COLIBRI

Unsupervised Link Discovery Through Knowledge Base Repair

Axel-Cyrille Ngonga Ngomo Mohamed Ahmed Sherif Klaus Lyko

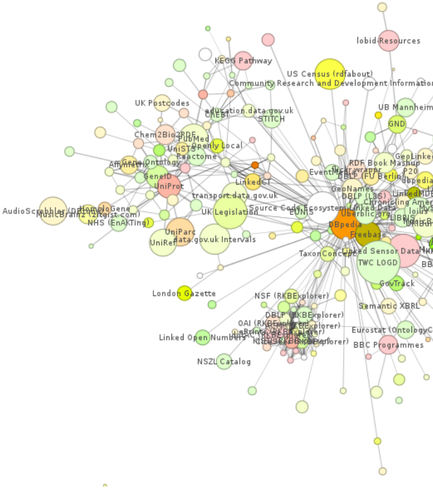
ESWC 2014, Crete, Greece

Outline

1. Motivation
2. Approach
3. Evaluation
4. Conclusion and Future Work

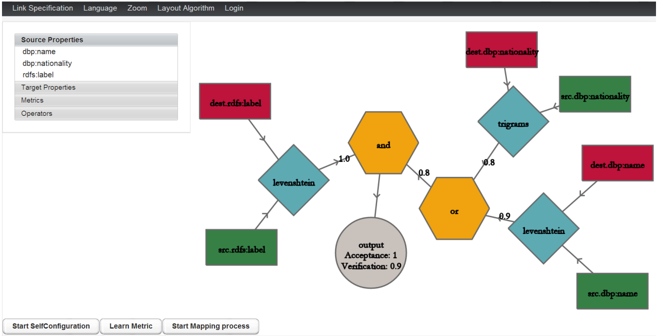
Why Link Discovery?

1. Fourth principle
2. Links are central for
   - Cross-ontology QA
   - Data Integration
   - Reasoning
   - Federated Queries
   - ...



Why is it difficult?

- **Time complexity**
  - Large number of triples
  - Quadratic runtime
- **Complexity of specifications**
  - Combination of several attributes required for high precision
  - Tedious discovery of most adequate mapping
  - Dataset-dependent similarity functions



Solution

1. Use unsupervised link discovery
   - No need for training data
   - Minimizes load on user
2. Combine results of linking tasks over $n > 2$ knowledge bases
   - Make explicit use of the topology of the Data Web
3. Repair noisy data to improve link discovery
   - Address different quality of datasets across the Data Web

