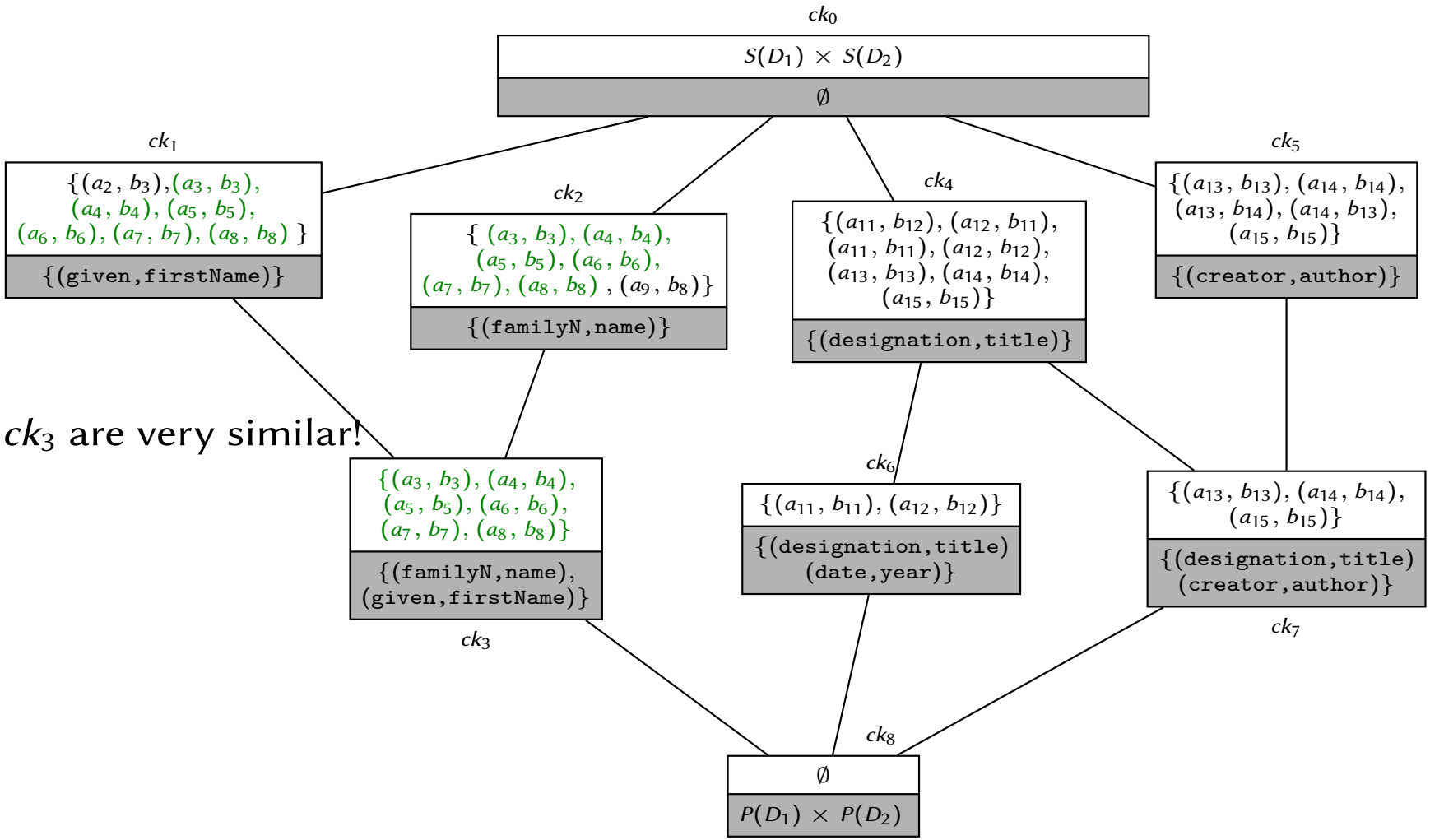
#LKC: number of link key candidates. #NRLKC: number of ”non redundant” link key candidates.

