KR Meets Data Quality

Meghyn Bienvenu

CNRS - LaBRI, Université de Bordeaux

Challenge: Handling messy real-world data

Bad data is the norm. Every day, businesses send packages to customers, managers decide which candidate to hire, and executives make long-term plans based on data provided by others. When that data is incomplete, poorly defined, or wrong, there are immediate consequences: angry customers, wasted time,

Only 3% of Companies' Data Meets Basic Quality Standards

"decisions are no better than the data on which they're based"

- 50% — the amount of time that knowledge workers waste in hidden data factories, hunting for data, finding and correcting errors, and searching for confirmatory sources for data they don't trust.

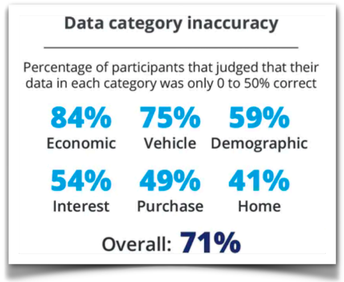
Predictably inaccurate: The prevalence and perils of bad big data

Deloitte Review, issue 21

It's pretty scary how wrong data collected about you can be—especially if people make important decisions based on this incorrect information. This becomes more frightening as more and more decisions become information-based.

DATA

Bad Data Costs the U.S. $3 Trillion Per Year



The New York Times

For Big-Data Scientists, 'Janitor Work' Is Key Hurdle to Insights

Sources: MIT Sloan Management Review, Harvard Business Review, New York Times, Deloitte Review

Data quality widely acknowledged to be a **serious and pervasive issue**

Data quality: A multi-faceted problem

Real-world data suffers from a variety of different quality issues, including:

Incorrect facts

Outdated information

Incompleteness: missing values or facts

Wrong or inconsistent format

Duplicates: multiple tuples / ids for same entity

Employee

empId	Name	Birthdate	Department	Position	Year Hired
E233	Jen R. Smith	02/11/1983	CompSci	Professor	2018
E367	Amir Aziz	14/08/1968	Math	DeptHead	1995
E546	Anna Liu	13/03/1978	Math	DeptHead	2020
E722	Jenny Smith	Nov 2 1983	CompSci	PhDStudent	2005
E767	Jie Xu	18/12/9190	Biology	Postdoc	null

Sources of issues: *faulty data entry, missing updates, integrating heterogeneous datasets…*

Vast and evolving field of research

Assessing data quality and **identifying issues**:

- syntactic and semantic constraints (declared or learned)
- statistical or ML methods to identify outliers, implausible values

(Semi)automatically cleaning data:

- conflict resolution: modify data to resolve constraint violations
- entity resolution / deduplication: identify and merge duplicates

Querying inconsistent data -> **consistent query answering**

Declarative methods
- constraints
- rules for cleaning, matching

&

Machine learning
- supervised
- unsupervised
- LLMs

Despite many advances, data quality is not a solved problem, calls for:

- holistic approaches that jointly tackle multiple data quality issues
- trustworthy and interpretable methods - don't want to introduce further errors!

Why should KR researchers care?

Nicely ties into existing KR research on handling imperfect information

- belief change, argumentation, paraconsistent / prob. / fuzzy logics, inconsistency measures…
- KR community well equipped to design formal frameworks for data quality

Increasingly sophisticated reasoning algorithms & implementations (Datalog, ASP, ontologies)

- such systems can be useful for implementing data quality tasks
- opportunity to showcase / test KR systems

Data quality needs to be addressed in data-centric KR tasks (ontology-based data access)

Natural area to combine **learning and reasoning**

Today's talk

Illustrate **synergies between data quality & KR research**

- querying inconsistent data using repair-based semantics
- logical approaches to entity resolution

Highlight **how data quality research informs KR research and vice versa**

High-level, not (too) technical, far from exhaustive survey of these lines of research