Outline

1 Motivation

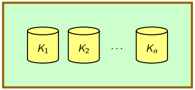
2 Approach

3 Evaluation

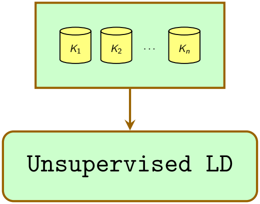
4 Conclusion and Future Work

COLIBRI *overview*

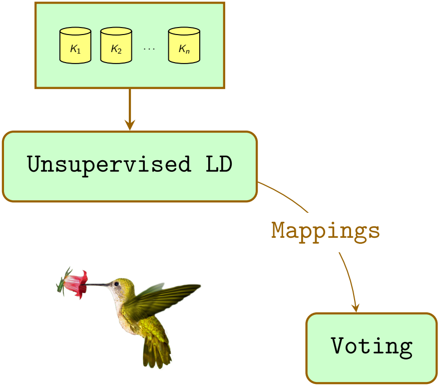
COLIBRI *overview*



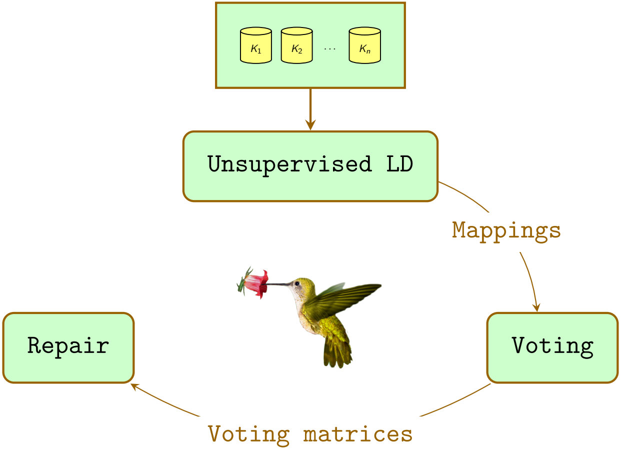
COLIBRI *overview*



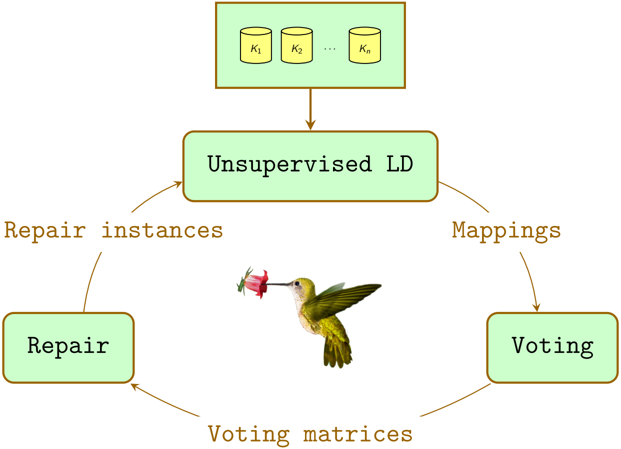
COLIBRI *overview*



COLIBRI *overview*

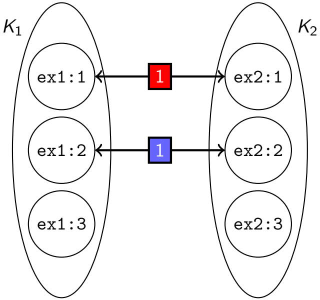


COLIBRI *overview*



Key Concepts

- Mapping matrix
  - $M_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

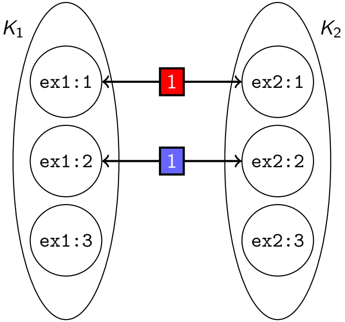


Key Concepts

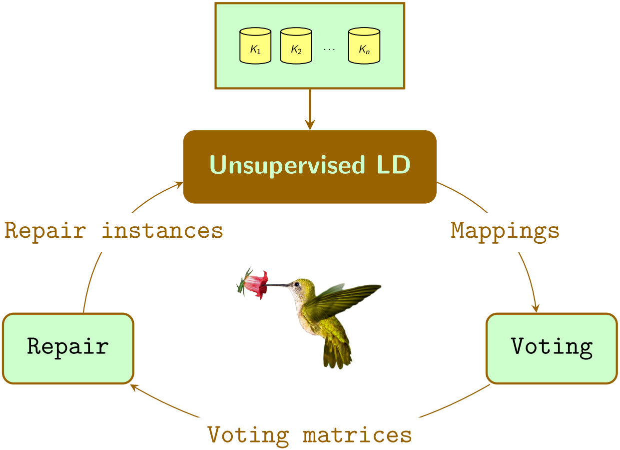
- Pseudo-F-measure as objective function
- $\mathcal{P}(M_{ij}) = \frac{|links(K_i,M_{ij})|+|links(K_j,M_{ij})|}{2|M_{ij}|}$
- $\mathcal{R}(M_{ij}) = \frac{|links(K_i,M_{ij})|+|links(K_j,M_{ij})|}{|K_i|+|K_j|}$
- $\mathcal{F}_\beta = (1+\beta^2)\frac{\mathcal{P}\mathcal{R}}{\beta^2\mathcal{P}+\mathcal{R}}$

Example:

- $\mathcal{P}(M_{12}) = 1$
- $\mathcal{R}(M_{12}) = \frac{2}{3}$
- $\mathcal{F}_1(M_{12}) = \frac{4}{5}$