Datasets from

Danai Symeonidou, Luis Galárraga, Nathalie Pernelle, Fatiha Saïs, Fabian M. Suchanek: VICKEY: Mining Conditional Keys on Knowledge Bases. ISWC (1) 2017: 661-677

Similarity between link keys

However, a lot of link key candidates generate almost the same links...



ck_1, ck_2, ck_3 are very similar!

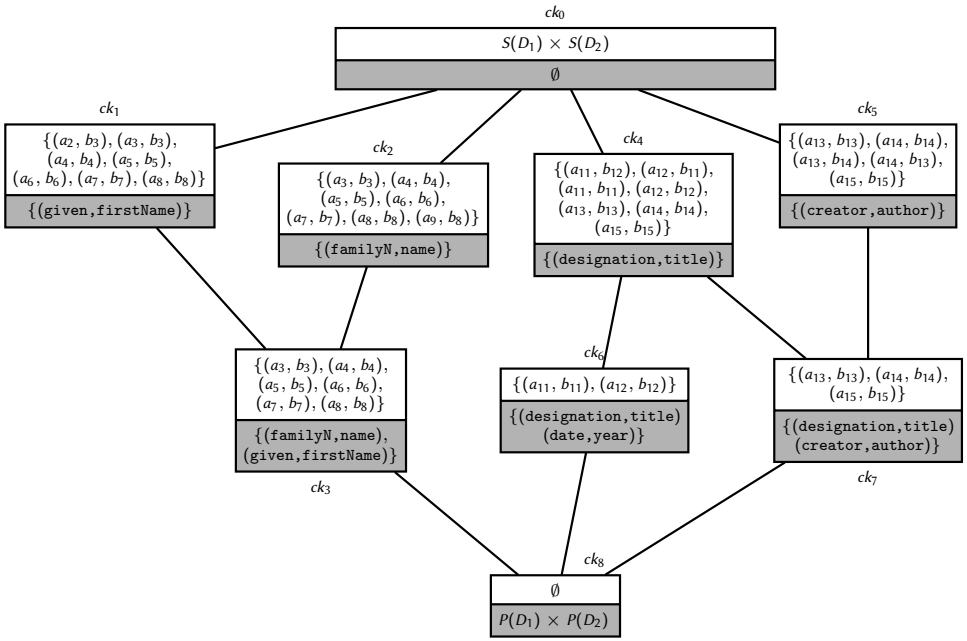
LKC clustering

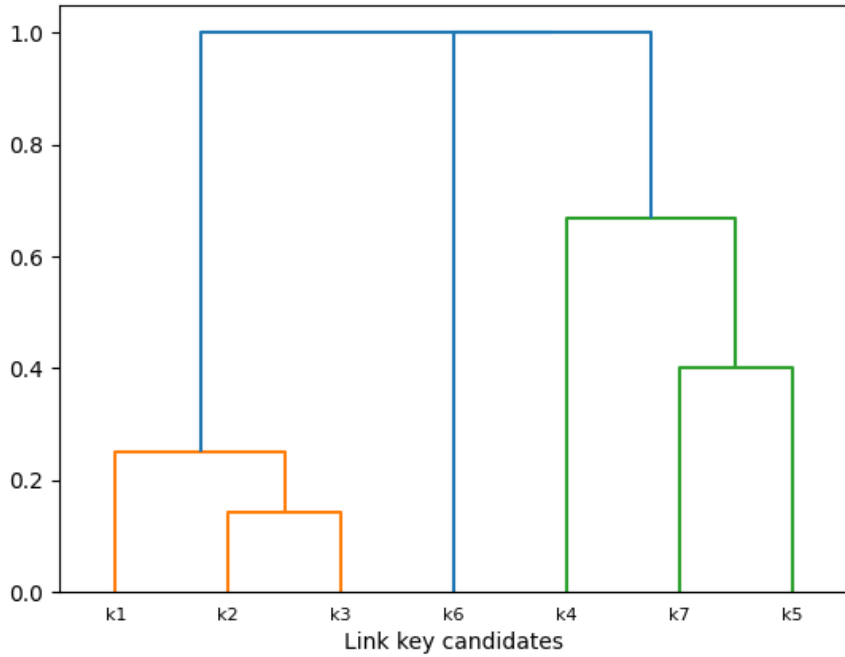
We propose to select a subset of representative candidates.

