CoDEx: A Comprehensive Knowledge Graph Completion Benchmark

Tara Safavi
University of Michigan, Ann Arbor
CDL Meetup April 2021

@tararootcake
tsafavi.github.io

COMPUTER SCIENCE & ENGINEERING
UNIVERSITY OF MICHIGAN

CoDEx: A Comprehensive Knowledge Graph Completion Benchmark

Tara Safavi
University of Michigan
tsafavi@umich.edu

Danai Koutra
University of Michigan
dkoutra@umich.edu

Abstract

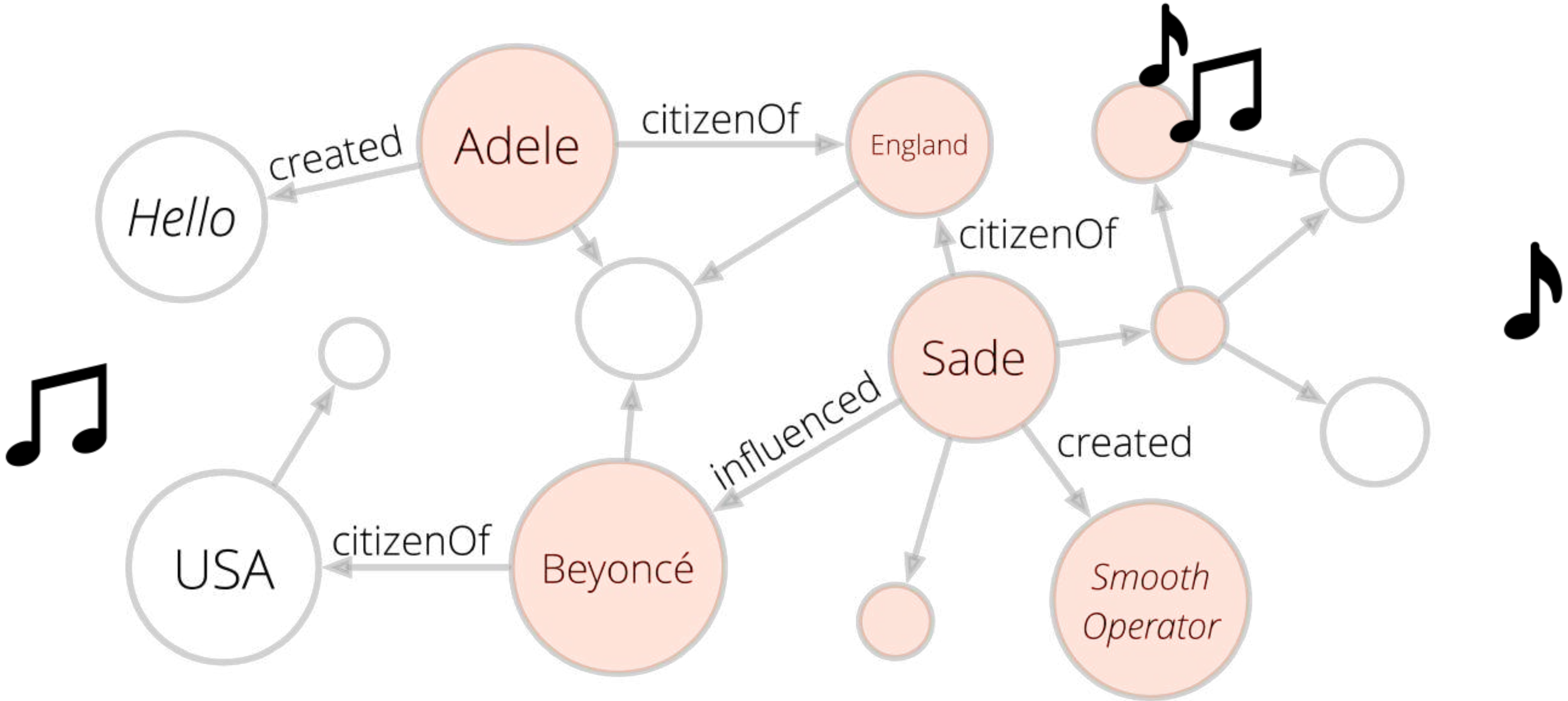
We present CoDEx, a set of knowledge graph

et al., 2008) are most commonly used for evaluation in KGC, even though Freebase had known quality issues (Tanon et al., 2016) and was eventu-

How do we measure the quality of a KG?

Knowledge graphs (KGs)

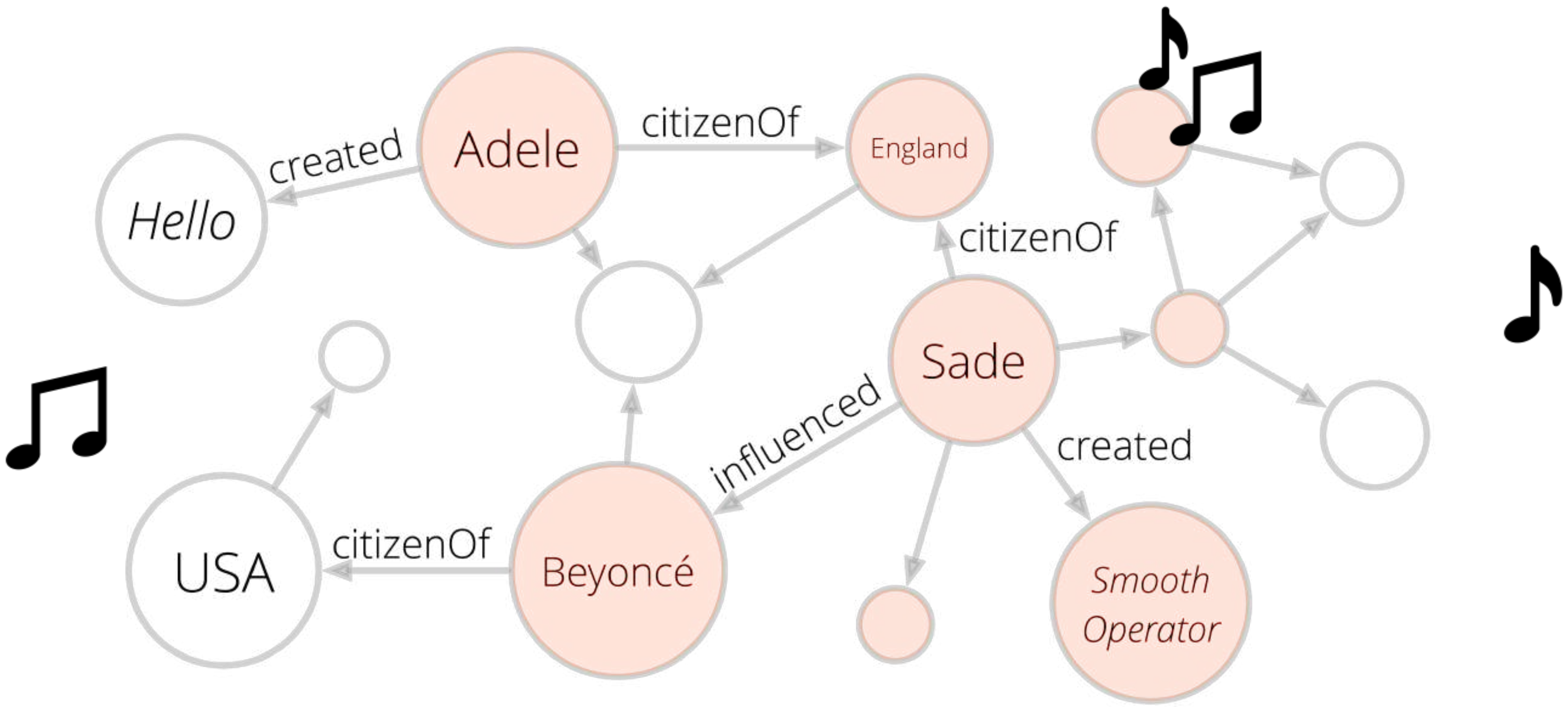
Symbolic entity-relation model of machine knowledge



[Safavi and Koutra 2021]: *more on machine knowledge representation with/without schemas

What is the precision of KGs?