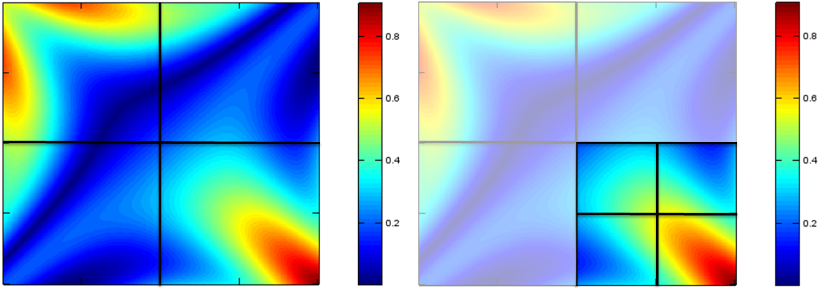


Step 1: Unsupervised Link Discovery



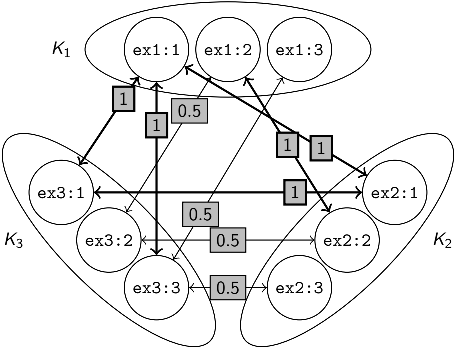
Step 1: Unsupervised Link Discovery

- Link all pairs (K_i, K_j) using any unsupervised link discovery approach
- Here, EUCLID
  - Specifications are points in a similarity space
  - Find accurate specification by using hierarchical grid search
  - Detect specification which maximizes $\mathcal{F}_\beta$

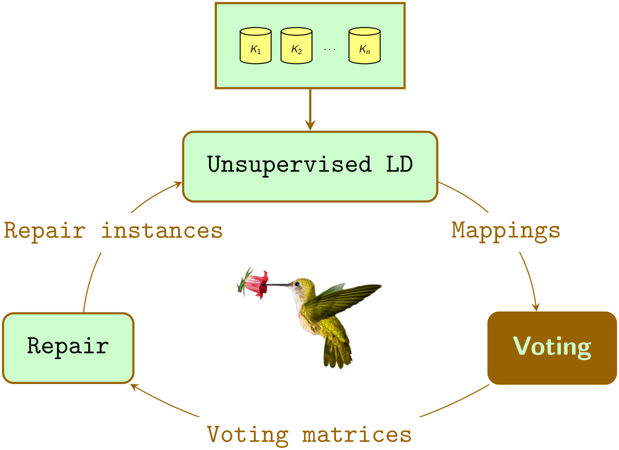


Step 1: Unsupervised Link Discovery

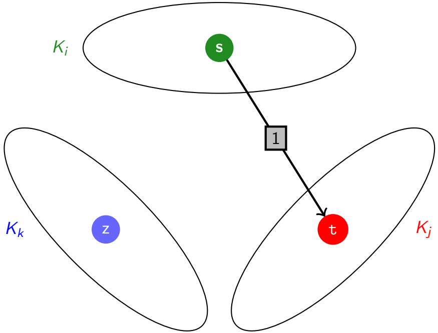
- Mapping matrices
  - $M_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$
  - $M_{13} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$
  - $M_{23} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$