Conclude with **discussion of research challenges & opportunities**

Querying Inconsistent Data

Consistent query answering in databases

Arenas, Bertossi, Chomicki. PODS 1999

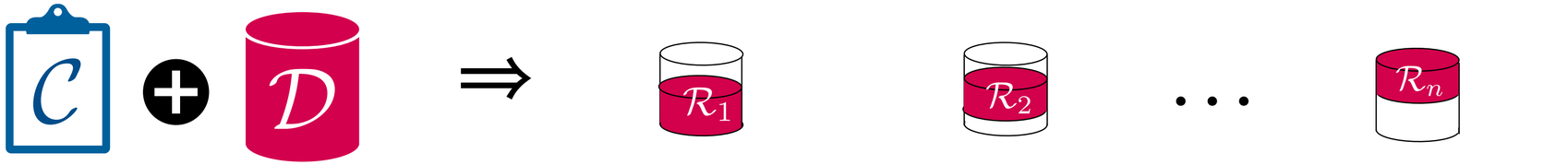
Gem of PODS, 2019

Often not enough information to precisely determine and fix data quality issues

Aim: obtain meaningful answers from inconsistent data (i.e. violates constraints)

Repair = consistent with constraints and minimally differs from original data

- maximal for set inclusion, superset, symmetric difference...



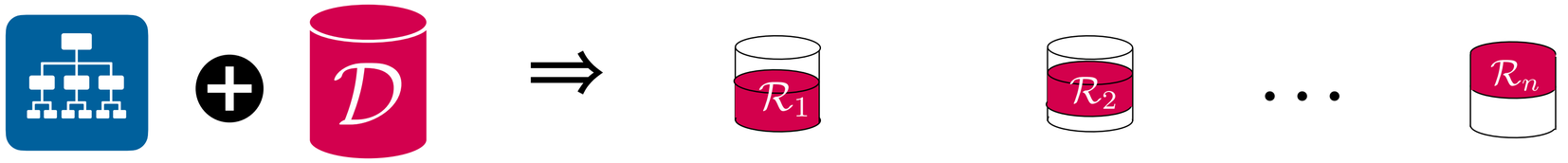
Consistent query answering (CQA): tuples that are answers in every repair

Extensively studied over past 25 years: different settings, complexity classifications

Repair-based semantics for inconsistent KBs

Surveys: Bienvenu KI 2020, Bienvenu & Bourgaux, RW 2016

Inspired study of repair-based semantics in ontology-mediated query answering



$\mathcal{O}, \mathcal{D} \models q(\vec{a})$?

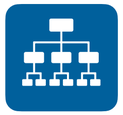
$\mathcal{O}, \mathcal{R}_1 \models q(\vec{a})$? $\quad$ $\mathcal{O}, \mathcal{R}_2 \models q(\vec{a})$? $\quad$ $\mathcal{O}, \mathcal{R}_3 \models q(\vec{a})$?

Several different repair-based semantics have been considered, including:

- **Brave semantics**: tuples that are answers w.r.t. **at least one repair** — **possible answers**
- **AR semantics (CQA)**: tuples that are answers w.r.t. **every repair** — **plausible answers**
- **IAR semantics**: tuples that are answers w.r.t. **intersection of repairs** — **surest answers**

Multiple semantics: characterize different kinds of answers or use as approximations

Illustrative example



$\text{Prof}(x) \rightarrow \text{PhDHolder}(x)$ $\text{Postdoc}(x) \rightarrow \text{PhDHolder}(x)$ $\text{Prof}(x) \wedge \text{Postdoc}(x) \rightarrow \bot$

Prof(kim) Postdoc(kim) taughtBy(cs90,kim)

Data is inconsistent with ontology, gives rise to two repairs:

$\mathcal{R}_1$ Prof(kim) taughtBy(cs90,kim)

$\mathcal{R}_2$ Postdoc(kim) taughtBy(cs90,kim)

What can we infer using the different semantics?