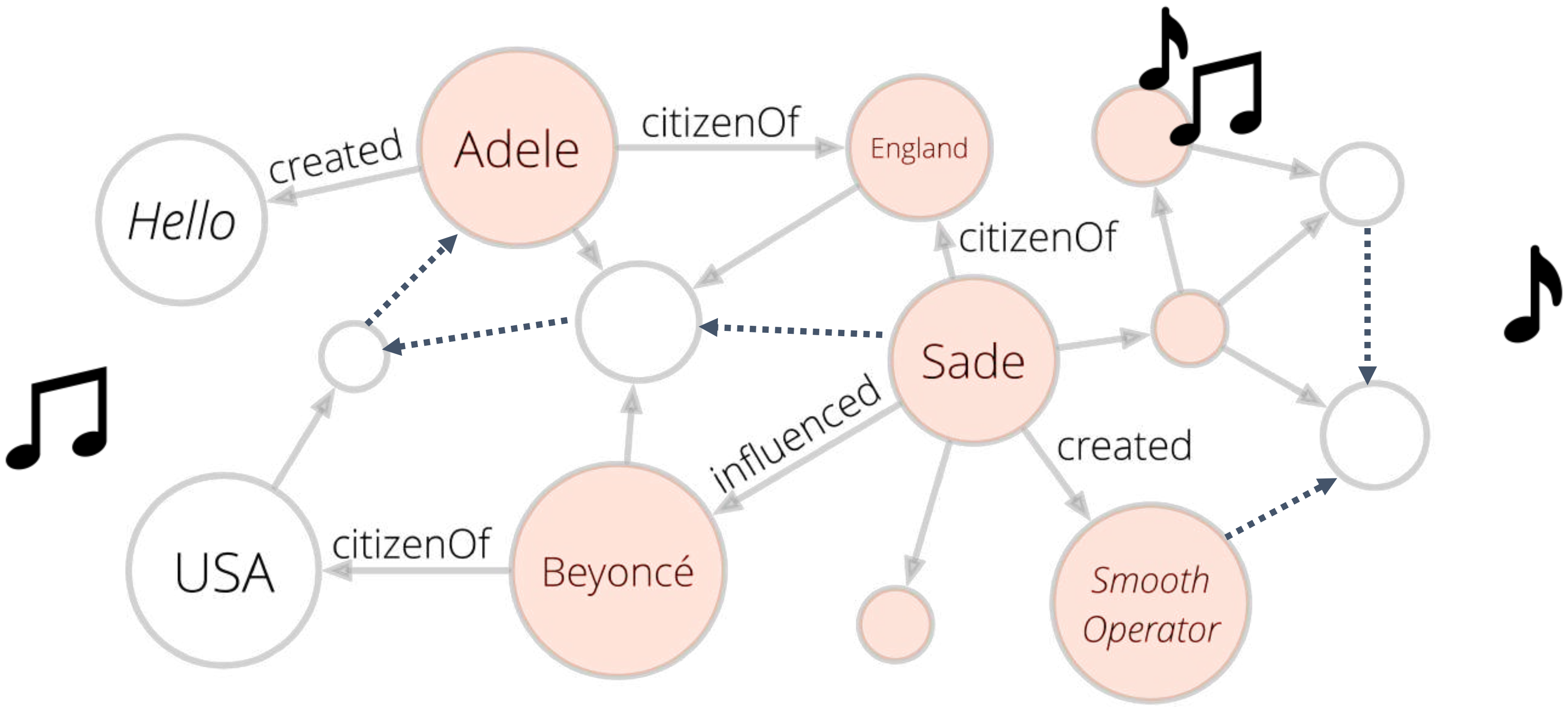
Constructed and/or verified by humans → high precision (accuracy)



[Liu and Singh 2004] [Mahdisoltani et al 2014] [Weikum et al 2020]

What is the recall of KGs?

Always growing…but never truly “complete”



[Galarraga et al 2017] [Weikum et al 2020]

How to improve recall?

How to improve recall?

Hire experts or crowd workers

How to improve recall?

Hire experts or crowd workers

Information extraction from documents

How to improve recall?