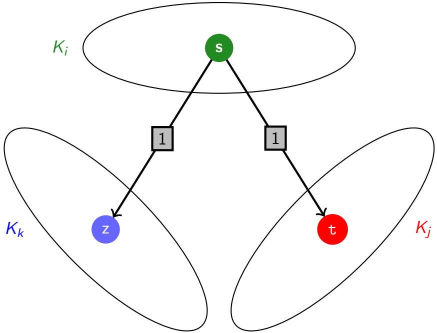
Step 2: Voting



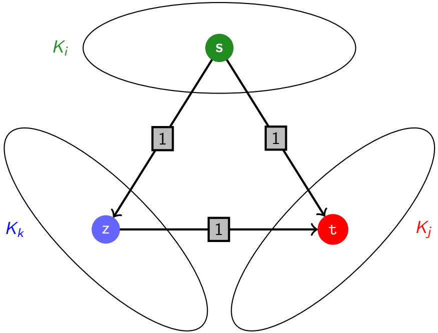
Step 2: Voting



Step 2: Voting



Step 2: Voting

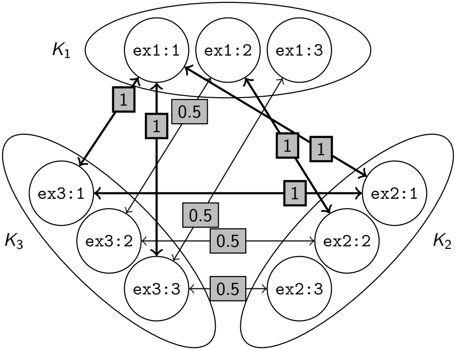


Step 2: Voting

- $V_{ij} = \frac{1}{n-1}\left(M_{ij} + \sum_{\substack{k=1\\k\neq i,j}}^{n} M_{ik}M_{kj}\right)$

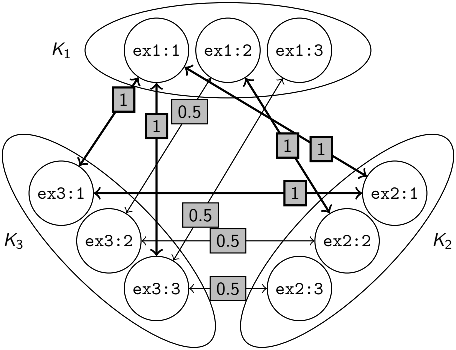
Step 2: Voting

- $V_{ij} = \frac{1}{n-1}\left(M_{ij} + \sum_{\substack{k=1 \\ k \neq i,j}}^{n} M_{ik} M_{kj}\right)$
- Mapping matrices
  - $M_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$
  - $M_{13} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$
  - $M_{23} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$



Step 2: Voting

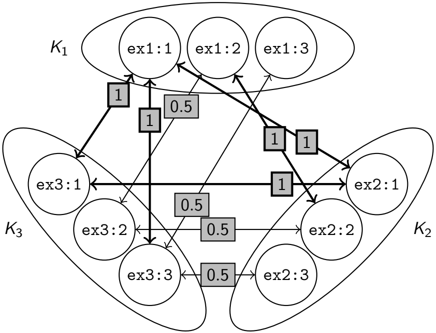
- $V_{ij} = \frac{1}{n-1}\left(M_{ij} + \sum_{\substack{k=1 \\ k\neq i,j}}^{n} M_{ik}M_{kj}\right)$
- Mapping matrices
  - $M_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$
  - $M_{13} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$
  - $M_{23} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}$



$$V_{12} = \begin{pmatrix} 1 & 0 & 0.25 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$

Step 2: Voting

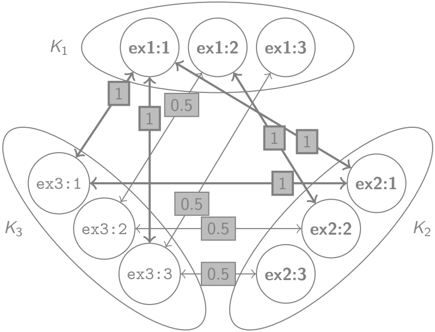
- Voting matrices
  - $V_{12} = \begin{pmatrix} 1 & 0 & 0.25 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$
  - $V_{13} = \begin{pmatrix} 1 & 0 & 0.5 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.25 \end{pmatrix}$
  - $V_{23} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.25 \end{pmatrix}$
- Post-processed matrices
  - $\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$



Step 2: Voting

- Assume links in $\tilde{V}_{ij}$ to be correct
- $\tilde{v}_{ij} = 1 \rightarrow$ All matrices agree on how to link (K_i, K_j)
  e.g., $\tilde{V}_{12}(\texttt{ex1:1}, \texttt{ex2:1})$
- For all $\tilde{v}_{ij} < 1$ assume either
  1. *Missing links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:3}, \texttt{ex2:3})$ not contained in M_{12}
  2. *Weak links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:2}, \texttt{ex2:2}) < 1$ is due to $M_{13}(\texttt{ex1:2}, \texttt{ex3:2})$ and $M_{32}(\texttt{ex3:2}, \texttt{ex2:2})$ being 0.5