Brave semantics Prof(kim) Postdoc(kim) PhDHolder(kim) taughtBy(cs90,kim) ~~$\text{Prof(kim)} \wedge \text{Postdoc(kim)}$~~

AR semantics PhDHolder(kim) taughtBy(cs90,kim) ~~Prof(kim)~~ ~~Postdoc(kim)~~

IAR semantics taughtBy(cs90,kim) ~~PhDHolder(kim)~~

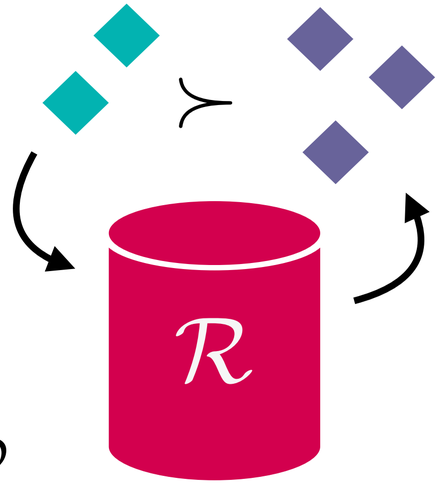
Incorporating reliability information

Staworko et al. AMAI 2012

Bienvenu & Bourgaux. KR 2020

Whenever possible, should **refine repairs by exploiting reliability information**

- priority levels, cardinality, weights, priority relation $\succ$ between facts



Three ways to use priority relation $\succ$ to select 'best' repairs:

- **Pareto-optimal repair**: cannot 'improve' $\mathcal{R}$ by adding $\alpha \in \mathcal{R} \setminus \mathcal{D}$ and removing the $\beta_1, \ldots, \beta_n$ with $\alpha \succ \beta_i$
- **globally-optimal repair**: cannot 'improve' $\mathcal{R}$ by adding $\alpha_1, \ldots, \alpha_m \in \mathcal{R} \setminus \mathcal{D}$ and removing $\beta_1, \ldots, \beta_n$ such that for every β_j, $\alpha_i \succ \beta_j$ for some α_i
- **completion-optimal repair**: greedily build repair from total order extending $\succ$

Three notions are distinct in general case (but coincide when $\succ$ given by priority levels)

Question: **which notion of prioritized repair should we adopt?**

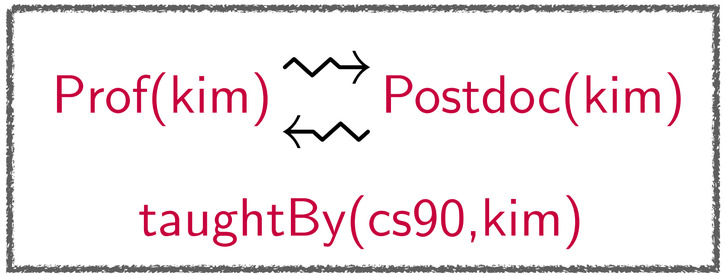
Argumentation connection

Bienvenu & Bourgaux. KR 2020

To help answer this question, establish connection to argumentation

Map prioritized DB / KB $\mathcal{K}_\succ = (\mathcal{O}, \mathcal{D}, \succ)$ to (pref-based set-based) argumentation framework $F_{\mathcal{K}_\succ}$

- use facts $\mathcal{D}$ as the arguments
- use $\succ$ as the preference
- attacks $C \setminus \{\alpha\} \rightsquigarrow \alpha$ with C a conflict
  (min incons subset of $\mathcal{D}$ wrt $\mathcal{O}$)



Theorem: Pareto-optimal repair of $\mathcal{K}_\succ$ $\Leftrightarrow$ stable extension of PSETAF $F_{\mathcal{K}_\succ}$ (often preferred extensions too)

no such correspondence for globally- and completion-optimal repairs

Provides evidence in favour of adopting Pareto-optimal repairs

Bonus: grounded semantics for prioritized databases / KBs with nice properties

Repair-based semantics via SAT solvers

Bienvenu, Bourgaux, Goasdoué. AAAI 2014, JAIR 2019

Dixit & Kolaitis. SAT 2019 & SIGMOD 2021

Bienvenu & Bourgaux. KR 2022

Querying with repair-based semantics: coNP-hard data complexity even for simple settings

AR (CQA) semantics with standard repairs

IAR / brave semantics with prioritized repairs

Independently, two SAT-based approaches were developed:

- ontology: separate SAT call for each candidate answer
- database: all answers treated together via MaxSAT calls

which approach is better?

other ways to use SAT solvers?