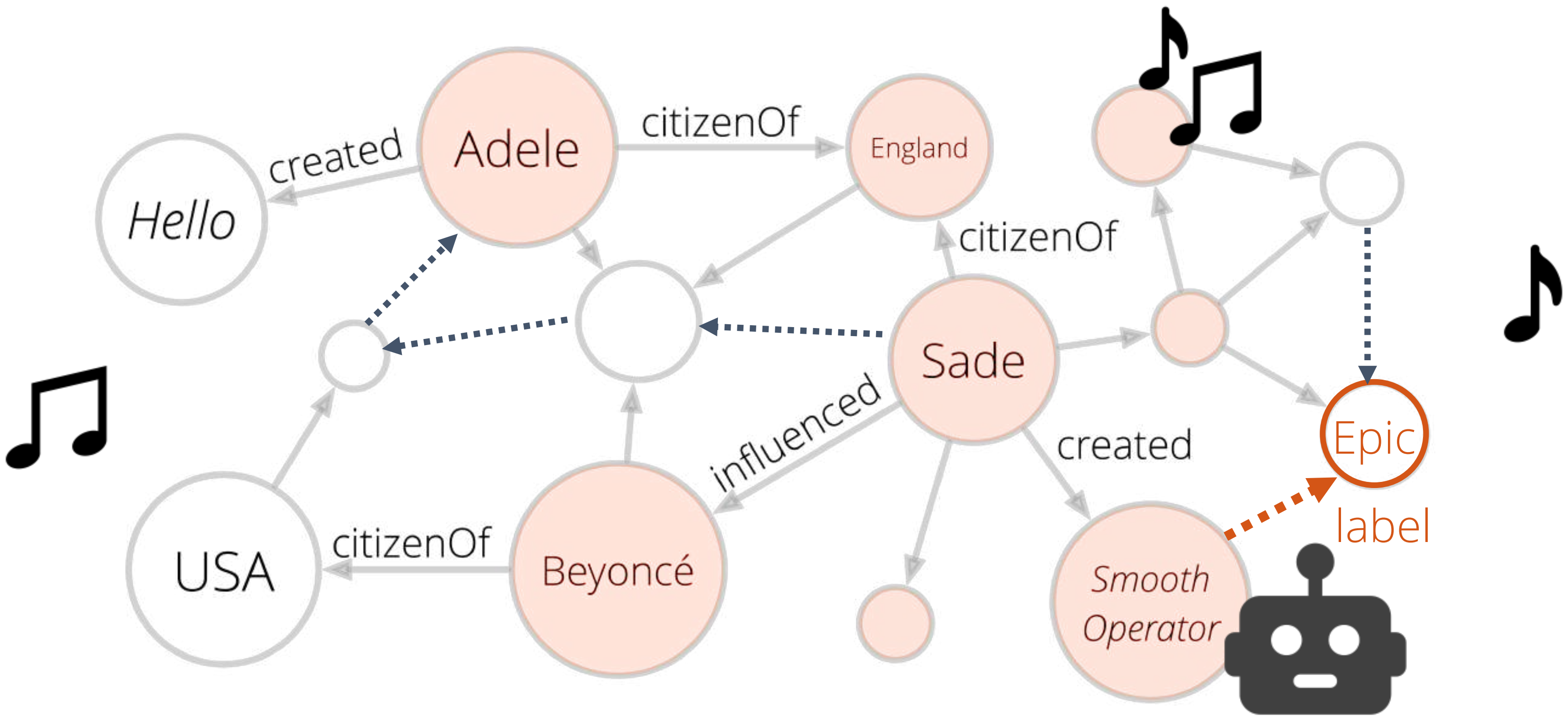
Hire experts or crowd workers

Information extraction from documents

Automate link prediction in KGs

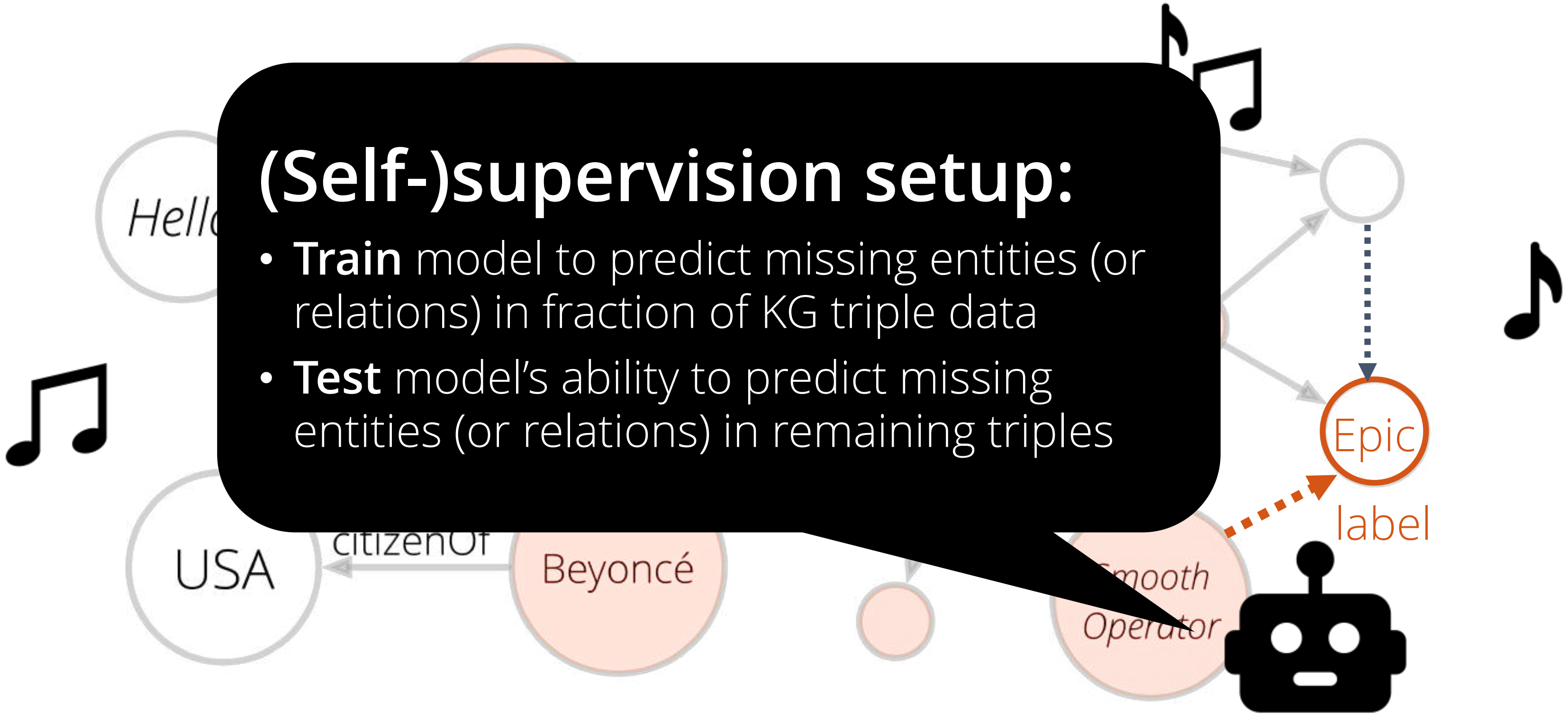
How to improve recall?

KG completion (KGC): Automate prediction of missing edges



Can we automate KGC?

Devise shallow/deep learning models for KGC



(Self-)supervision setup:

- **Train** model to predict missing entities (or relations) in fraction of KG triple data
- **Test** model's ability to predict missing entities (or relations) in remaining triples

Can we automate KGC?

	Reference	FB15K	FB15K-237	FB13	WN18	WN18RR	WN11	Other (Datasets)	Link pred.	Triple class.	Other (Evaluation tasks)
AAAI, IJCAI	(Wang et al., 2014)	✓		✓	✓		✓	FB5M	✓	✓	relation extraction (FB5M)
	(Lin et al., 2015b)	✓		✓	✓		✓	FB40K	✓	✓	relation extraction (FB40K)
	(Wang et al., 2015)							NELL (Location, Sports)	✓		
	(Nickel et al., 2016)	✓			✓			Countries	✓		
	(Lin et al., 2016)							FB24K	✓		
	(Wang and Cohen, 2016)	✓			✓				✓		
	(Xiao et al., 2016a)	✓		✓	✓		✓		✓	✓	
	(Jia et al., 2016)	✓		✓	✓		✓		✓	✓	
	(Xie et al., 2016)	✓						FB15K+	✓	✓	
	(Shi and Weninger, 2017)	✓						SemMedDB, DBPedia	✓		fact checking (not on FB15K)
	(Dettmers et al., 2018)	✓	✓		✓	✓		YAGO3-10, Countries	✓		
	(Ebisu and Ichise, 2018)	✓			✓				✓		
	(Guo et al., 2018)	✓						YAGO37	✓		
	(Zhang et al., 2020)	✓	✓		✓	✓			✓		
	(Vashishth et al., 2020a)		✓			✓		YAGO3-10	✓		
ICML, ICLR, NeurIPS	(Yang et al., 2015)	✓			✓			FB15K-401	✓		rule extraction (FB15K-401)
	(Trouillon et al., 2016)	✓			✓				✓		
	(Liu et al., 2017)	✓			✓				✓		
	(Kazemi and Poole, 2018)	✓			✓				✓		
	(Das et al., 2018)		✓			✓		NELL-995, UMLS, Kinship, Countries, WikiMovies	✓		QA (WikiMovies)
	(Lacroix et al., 2018)	✓	✓		✓	✓		YAGO3-10	✓		
	(Guo et al., 2019)	✓	✓		✓			DBPedia-YAGO3, DBPedia-Wikidata	✓		entity alignment (DBPedia graphs)
	(Sun et al., 2019)	✓	✓		✓	✓			✓		
	(Zhang et al., 2019)	✓	✓		✓	✓			✓		
	(Balazevic et al., 2019a)		✓			✓			✓		
	(Vashishth et al., 2020b)		✓			✓		MUTAG, AM, PTC	✓		graph classification (MUTAG, AM, PTC)

AI/ML conferences

NLP conferences

ACL, EMNLP, NAACL	(Ji et al., 2015)	✓		✓	✓		✓		✓	✓	
	(Guo et al., 2015)							NELL (Location, Sports, Freq)	✓	✓	
	(Guu et al., 2015)			✓			✓		✓	✓	
	(Garcia-Duran et al., 2015)	✓						Families	✓		
	(Lin et al., 2015a)	✓						FB40K	✓		relation extraction (FB40K)
	(Xiao et al., 2016b)	✓		✓	✓		✓		✓	✓	
	(Nguyen et al., 2016)	✓			✓				✓		
	(Xiong et al., 2017)		✓					NELL-995	✓		rule mining
	(Lin et al., 2018)		✓			✓		NELL-995, UMLS, Kinship	✓		
	(Nguyen et al., 2018)		✓			✓			✓		
	(Bansal et al., 2019)		✓			✓			✓		
	(Xu and Li, 2019)	✓	✓		✓	✓		YAGO3-10, Family	✓		
	(Balazevic et al., 2019b)	✓	✓		✓	✓			✓		
	(Vu et al., 2019)		✓			✓		SEARCH17	✓		personalized search (SEARCH17)
	(Nathani et al., 2019)		✓			✓		NELL-995, UMLS, Kinship	✓		
	(Jiang et al., 2019)	✓	✓		✓	✓			✓		