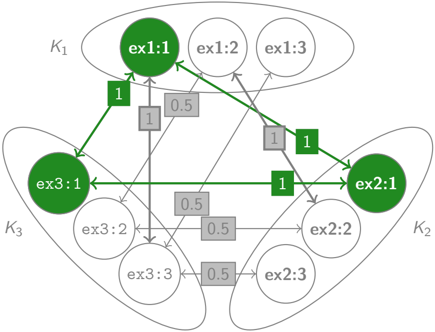
$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$



Step 2: Voting

$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$

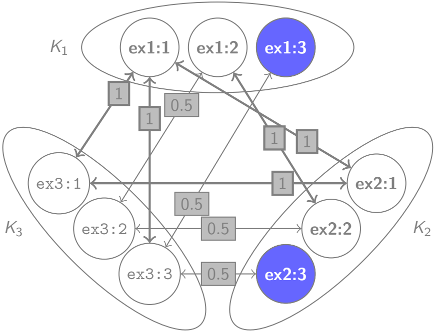
- Assume links in $\tilde{V}_{ij}$ to be correct
- $\tilde{v}_{ij} = 1 \rightarrow$ All matrices agree on how to link (K_i, K_j)
  e.g., $\tilde{V}_{12}(\texttt{ex1:1}, \texttt{ex2:1})$
- For all $\tilde{v}_{ij} < 1$ assume either
  1. *Missing links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:3}, \texttt{ex2:3})$ not contained in M_{12}
  2. *Weak links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:2}, \texttt{ex2:2}) < 1$ is due to $M_{13}(\texttt{ex1:2}, \texttt{ex3:2})$ and $M_{32}(\texttt{ex3:2}, \texttt{ex2:2})$ being 0.5



Step 2: Voting

$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$

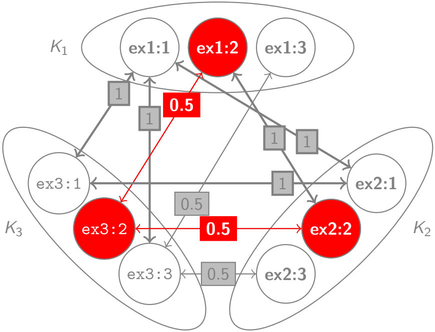
- Assume links in $\tilde{V}_{ij}$ to be correct
- $\tilde{v}_{ij} = 1 \rightarrow$ All matrices agree on how to link (K_i, K_j)
  e.g., $\tilde{V}_{12}(\texttt{ex1:1}, \texttt{ex2:1})$
- For all $\tilde{v}_{ij} < 1$ assume either
  1. *Missing links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:3}, \texttt{ex2:3})$ not contained in M_{12}
  2. *Weak links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:2}, \texttt{ex2:2}) < 1$ is due to $M_{13}(\texttt{ex1:2}, \texttt{ex3:2})$ and $M_{32}(\texttt{ex3:2}, \texttt{ex2:2})$ being 0.5



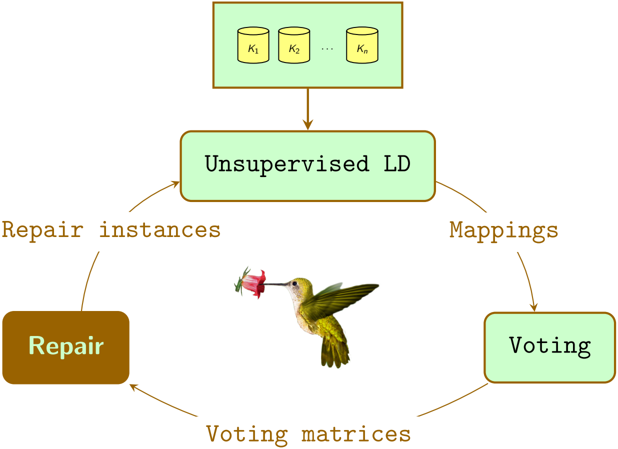
Step 2: Voting

- Assume links in $\tilde{V}_{ij}$ to be correct
- $\tilde{v}_{ij} = 1 \rightarrow$ All matrices agree on how to link (K_i, K_j)
  e.g., $\tilde{V}_{12}(\texttt{ex1:1}, \texttt{ex2:1})$
- For all $\tilde{v}_{ij} < 1$ assume either
  1. *Missing links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:3}, \texttt{ex2:3})$ not contained in M_{12}
  2. *Weak links*
     e.g., $\tilde{V}_{12}(\texttt{ex1:2}, \texttt{ex2:2}) < 1$ is due to $M_{13}(\texttt{ex1:2}, \texttt{ex3:2})$ and $M_{32}(\texttt{ex3:2}, \texttt{ex2:2})$ being 0.5