Motivated general exploration of SAT-based approaches:

- modular encodings built from small number of building blocks
- portfolio of algorithms employing weighted MaxSAT, MUS enum, iterative SAT
- cover AR (CQA), IAR, brave semantics, standard & prioritized (Pareto / completion) repairs

Extensive evaluation: compare encodings, algos, semantics, use DB & ontology benchmarks

Takeaways: choice of algorithm + encoding -> huge impact, dedicated IAR algos best

Some other recent and ongoing work

How to explain query (non)answers under repair-based semantics?

-> different notions of explanation, show how to compute using SAT solvers

Bienvenu, Bourgaux, Goasdoué. **Computing and Explaining Query Answers over Inconsistent DL-Lite Knowledge Bases**. JAIR 2019

How to extend preferred repairs to more expressive database constraints?
What is the relationship to active integrity constraints?

-> more evidence for Pareto-optimal repairs, also provides new insights into AIC formalism

Bienvenu & Bourgaux. **Inconsistency Handling in Prioritized Databases with Universal Constraints: Complexity Analysis and Links with Active Integrity Constraints**. KR 2023

How to adapt repair-based semantics to accommodate soft ontology axioms?

-> explore quantitative, cost-based semantics for inconsistent KBs (inspired by work on soft DB constraints)

Bienvenu, Bourgaux, Jean. **Cost-Based Semantics for Querying Inconsistent Weighted Knowledge Bases**. KR 2024

Logical Approaches to Entity Resolution

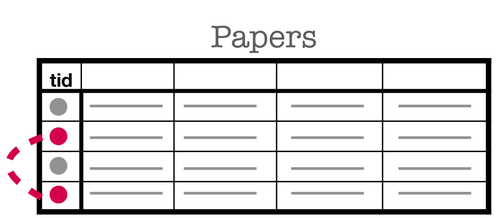
Entity resolution

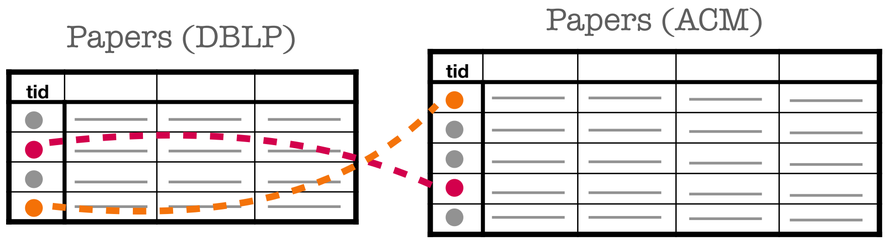
Entity resolution (ER): identify different constants denoting the same entity

(aka deduplication, duplicate detection, record linkage, reference reconciliation, merge-purge…)

Traditional ER: single entity type (e.g. papers)

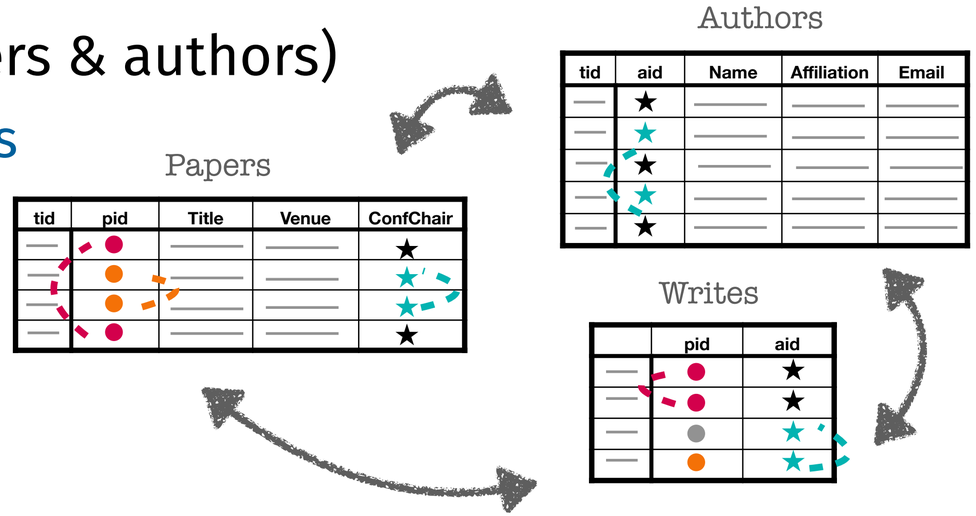
- match records within single table
- binary/pairwise: match records between two tables