But let's rewind…

Progress in AI research depends heavily on benchmarks

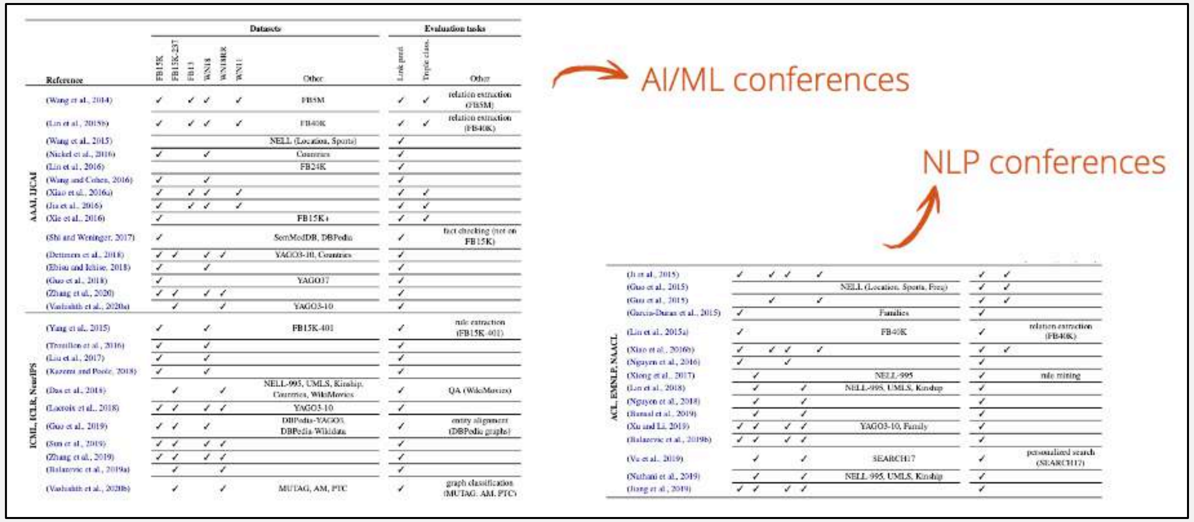
- **Benchmark**: Dataset of input/output pairs that sufficiently represents a real-world use case [Paullada et al 2020]
- Allows for comparison of competing systems (or algorithms) according to given metric(s)

But let’s rewind…

What do benchmarks look like in KGC research?

🤔

Most existing KGC benchmarks*



Reliance on outdated data sources

Leakage between train and test splits

Non-standardized versions

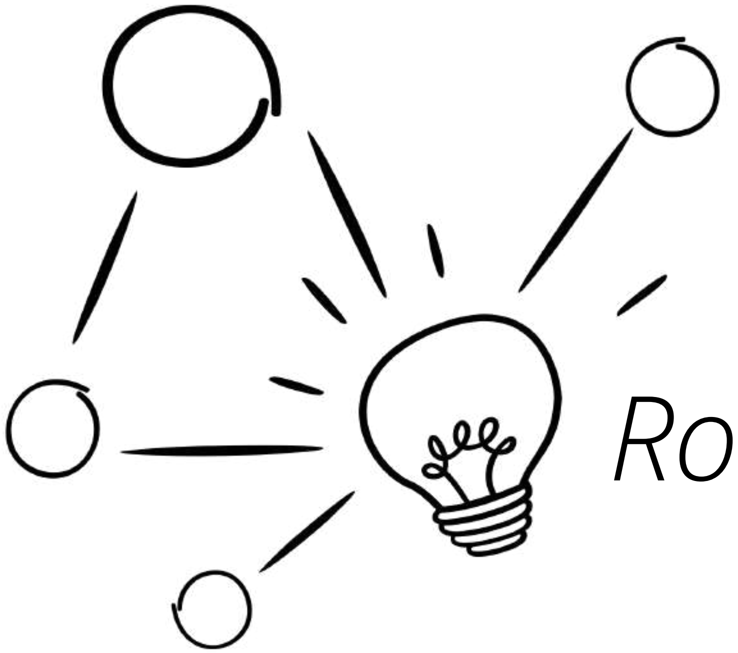
Lack of difficult test examples

Low interpretability for practitioners

*We survey 40+ KGC papers and 12 evaluation datasets across AI/ML/NLP venues

A set of knowledge graph **Co**mpletion **D**atasets **Ex**tracted from Wikidata and Wikipedia

+ everything (data, models, code) is publicly accessible!



Roadmap

How was CoDEx designed and collected?

What KGC tasks can I test on CoDEx?

How does CoDEx compare to existing KGC benchmarks?

How was CoDEx designed and collected?

What KGC tasks can I test on CoDEx?

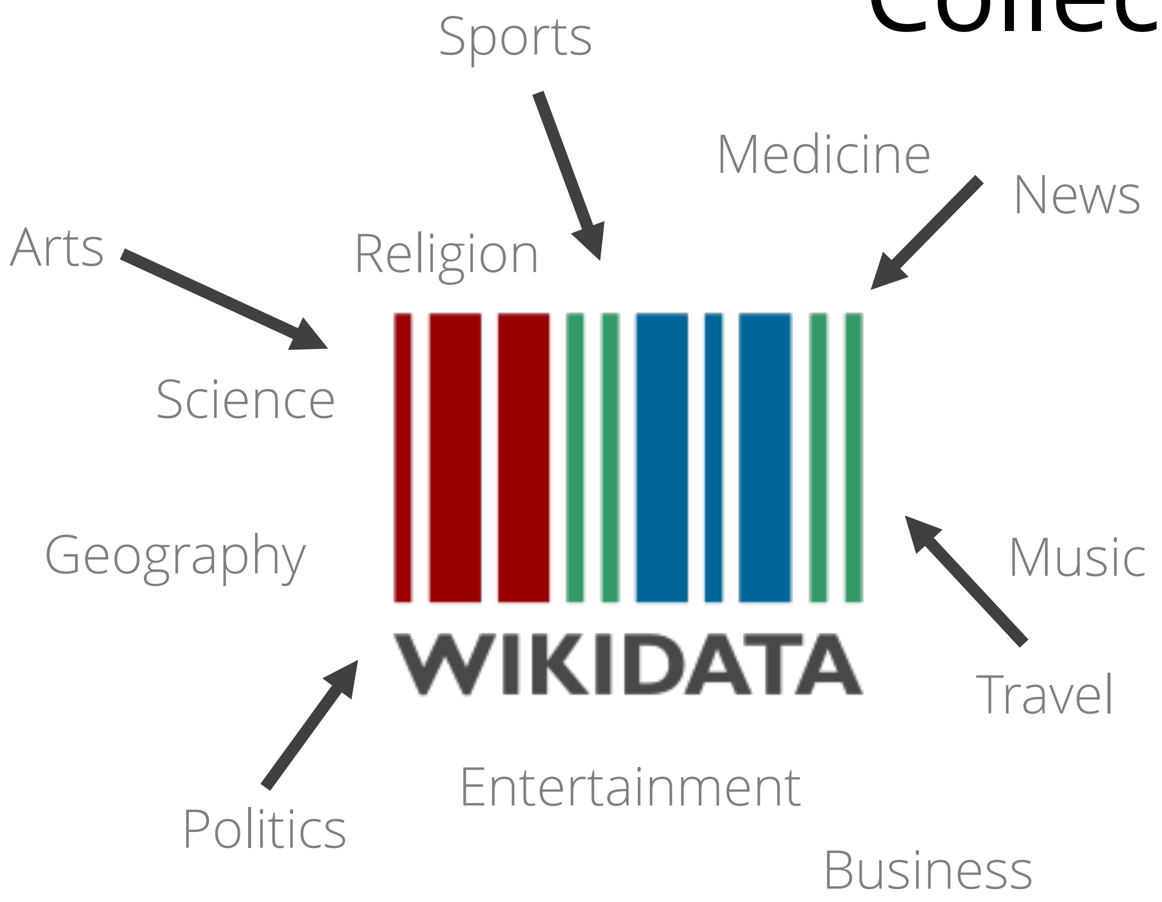
How does CoDEx compare to existing KGC benchmarks?