The procedure is as follows:

1. similarity is computed thanks to Jaccard index (between sets of links)
2. LKC are clustered with hierarchical agglomerative clustering
3. clusters are extracted from the resulting hierarchy
4. a representative LK is selected for each cluster
   - the LKC that minimizes the distance to the other

LKC clustering

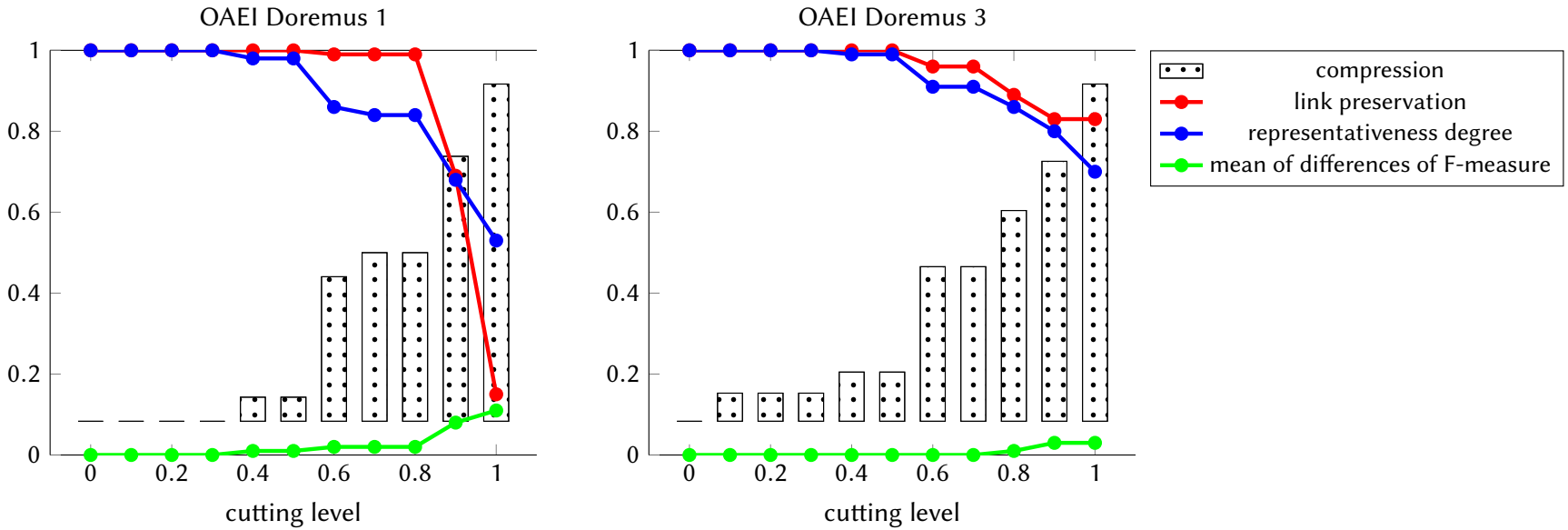




Cut at 0.5:

- 4 clusters: $\{k_1, k_2, k_3\}$, $\{k_4\}$ and $\{k_6\}$
- representatives (core): $\{k_3, k_4, k_6, k_7\}$
- This gives a compression ratio of 0.43 while preserving 77% of links.

Results on benchmarks



- Almost 50% of LKC can be removed without loss
- Core LKC are good representative (and preserve F-Measure).

