Entity Resolution via ASP

Zhiliang Xiang[1]

School of Computer Science and Informatics
Cardiff University

July 20, 2022

Outline

- 1. Brief Recap
- 2. Implementation
- 3. Experiments and Empirical Issues
- 4. Next Step

Core Components

LACE Framework [Bienvenu et al., 2022]: Declarative framework for collective entity resolution (ER)

- ER Specifications: Hard/Soft Rules, Denial Constraints (DCs)
- Dynamic/Global Semantics: Enforce, re-evaluate, repeat
- Maximal Solutions: the best solution w.r.t. $\subseteq$

ER Specification

- **Hard Rules:** Specify merges that are sufficiently evident.

$$q(x, y) \Rightarrow EQ(x, y),$$

- **Soft Rules:** Specify merges that are less certain but possibly true.

$$q(x, y) \dashrightarrow EQ(x, y)$$

- **DCs:** Enforce consistency, reject solutions with undesired properties

$$\forall \vec{x}. \neg(\phi(\vec{x}))$$

*$q(x, y)$ finite conjunctive query may include $\approx$
*$EQ/2$ predicate indicates equality (merge), closed w.r.t. transitivity/reflectivity/symmetry
*$\phi(\vec{x})$ finite conjunction may include $\neq$

Dynamic & Global Semantics

Dynamics

- Step-by-step, induce new merges by including the merges already derived by either hard rules or soft rules
- Propagate equalities via transitivity, symmetry
- Induced database submit to DCs and hard rules

Globalilty

- Derived merges are applied globally to all their occurrences throughout the database

Maximal Solution and etc.

Maximal Solutions

- Soft rules give a set of solutions under different interpretations
- We care about the set of *Maximal solutions* w.r.t. set-inclusion ($\subseteq$) on *EQ*-facts

Sim-Safe

- *Merge Attributes*: attributes occurred rule heads of any merge rules, usually tuple IDs, e.g. paper_id
- *Sim Attributes*: evaluated by $\approx$, usually value attributes, e.g. title, name (value domain)
- A set of rules is called *sim-safe* if no attribute appear as both *merge attribute* and *sim attribute* among the rules, i.e. Merge attributes $\cap$ Sim attributes $=\emptyset$

Current Progess

- Core components are finished
- Experimented on 2 datasets in bibliographical domain:
  DBLP-ACM [Köpcke et al., 2010], Cora (under tuning) [cor, 2008]
- Built up *Meta Construct* for scaling to local merge and rule generation

ER Specification

Hard Rules: A general encoding

$$Eq(X, Y) \leftarrow R_1(X, \vec{A_1}), R_2(Y, \vec{A_2}), \vec{A_1} \approx^{t_A} \vec{A_2},$$
$$\Phi(X, \vec{T_1}, \vec{V_1}, Y, \vec{T_2}, \vec{V_2}), \vec{V_1} \approx^{t_V} \vec{V_2}, \vec{T_1} =' \vec{T_2}.$$

*R_1 and R_2 as generic indications for targeting relations
*Merge attributes are in Green, Sim attributes are in orange
*$\Phi(X, \vec{T_1}, \vec{V_1}, Y, \vec{T_2}, \vec{V_2})$ atoms conjunction of referential relations

ER Specification

Hard Rules: A general encoding

$$Eq(X,Y) \leftarrow R_1(X,\vec{A_1}), R_2(Y,\vec{A_2}), \vec{A_1}' \approx^{t_A} \vec{A_2}',$$
$$\Phi(X,\vec{T_1},\vec{V_1},Y,\vec{T_2},\vec{V_2}), \vec{V_1}' \approx^{t_V} \vec{V_2}', \vec{T_1} =' \vec{T_2}.$$

* $='$ is implemented as cardinality constraint of the form:
1#sum[Eq(X,Y)=1, (X=Y)=1]1
* Similarly $\neq'$ is implemented of the form:
1#sum[NotEq(X,Y)=1, (X≠Y)=1]1

ER Specification

Soft Rules: A general encoding

$$
\begin{aligned}
&\mathit{Active}(X, Y) \leftarrow R_1(X, \vec{A_1}), R_2(Y, \vec{A_2}), \vec{A_1}' \approx^{t_A} \vec{A_2}', \\
&\qquad\qquad\qquad \Phi(X, \vec{T_1}, \vec{V_1}, Y, \vec{T_2}, \vec{V_2}), \vec{V_1}' \approx^{t_V} \vec{V_2}', \vec{T_1} =' \vec{T_2}. \\
&\mathit{Eq}(X, Y) \vee \mathit{NotEq}(X, Y) \leftarrow \mathit{Active}(X, Y).
\end{aligned}
$$