Collecting CoDEx

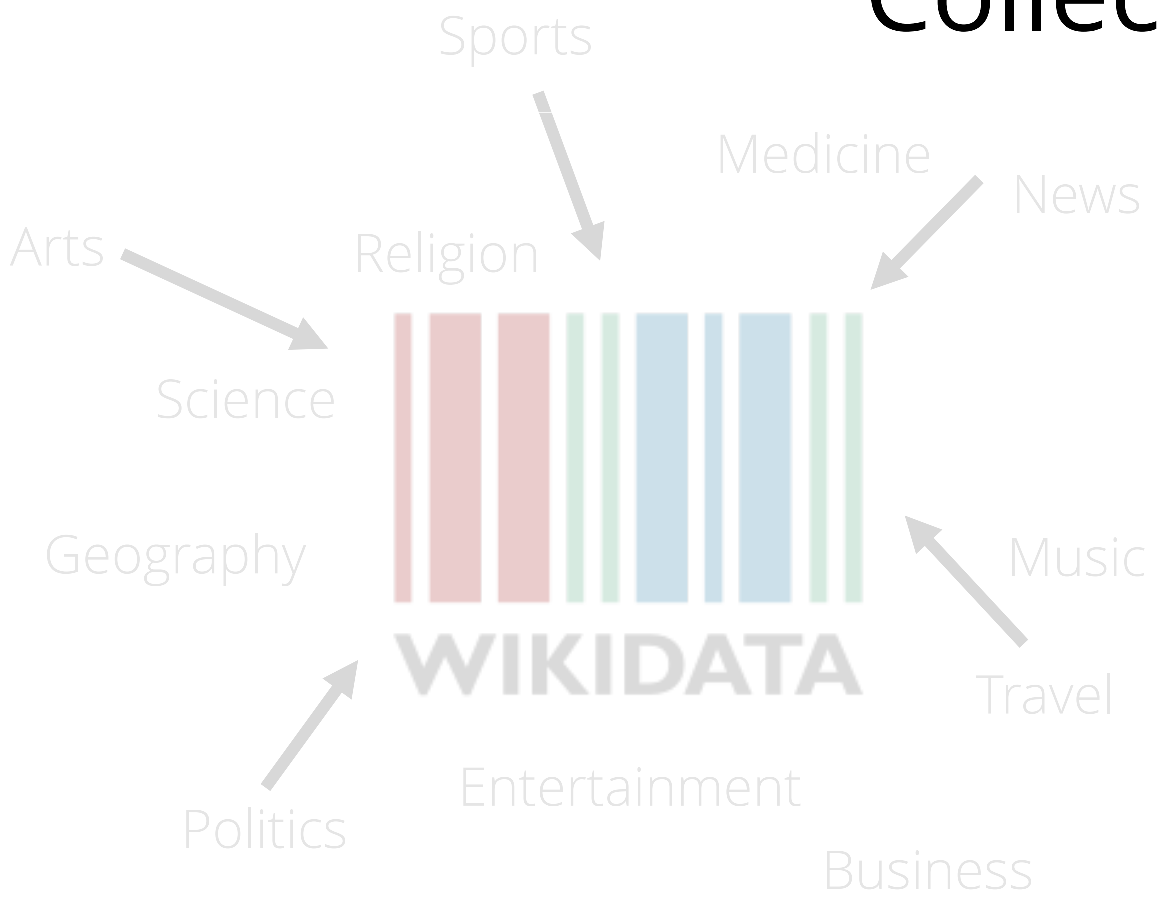
WIKIDATA

CoDEx

Collecting CoDEx



Collecting CoDEx



	# entities	# relations	# triples
CoDEx-S	2K	42	36K
CoDEx-M	17K	51	206K
CoDEx-L	78K	69	612K

```python
import random
codex = Codex(code="en", size="m")

eid = random.choice(list(codex.entities()))
triples = codex.triples()
triples = triples[
    (triples["head"] == eid) | (triples["tail"] == eid)
]

for (head, relation, tail) in triples.values:
    print(f"({codex.entity_label(head)},
          {codex.relation_label(relation)},
          {codex.entity_label(tail)})")
```

```
(Virginia Woolf, country of citizenship, United Kingdom)
(Virginia Woolf, occupation, diarist)
(Virginia Woolf, occupation, feminist)
(Ursula K. Le Guin, influenced by, Virginia Woolf)
(Virginia Woolf, influenced by, George Eliot)
(Virginia Woolf, genre, prose)
(Virginia Woolf, occupation, essayist)
(Leonard Sidney Woolf, spouse, Virginia Woolf)
(Virginia Woolf, genre, drama)
(Samuel R. Delany, influenced by, Virginia Woolf)
(Virginia Woolf, languages spoken, written, or signed, English)
```

Collecting CoDEx

WIKIDATA

WIKIPEDIA

Collecting CoDEx

```
eid = "Q51"

for code in codes:
    codex = Codex(code=code)
    print(codex.entity_label(eid))
```

```
القارة القطبية الجنوبية
Antarktika
Antarctica
Antártida
Антарктида
南极洲
```

```
codex = Codex(code="en")
print(f"From {codex.entity_wikipedia_url(eid)}:")
print(f"  '{codex.entity_extract(eid)[:400]}...'")
```

```
From https://en.wikipedia.org/wiki/Antarctica:
  'Antarctica ( or  (listen)) is Earth's southernmost contin
e, almost entirely south of the Antarctic Circle, and is sur
est continent and nearly twice the size of Australia. At 0.0
```

```
codex = Codex(code="en")
types = codex.entity_types(eid)
for etype in types:
    print(codex.entity_label(eid), "is of type", codex.entit
```

```
Antarctica is of type continent
Antarctica is of type geographic region
```

WIKIPEDIA

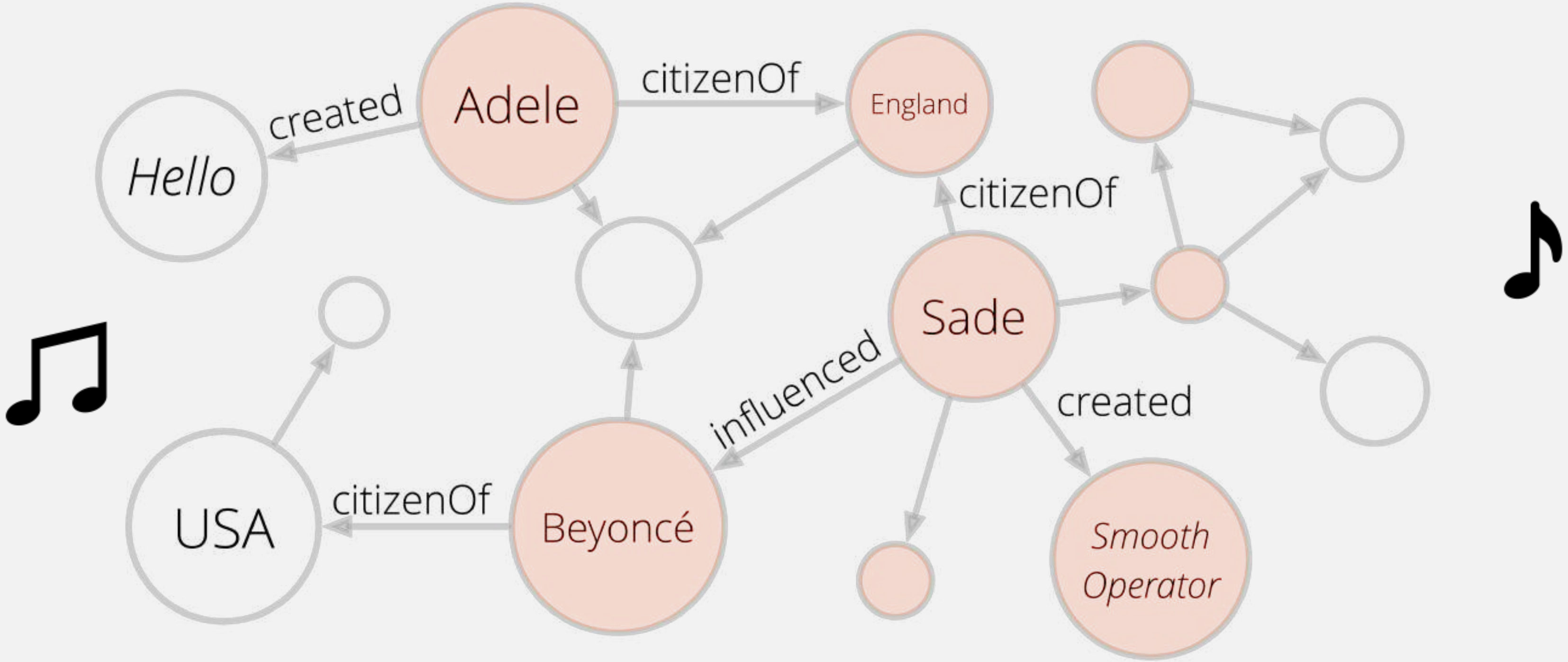
Entity types + text in Arabic, German, English, Spanish, Russian, Chinese

Existing KGC benchmarks in English only (or don't have text)!