Conclusion

The context:

- Link keys are symbolic and meaningful tool for interlinking data
- Fully unsupervised extraction from data with minimal input
- But... a lot of candidates can be discovered

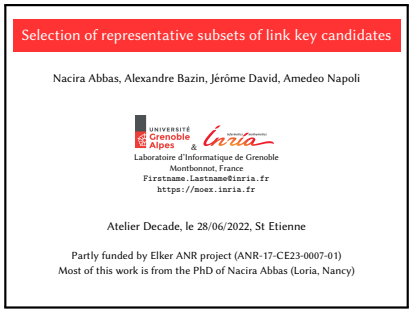
Contributions:

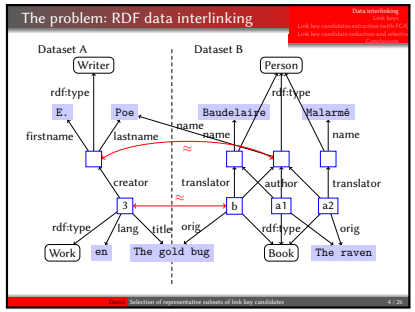
- A pattern structure for non redundant link key w.r.t. `owl:sameAs`
- A clustering based strategy for selecting representative subset of LKC

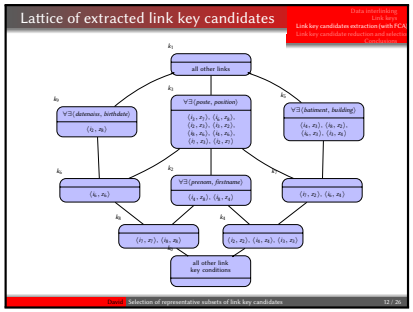
Perspectives:

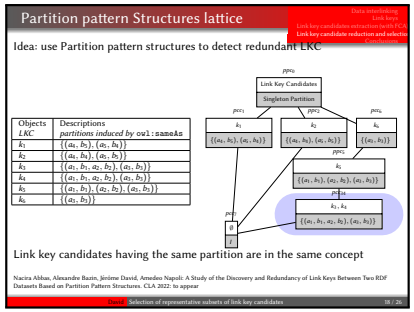
- Combine this approach with selecting disjunctions of LKC
- Generalize this clustering approach to FCA lattice reduction
  - Goal: reduce the # of concepts while preserving the order
  - adapt the similarity measure to FCA lattices
    $\frac{|L(a \wedge b)|}{|L(a \vee b)|}$ instead of $\frac{|L(a) \cap L()b)|}{|L(a) \cup L(b)|}$
  - design an optimised algorithm

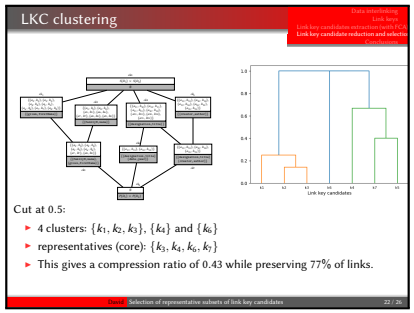
Questions?


Selection of representative subsets of link key candidates
Nacira Abbas, Alexandre Bazin, Jérôme David, Amedeo Napoli
Laboratoire d'Informatique de Grenoble
Montbonnot, France
Firstname.Lastname@inria.fr
https://moex.inria.fr
Atelier Decade, le 28/06/2022, St Etienne
Partly funded by Elker ANR project (ANR-17-CE23-0007-01)
Most of this work is from the PhD of Nacira Abbas (Loria, Nancy)


The problem: RDF data interlinking
Dataset A
Dataset B
Writer
Person
rdf:type
E.
Poe
Baudelaire
Malarmé
firstname
lastname
name
creator
translator
author
3
b
a1
a2
lang
title
orig
Work
en
The gold bug
Book
The raven


Lattice of extracted link key candidates
all other links
all other link key conditions


Partition pattern Structures lattice
Idea: use Partition pattern structures to detect redundant LKC
Objects
LKC
Descriptions
partitions induced by owl:sameAs
Link Key Candidates
Singleton Partition
Link key candidates having the same partition are in the same concept


LKC clustering
Cut at 0.5:
4 clusters: {k1, k2, k3}, {k4} and {k8}
representatives (core): {k3, k4, k6, k7}
This gives a compression ratio of 0.43 while preserving 77% of links.

https://moex.inria.fr

Nacira . Abbas
Jerome . David @ inria . fr
Amedeo . Napoli @ loria . fr