$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$



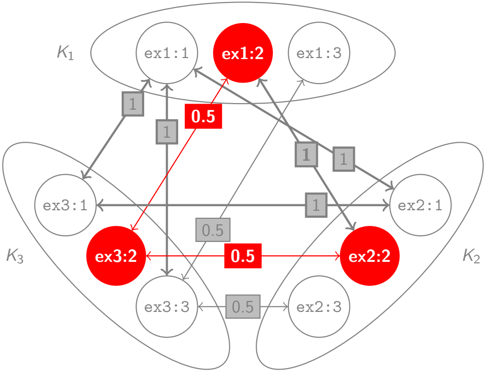
Step 3: Repair



Step 3: Repair

$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$

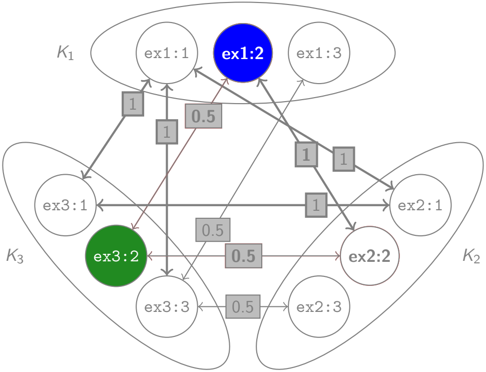
- **Goal**: Repair instance data so as to improve $\tilde{v}_{ij} < 1$
- Link to be repaired is (`ex1:2`, `ex2:2`).
- Reason for this link:
  - rs = `ex1:2` and
  - rt = `ex3:2`.
- Computing *average similarity*:
  - $\bar{\sigma}$(`ex1:2`) = 0.75 while
  - $\bar{\sigma}$(`ex3:2`) = 0.5.
- Colibri overwrite the values of `ex3:2` with those of `ex1:2`.



Step 3: Repair

$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \color{red}{0.625} & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$

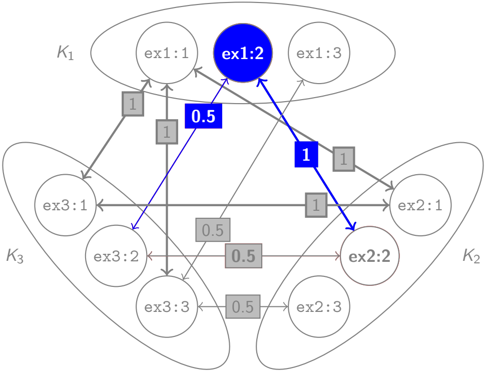
- **Goal**: Repair instance data so as to improve $\tilde{v}_{ij} < 1$
- Link to be repaired is (ex1:2, ex2:2).
- Reason for this link:
  - rs = ex1:2 and
  - rt = ex3:2.
- Computing *average similarity*:
  - $\bar{\sigma}$(ex1:2) = 0.75 while
  - $\bar{\sigma}$(ex3:2) = 0.5.
- COLIBRI overwrite the values of ex3:2 with those of ex1:2.



Step 3: Repair

$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$

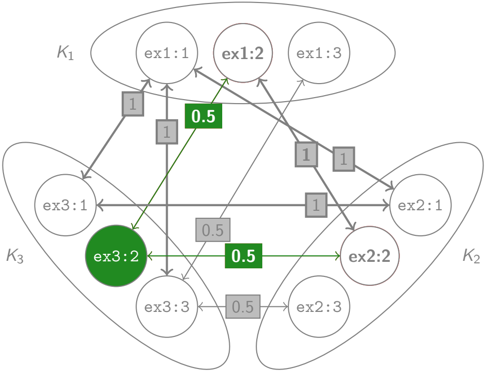
- **Goal**: Repair instance data so as to improve $\tilde{v}_{ij} < 1$
- Link to be repaired is (ex1:2, ex2:2).
- Reason for this link:
  - rs = ex1:2 and
  - rt = ex3:2.
- Computing *average similarity*:
  - $\bar{\sigma}$(ex1:2) = 0.75 while
  - $\bar{\sigma}$(ex3:2) = 0.5.
- COLIBRI overwrite the values of ex3:2 with those of ex1:2.