CoDEx

Positive and negative knowledge

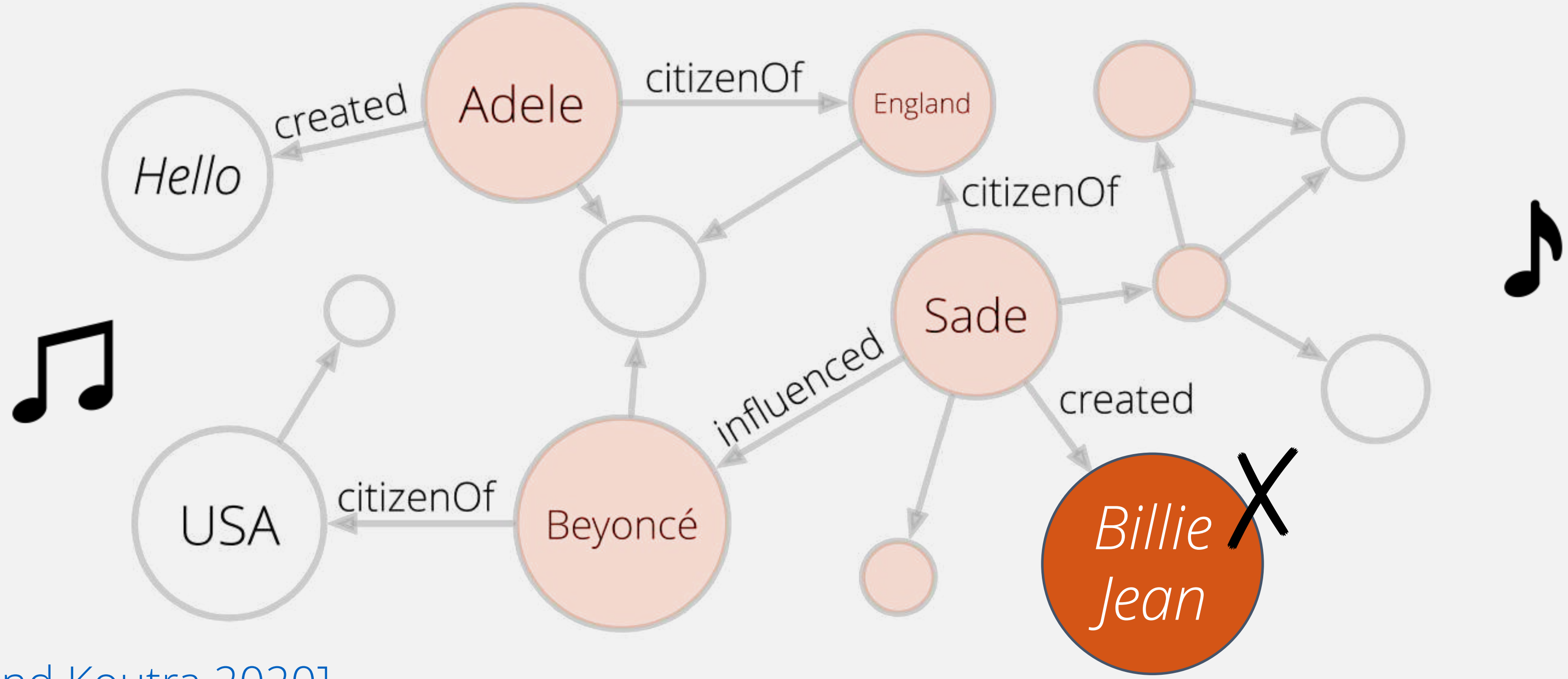
Most KGs include positive (true) data only…



CoDEx

Positive and negative knowledge

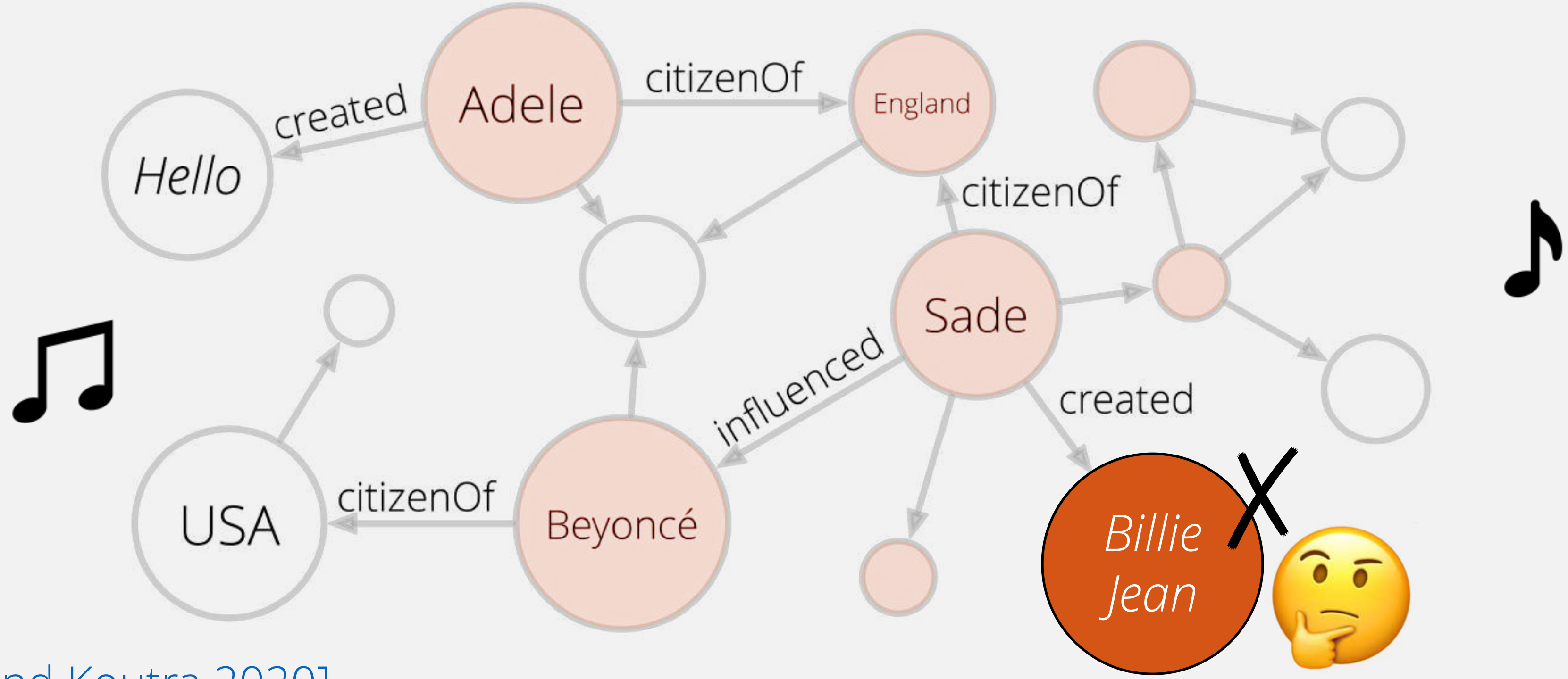
But negative (false) knowledge can be useful too!