Collective (aka relational) ER: multiple entities (e.g. papers & authors)

- match entity-referring constants within and across tables
- exploit relationships between entities :
  - matching authors helps to match papers, and vice-versa



Aim: explainable approaches to collective ER

Local vs global semantics for ER rules

Bienvenu et al. KR 2023

Fagin et al. KR 2023

Use **rules** to specify conditions under which pairs of **object constants** denote the **same entity**

$$\texttt{Authors}(t, \mathbf{x}, \mathbf{n}, \mathbf{i}, \mathbf{e}) \wedge \texttt{Authors}(t', \mathbf{y}, \mathbf{n}', \mathbf{i}', \mathbf{e}) \wedge \mathbf{n} \approx \mathbf{n}' \wedge \mathbf{i} \approx \mathbf{i}' \Rightarrow \texttt{EqO}(\mathbf{x}, \mathbf{y})$$

similar names, similar institution, & same email -> same author

Authors

tid	aid	Name	Institution	Email
t1	{a1,a2}	{John Lee, J. Lee}	U Toronto	jl@uoft.ca
t2	{a1,a2}	{John Lee, J. Lee}	Toronto	jl@uoft.ca
t3	a3	Jane Lee	CNRS	j.lee@cnrs.fr
t4	a4	J. Lee	LaBRI	jl@labri.fr
t5	a5	J. Lea	NII	lj@nii.jp

Adopt **global semantics** for such rules:

all occurrences of matched constants are **merged**

Writes

tid	pid	aid
t6	p1	a4
t7	p1	{a1,a2}
t8	p23	a5
t9	p15	{a1,a2}

Also use **rules** to identify **alternative representations of data values**

$$\texttt{Authors}(\mathbf{t}, \mathbf{x}, \mathbf{n}, i, e) \wedge \texttt{Authors}(\mathbf{t'}, \mathbf{x}, \mathbf{n'}, i', e') \wedge \mathbf{n} \approx \mathbf{n'} \Rightarrow \texttt{EqV}(\mathbf{t}, 2, \mathbf{t'}, 2)$$

same author id & similar names -> variants of same name

Need to use a **local semantics** for such rules: **only merge specific occurrences**

Important: **evaluate rules w.r.t. current induced database**

- redefine how to evaluate joins, similarity atoms over sets of constants

LACE: Logical Approach to Collective ER

Bienvenu, Cima, Gutierrez-Basulto. PODS 2022

Bienvenu et al. KR 2023

LACE specification consists of:

- **hard and soft rules for objects** $\mathtt{q}(\mathbf{x},\mathbf{y}) \Rightarrow \mathtt{Eq0}(\mathbf{x},\mathbf{y})$ $\mathtt{q}(\mathbf{x},\mathbf{y}) \dashrightarrow \mathtt{Eq0}(\mathbf{x},\mathbf{y})$
- **hard and soft rules for values** $\mathtt{q}(\mathbf{t},\mathbf{t}') \Rightarrow \mathtt{EqV}(\mathbf{t},i,\mathbf{t}',j)$ $\mathtt{q}(\mathbf{t},\mathbf{t}') \dashrightarrow \mathtt{EqV}(\mathbf{t},i,\mathbf{t}',j)$
- **denial constraints** $\mathtt{q} \rightarrow \bot$ $\mathtt{Authors}(t,x,n,i,e) \wedge \mathtt{Authors}(t',x,n',i',e') \wedge n \neq n' \rightarrow \bot$

same aid -> same name

ER solutions: pair of equiv relations $\langle E, V \rangle$ over object constants and value cells resp.