DBLP-ACM dataset

DBLP

id	title	authors	venue	year
journals/tods/LiuDL02	A logical foundation for deductive object-oriented databases	Mengchi Liu**,** Gillian Dobbie**,** Tok Wang Ling	ACM Transactions on Database Systems (TODS)	2002
conf/vldb/VeltriCV01	Views in a Large Scale XML Repository	Dan Vodislav, Sophie Cluet, Pierangelo Veltri	**VLDB**	2001

ACM

id	title	authors	venue	year
507237	A logical foundation for deductive object-oriented DBs	Mengchi Liu**,** Tok W.Ling, Gillian Dobbie**, Yihong Zhao**	**NAN**	2002
641273	Views in a large-scale XML repository	**Vincent Aguilera**, Sophie Cluet, Tova Milo, Pierangelo Veltri, Dan Vodislav	The VLDB Journal — The International Journal on Very Large Data Bases	2002

Figure 1: DBLP/ACM table-pair

DBLP-ACM dataset

DBLP

id	title	authors	venue	year
journals/tods/LiuDL02	A logical foundation for deductive object-oriented databases	Mengchi Liu, Gillian Dobbie, Tok Wang Ling	ACM Transactions on Database Systems (TODS)	2002
conf/vldb/VeltriCV01	Views in a Large Scale XML Repository	Dan Vodislav, Sophie Cluet, Pierangelo Veltri	VLDB	2001

Publication_0

pid	id	title	vid
1	journals/tods/LiuDL02	A logical foundation	6
2	conf/vldb/VeltriCV01	Views in a Large Scale	7

$Pub(pid, id, title, vid)$

Venue_0

vid	v_name	year
6	ACM Transactions on Database Systems	2002
7	VLDB	2001

$Venue(vid, name, year)$

Authors_0

aid	a_name	position
3	Mengchi Liu	1
4	Gillian Dobbie	2
5	Tok Wang Ling	3
7	Dan Vodislav	1
8	Sophie Cluet	2
9	Pierangelo Veltri	3

$Author(aid, name)$

Wrote_0

aid	pid
3	1
4	1
5	1
7	2
8	2
9	2

$Wrote(aid, pid, pos)$

Figure 2: DBLP/ACM Split Views

ER Specification

Hard Rules: Example:

```
eq(X,Y) :- publication0(X,_,T,C), publication1(Y,_,T',C'),
                    similar(T,T',S1), S1>=90,
                    venue0(C,_,_), venue1(C',_,_),
                    #count{
                    1:  C=C';
                    2: eq(C,C')
                    }>=1,
                    #count{A1,A2: wrote0(A1,X,_),wrote1(A2, Y,_),A1 = A2;
                    A1,A2: wrote0(A1,X,_),wrote1(A2, Y,_),eq(A1,A2)}>=1.
```

Figure 3: example of hard rule ASP encoding

ER Specification

Soft Rules: Encoding example:

```
active(X,Y):- publication0(X,_,T,C), publication1(Y,_,T',C'),
                     wrote0(A,X,_), wrote1(A',Y,_),
                     authors0(X,AN),authors1(Y,AN'),
                     similar(T,T',S2), S2>=75,
                     similar(AN,AN',S3), S3>=75.

{eq(X,Y);neq(X,Y)}= 1 :- active(X,Y).
```

Figure 4: example of soft rule ASP encoding

ER Specification

DCs: Encoding example:

```
:- eq(X,Y), X!=Y, #count{1:publication0(X,_,_,_), not publication1(Y,_,_,_);
                         2:  publication0(Y,_,_,_), not publication1(X,_,_,_)}>1.
:- eq(X,Y), X!=Y, #count{1:authors0(X,_), not authors1(Y,_);
                          2: authors0(Y,_), not authors1(X,_)}>1.
:- eq(X,Y), X!=Y, #count{1:venue0(X,_,_), not venue1(Y,_,_);
                          2: venue0(Y,_,_), not venue1(X,_,_)}>1.
```

Figure 5: example of DCs ASP encoding

Globality & Dynamic Procedure, Maximal Solution

Globalilty

- equalities to (generated) tuple IDs of objects
- combining cardinality constraint, 1#`sum`[Eq(X,Y)=1, (X=Y)=1]1

Dynamics

- Iterative grounding for recursive rules in ASP [Gebser et al., 2012]
- Incremental solving by successive calls to answer set solver

Maximal Solution

- Optimisation by preferring the solutions have higher degree of set inclusion w.r.t. *Eq*-facts specified as preference statements [Bienvenu et al., 2010]
- Embed the Asprin system [Brewka et al., 2015]

Meta Construct

Generic construct to store and maintain

- schema
- object/value domain
- relation and attribute
- dependencies

Will be useful for local merges and automatic rule generation/tuning

DBLP-ACM dataset

Stats on DBLP-ACM dataset

	Source size (#entities)		Mapping size (#correspondences)		
Sources	Source 1	Source 2	Full input mapping (Cartesian product)	Reduced input mapping (blocking result)	perfect result
DBLP-ACM	2,616	2,294	6 million	494,000	2,224