CoDEx [Safavi and Koutra 2020]

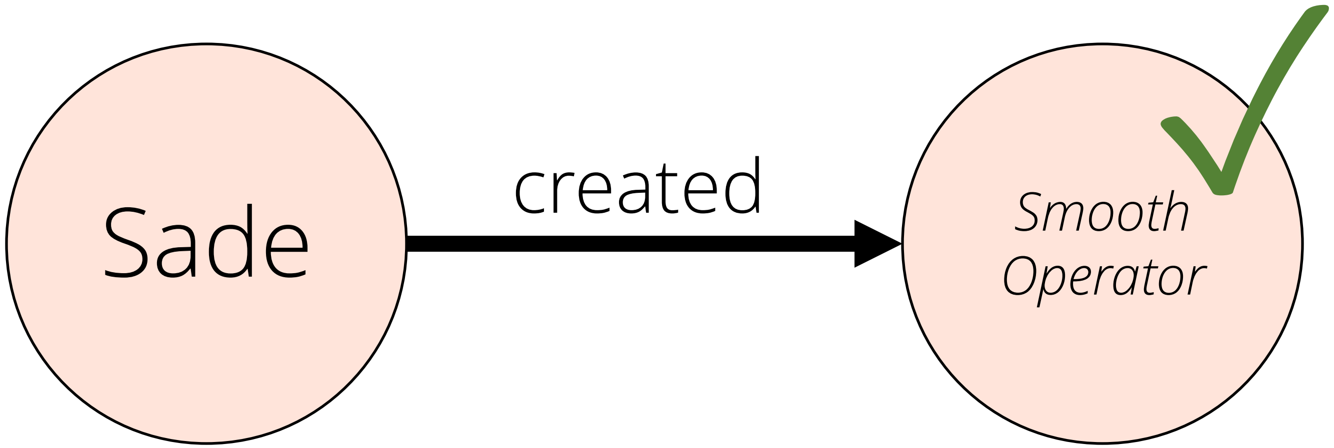
Positive and negative knowledge

How to construct negatives from positives?



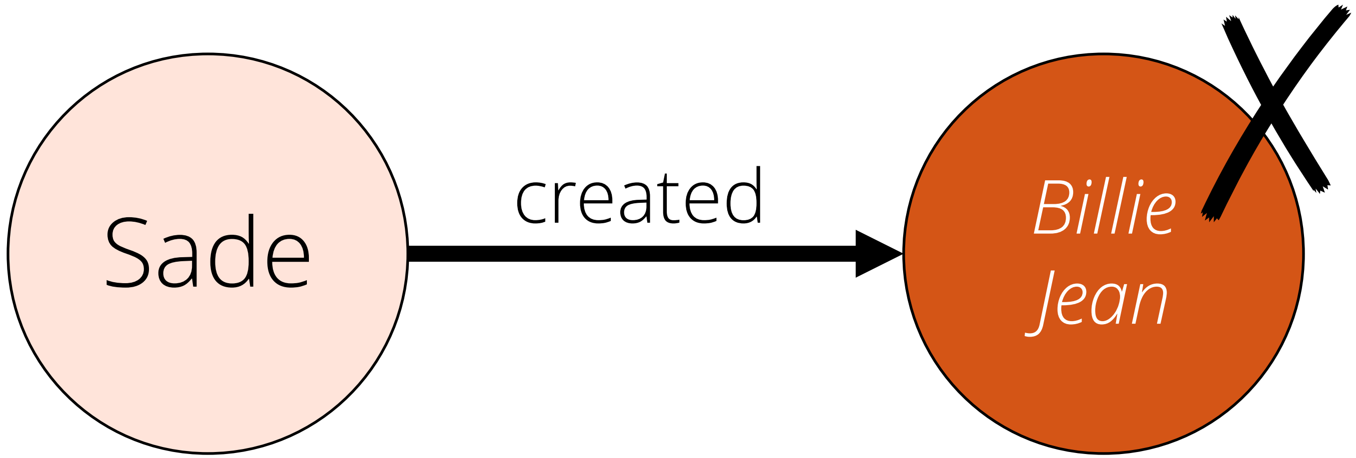
CoDEx [Safavi and Koutra 2020]

Augmenting CoDEx



CoDEx

Augmenting CoDEx



Same type (pop song)
Triple not in KG

Augmenting CoDEx

Table 3: Selected examples of hard negatives in CoDEx with explanations.

Negative	Explanation
(*Frédéric Chopin*, *occupation*, *conductor*)	Chopin was a pianist and a composer, not a conductor.
(*Lesotho*, *official language*, *American English*)	English, not American English, is an official language of Lesotho.
(*Senegal*, *part of*, *Middle East*)	Senegal is part of West Africa.
(*Simone de Beauvoir*, *field of work*, *astronomy*)	Simone de Beauvoir's field of work was primarily philosophy.
(*Vatican City*, *member of*, *UNESCO*)	Vatican City is a UNESCO World Heritage Site but not a member state.

CoDEx

Explore CoDEx.ipynb

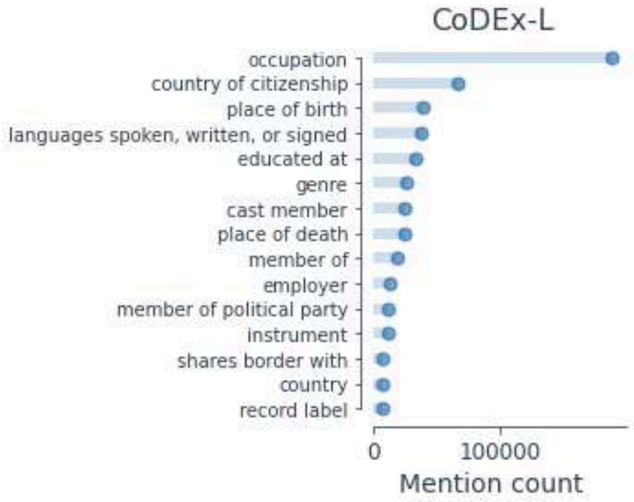
```
count_df = count_relations(triples)
count_df["label"] = [
    codex.relation_label(rid) for rid in count_df["relation"]]

k = 15

ax = plot_top_k(
    count_df,
    k=k,
    color=palette[-1],
    linewidths=6,
    figsize=(5, 4)
)

ax.set_xscale("linear")
ax.set_xlabel("Mention count", fontsize=14)
ax.set_title(codex.name(), fontsize=16)
ax.tick_params("x", labelsize=12)

plt.tight_layout()
plt.show()
```



How was CoDEx designed and collected?

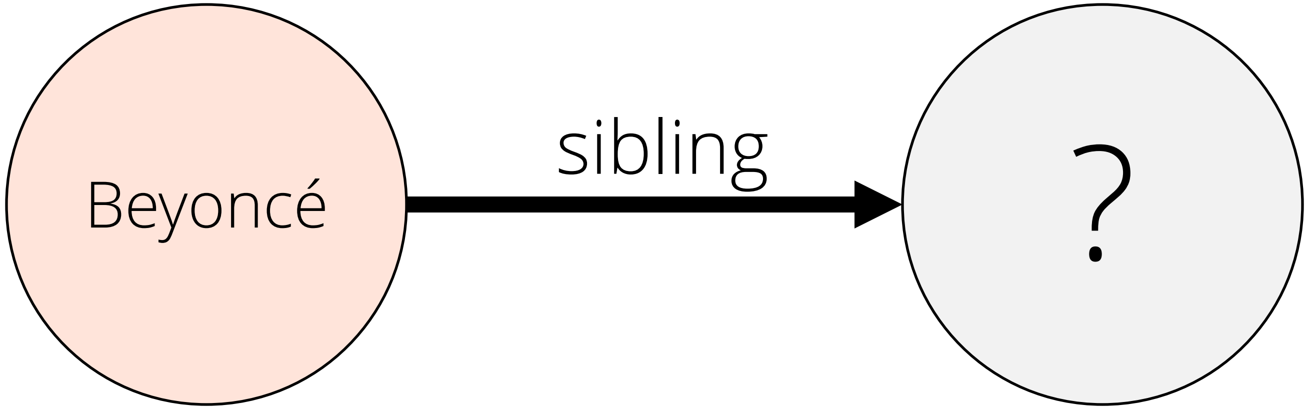
What KGC tasks can I test on CoDEx?