- obtained by (poss. empty) sequence of rule applications starting from initial DB
- final induced DB satisfies all hard rules and constraints

Interested in discovering merges -> focus on **inclusion-maximal solutions**

Space of maximal solutions: **possible & certain merges** and **query answers**

Implementing collective ER with ASP

Xiang et al. KR 2024

ASP encoding: define normal logic program whose **answer sets capture LACE solutions**

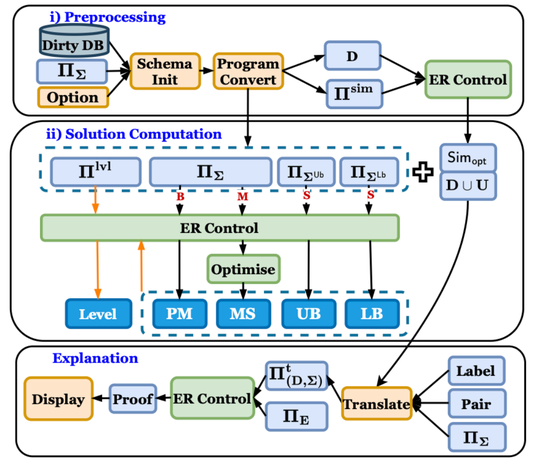
- modify rule bodies to simulate evaluation w.r.t. induced database
- maximal solutions = preferred answer sets (set-inclusion preference)

Key practical issue: **how to compute similarity facts?** *(infeasible to compare all pairs of constants!)*

- optimized similarity computation (online function calls + exploit program structure)

ASPEn system: Python implementation with calls to clingo

- generates variants of encoding, orchestrates calls to clingo
- input: database, ASP encoding, desired outputs
- outputs: different sets of merges (possible merges, upper & lower bound mergesets), fixed # of maximal solutions, explanations of possible merges



Promising experimental results, especially for complex multi-relational settings

Combining ER & consistent query answering

Bienvenu, Cima, Gutierrez-Basulto. IJCAI 2023

Merging constants can help to resolve some inconsistencies, but not all

- **also need repairing operations** (which may in turn enable further merges)

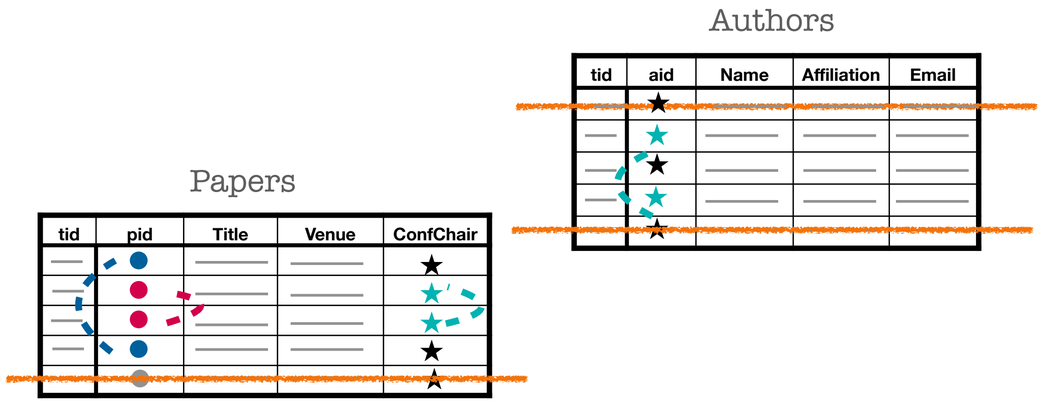
REPLACE: combines **LACE framework** with **database repairs**

- specifications as in (original) LACE framework: hard & soft rules for objects, denial constraints

Solutions take the form $\langle R, E \rangle$ where:

- R is set of database facts to remove
- E is equivalence relation over object constants

and E is a LACE solution w.r.t. database $D \setminus R$



Consider three kinds of optimal solution

min R then max E

max E then min R

Pareto: jointly min R and max E

Challenges & Opportunities

Formal frameworks for data quality

Develop new formal frameworks for data quality

- broader unified frameworks, e.g. integrate ER, repairs, and ontologies
- taking into account temporal data and knowledge
- how best to specify and integrate qualitative / quantitative preferences?
- how to define (and compute) different kinds of explanations?
- interactive approaches that exploit user feedback

Explore the computational properties of data quality frameworks

- complexity classifications: precisely delineate tractability frontier
- identify tractable settings / approximations
- devise pragmatic algorithmic approaches