Step 3: Repair

$$\tilde{V}_{12} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.625 & 0 \\ 0 & 0 & 0.125 \end{pmatrix}$$

- **Goal**: Repair instance data so as to improve $\tilde{v}_{ij} < 1$
- Link to be repaired is (ex1:2, ex2:2).
- Reason for this link:
  - rs = ex1:2 and
  - rt = ex3:2.
- Computing *average similarity*:
  - $\bar{\sigma}$(ex1:2) = 0.75 while
  - $\bar{\sigma}$(ex3:2) = 0.5.
- COLIBRI overwrite the values of ex3:2 with those of ex1:2.



Outline

1 Motivation

2 Approach

3 Evaluation

4 Conclusion and Future Work

Benchmark Generation Approach

- So far, no benchmark for linking $n > 2$ knowledge bases
- Benchmark generation approach (Ferrara et al., 2011)
- Generated $m - 1$ copies of initial dataset K_1
- Alteration operators:
  - Misspellings
  - Abbreviations
  - Word permutations
- Alteration strategy:
  - Pick random resource according to alteration probability
  - Pick random operator

Experimental Setup

- Datasets:
  - Two synthetic datasets (OAEI2010)
  - Three real-world datasets (Koepcke et al., 2010)
- COLIBRI:
  - Maximal number of iterations $= 10$
  - Number of knowledge bases $= \{3, 4, 5\}$
  - Alteration probability $ap = \{10\%, 20\%, \ldots, 50\%\}$
  - Repeat each experiment 5 times

Experimental Results (synthetic dataset)

KBs	F_{EUCLID}	F_{COLIBRI}	Runtime (sec)	Repaired links
3	0.89	0.98	0.4	43
4	0.90	1.00	0.9	35
5	0.88	1.00	1.3	34

- Restaurant dataset
- Average values after 10 iterations
- Alteration probability $ap = 50\%$

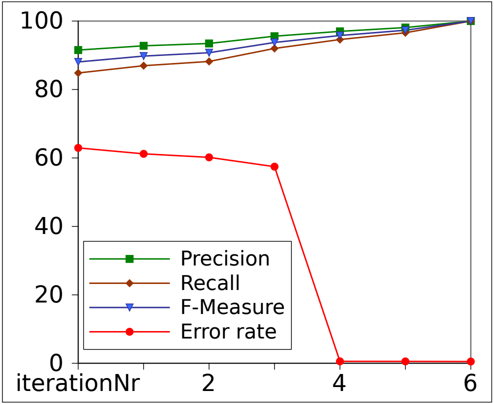
Experimental Results (real-world dataset)

KBs	F_{EUCLID}	F_{COLIBRI}	Runtime (sec)	Repaired links
3	0.86	0.98	81.8	300
4	0.85	0.99	160.4	150
5	0.84	0.88	246.8	60

- Amazon dataset
- Average values after 10 iterations
- Alteration probability $ap = 50\%$

Results on the Restaurants dataset

- Alteration probability $ap = 50\%$
- Knowledge bases = 5



Full results at:
https://github.com/AKSW/LIMES/tree/master/evaluationsResults/colibri

Outline

1. Motivation
2. Approach
3. Evaluation
4. Conclusion and Future Work

Conclusion and Future Work

- **Conclusion**
  - Presented COLIBRI
  - Improved F-measure of EUCLID up to 14%
- **Future Work**
  - Evaluation on other datasets
  - Interactive scenarios (i.e., consult user before dataset repair)
  - Combination with other unsupervised solutions (e.g., EAGLE)

Thank You!

Questions?

Mohamed Sherif
Augustusplatz 10
D-04109 Leipzig
sherif@informatik.uni-leipzig.de
http://aksw.org/MohamedSherif
http://limes.sf.net