Figure 6: DBLP/ACM stats

Format

- single table pair, each of which from a different schema
- attribute correspondences are known (schema-awared)

Experiment

Pre-processing

- Without external blocking techniques
- Similarities
  - Computed attribute-wise beforehand using syntactic measures [Doan et al., 2012]
  - Generated as facts, predicated $Sim(X, Y, S)$, symmetric and reflective
  - Thresholded by 50 (soft blocking)
- Split as views to take advantage of relations
- Empty values are replaced by special constant NAN

Experiment

Setup

- Ground and solved using Clingo [Gebser et al., 2012] Api
- Optimisation under `heuristic` mode approximation [Alviano et al., 2018]

Results on DBLP-ACM dataset

- Accuracy 0.93
- Running for 2 minutes, optimal model found in the 1st iteration (1.2 minutes approx.)

Lessons Learnt

- Datasets are unrealistic
  - ill-defined schema: all attributes in one table
  - over simplified: one pair of single tables of the same shape
  - Richer relation context (collectivity), schema in different shape (heterogeneity)
- Having accurate similarity measures is important
  - e.g. *VLDB* and *Very Large Database* fail to be merged due to a low syntactical similarity score.
  - lowering thresholds helps but loosen the restriction
- Domain knowledge comes from first impression could far from accurate (subjectivity of specification)
  - e.g. (first impression): authors at the same position of a publication are likely to be a merge. (in fact): the order is random
  - manual tuning would do the trick, but could be intangible as semantics are unclear

Interesting Empirical Questions

Specification optimisation/normalisation:

- tuning manually is daunting (matters a lot for quantitative approaches)
- specification to be effective and yet succinct
- templated and generated specifications
- studied in [Panahi et al., 2017]

Testing with more realistic setting

- more relation dependencies
- heterogeneous schema
- *in fact studies have a classification practical ER [Getoor and Machanavajjhala, 2012]

Syntactical similarity measures are inaccurate

Other Questions

- **Do we really need iterative solving? or does iterative grounding [Gebser et al., 2012] for recursive rules include the feature? How can we justify that?**

Next Step

- Trying out integrating Xclingo (might not work) [Cabalar et al., 2020] or other technique for explanation [Trieu et al., 2021]
- Creating a synthetic but realistic multi-relation large-scale dataset via [mus, 2021, Hildebrandt et al., 2020], and experimenting
- Figuring out generic ways to automate specification tuning

References I

(2008).
Cora citation matching dataset.
https://people.cs.umass.edu/~mccallum/data.html.
accessed 06/010/2021.

(2021).
Musicbrainz database.
https://musicbrainz.org/doc/MusicBrainz_Database.
accessed 06/010/2021.

Alviano, M., Romero, J., and Schaub, T. (2018).
Preference relations by approximation.
In *KR*, pages 2–11. AAAI Press.

References II

Bienvenu, M., Cima, G., and Gutiérrez-Basulto, V. (2022).
LACE: A logical approach to collective entity resolution.
In *PODS*, pages 379–391. ACM.

Bienvenu, M., Lang, J., and Wilson, N. (2010).
From preference logics to preference languages, and back.
In *KR*. AAAI Press.

Brewka, G., Delgrande, J. P., Romero, J., and Schaub, T. (2015).
asprin: Customizing answer set preferences without a headache.
In *AAAI*, pages 1467–1474. AAAI Press.

Cabalar, P., Fandinno, J., and Muñiz, B. (2020).
A system for explainable answer set programming.
In *ICLP Technical Communications*, volume 325 of *EPTCS*, pages 124–136.

References III

Doan, A., Halevy, A. Y., and Ives, Z. G. (2012).
Principles of Data Integration.
Morgan Kaufmann.

Gebser, M., Kaminski, R., Kaufmann, B., and Schaub, T. (2012).
Answer Set Solving in Practice.
Synthesis Lectures on Artificial Intelligence and Machine Learning.
Morgan & Claypool Publishers.

Getoor, L. and Machanavajjhala, A. (2012).
Entity resolution: Theory, practice & open challenges.
Proc. VLDB Endow., 5(12):2018–2019.

Hildebrandt, K., Panse, F., Wilcke, N., and Ritter, N. (2020).
Large-scale data pollution with apache spark.
IEEE Trans. Big Data, 6(2):396–411.

References IV

Köpcke, H., Thor, A., and Rahm, E. (2010).
Evaluation of entity resolution approaches on real-world match problems.
Proc. VLDB Endow., 3(1):484–493.

Panahi, F., Wu, W., Doan, A., and Naughton, J. F. (2017).
Towards interactive debugging of rule-based entity matching.
In *EDBT*, pages 354–365. OpenProceedings.org.

Trieu, L. L. T., Son, T. C., Pontelli, E., and Balduccini, M. (2021).
Generating explanations for answer set programming applications.
CoRR, abs/2104.08963.