Reasoning systems & integration with ML

Implement data quality tasks using reasoning systems

- naturally uses **many functionalities of ASP / SAT / Datalog systems**
  - brave & skeptical reasoning, preferences, external functions, explanation, ...
- **scalability** remains crucial issue -> explore parallel algorithms?
- develop specific optimizations, e.g. for **similarity computation** (blocking techniques)
- ER and repairs can serve as **challenging benchmarks for ASP / SAT / argumentation systems**

Combine declarative and machine learning approaches

- utilize **machine learning predicates** in place of string similarity measures
- use ML to **suggest missing values**, **value resulting from a merge**
- **learn entity resolution rules, constraints, preferences**

Data quality: Opportunities for KR research

Data quality: an important practical problem, attracting lots of industry attention

Topic has already inspired fruitful lines of research within the KR community

- repair-based semantics for querying inconsistent knowledge bases
- logical frameworks for entity resolution

KR approaches relevant and can bring new insights, even for ‘pure’ database setting

Many aspects of data quality that remain to be explored!

- expressive formal frameworks for repairing and reasoning about imperfect data
- challenging application to test and showcase KR reasoning systems
- natural domain to combine declarative and ML approaches

Camille

Quentin

Gianluca

Daniil

Questions?

Many thanks to all of my collaborators - in particular former and current students & postdocs!

Yvon

Robin

Pierre

Zhiliang (Leon)

References

Cited papers on repair-based semantics

Consistent query answering

Arenas, Bertossi, Chomicki. **Consistent Query Answers in Inconsistent Databases**. PODS 1999

Surveys on repair-based semantics in the ontology setting

Bienvenu. **A Short Survey on Inconsistency Handling in Ontology-Mediated Query Answering**. KI 2020

Bienvenu & Bourgaux. **Inconsistency-Tolerant Querying of Description Logic Knowledge Bases**. Reasoning Web 2016

Optimal repairs of prioritized databases & knowledge bases

Staworko, Chomicki, & Marcinkowski. **Prioritized repairing and consistent query answering in relational databases.** Annals of Mathematics and Artificial Intelligence (AMAI), 2012

Bienvenu & Bourgaux. **Querying and repairing inconsistent prioritized knowledge bases: Complexity analysis and links with abstract argumentation**. KR 2020

Argumentation connection

Cited papers on SAT-based approaches

Ontology setting

Bienvenu, Bourgaux, Goasdoué. **Querying Inconsistent Description Logic Knowledge Bases under Preferred Repair Semantics**. AAAI 2014

Bienvenu, Bourgaux, Goasdoué. **Computing and Explaining Query Answers over Inconsistent DL-Lite Knowledge Bases**. JAIR 2019

Database setting

Dixit & Kolaitis. **A SAT-Based System for Consistent Query Answering.** SAT 2019

Dixit & Kolaitis. **CAvSAT: Answering Aggregation Queries over Inconsistent Databases via SAT Solving.** SIGMOD 2021

Generic approach (databases & KBs)

Bienvenu & Bourgaux. **Querying Inconsistent Prioritized Data with ORBITS: Algorithms, Implementation, and Experiments**. KR 2022

Cited papers on entity resolution

Initial LACE framework

Meghyn Bienvenu, Gianluca Cima, Víctor Gutiérrez-Basulto.
LACE: A Logical Approach to Collective Entity Resolution. PODS 2022

Global & local semantics

Meghyn Bienvenu, Gianluca Cima, Víctor Gutiérrez-Basulto, Yazmín Ibáñez-García.
Combining Global and Local Merges in Logic-based Entity Resolution. KR 2023

Ronald Fagin, Phokion G. Kolaitis, Domenico Lembo, Lucian Popa, Federico Scafoglieri.
A Framework for Combining Entity Resolution and Query Answering in Knowledge Bases. KR 2023

Combining ER & repairs

Meghyn Bienvenu, Gianluca Cima, Víctor Gutiérrez-Basulto. **REPLACE: A Logical Framework for Combining Collective Entity Resolution and Repairing**. IJCAI 2023

ASP implementation

Zhiliang Xiang, Meghyn Bienvenu, Gianluca Cima, Víctor Gutiérrez-Basulto, Yazmín Ibáñez-García. **ASPEN: ASP-Based System for Collective Entity Resolution**. KR 2024