How does CoDEx compare to existing KGC benchmarks?

KGC tasks

Link prediction

Predict answers to queries like (head, relation, ?) and (?, relation, tail) by ranking candidates



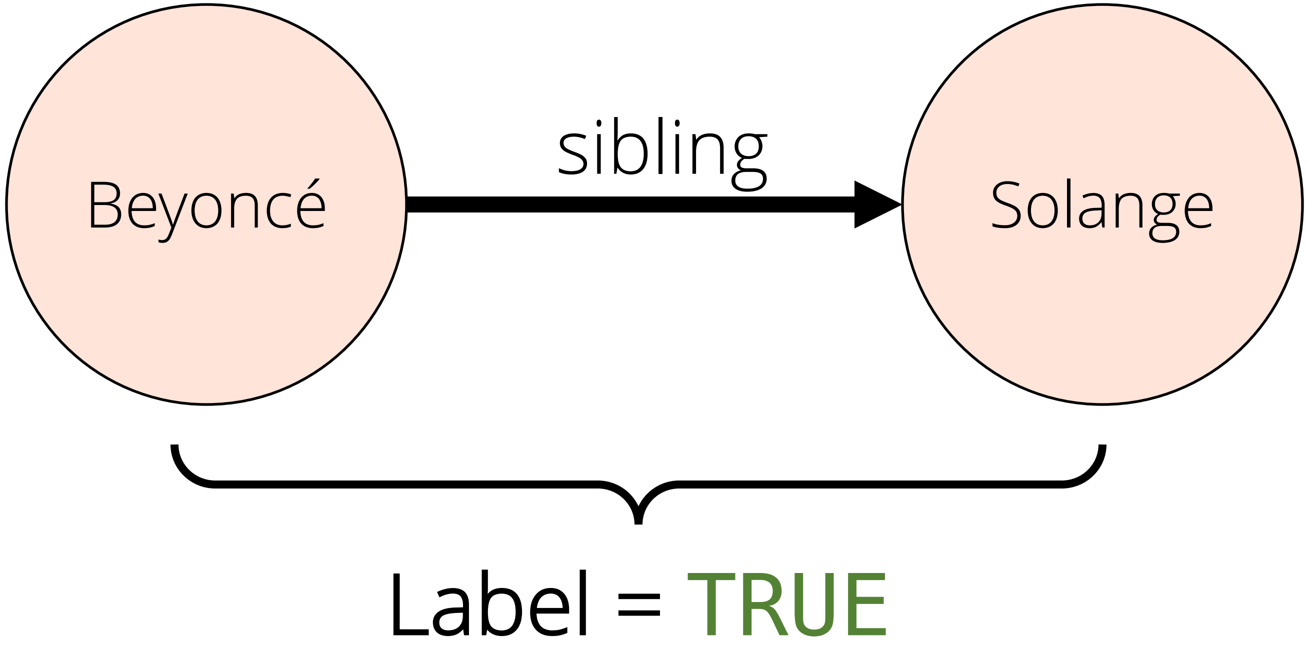
Compute ranking metrics like mean reciprocal rank (MRR) and hits@k

KGC tasks

Triple classification

Label provided triples as true or false

Compute classification metrics like accuracy



Models and implementation

	Type	Approach	Citation
RESCAL	Linear	Tensor decomp.	[Nickel et al 2011]
TransE	Linear	Translational	[Bordes et al 2013]
ComplEx	Linear	Matrix decomp.	[Trouillon et al 2016]
ConvE	Nonlinear	Deep convolutions	[Dettmers et al 2018]
TuckER	Linear	Tensor decomp.	[Balažević et al 2019]

LibKGE A knowledge graph embedding library

Important to fairly compare methods!!

PyTorch

ADAPTIVE EXPERIMENTATION PLATFORM

CoDEx [Broscheit et al 2020] https://github.com/uma-pi1/kge

Link prediction results

Table 5: Comparison of link prediction performance on CoDEx.

Year proposed

	CoDEx-S			CoDEx-M			CoDEx-L		
	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
RESCAL									
TransE									
ComplEx									
ConvE									
TuckER									

Link prediction results

Table 5: Comparison of link prediction performance on CoDEx.

	CoDEx-S			CoDEx-M			CoDEx-L		
	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
RESCAL	0.404	0.293	0.623	0.317	0.244	0.456	0.304	0.242	0.419
TransE	0.354	0.219	0.634	0.303	0.223	0.454	0.187	0.116	0.317
ComplEx	**0.465**	**0.372**	**0.646**	**0.337**	**0.262**	**0.476**	0.294	0.237	0.400
ConvE	0.444	0.343	0.635	0.318	0.239	0.464	0.303	0.240	0.420
TuckER	0.444	0.339	0.638	0.328	0.259	0.458	**0.309**	**0.244**	**0.430**

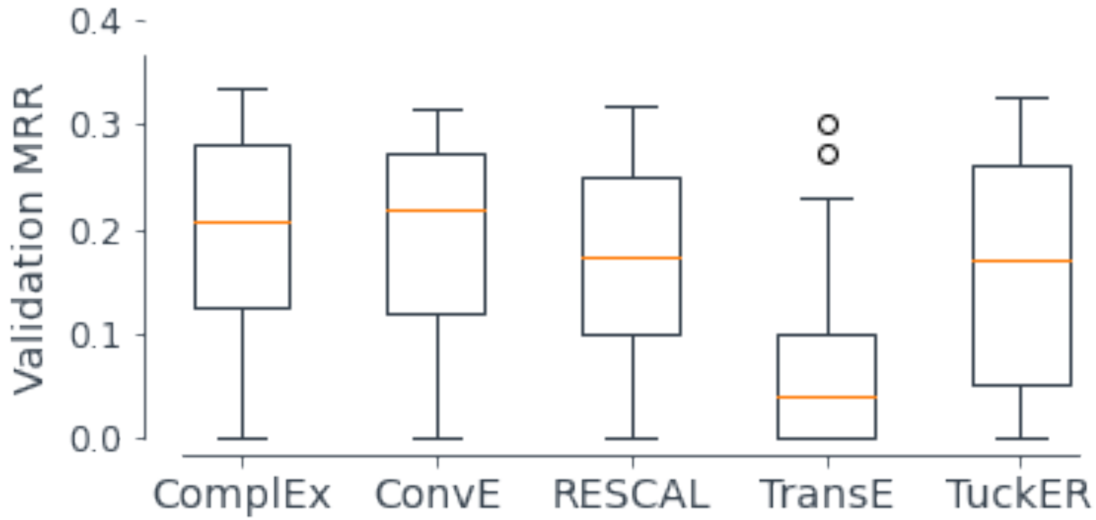
Year proposed

Is "deeper" really better?

For CoDEx-S/M, earlier model (ComplEx) performs best when models are fairly compared!

CoDEx

Link prediction results



Performance varies ±30% based on input hyperparameter configuration, consistent with the literature

Important to properly tune models!!

CoDEx [Ruffinelli et al 2020]

Triple classification results

Table 6: Comparison of triple classification performance on CoDEx by negative generation strategy.

	CoDEx-S						CoDEx-M					
	Uniform		Relative freq.		Hard neg.		Uniform		Relative freq.		Hard neg.	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
RESCAL												
TransE												
ComplEx												
ConvE												
TuckER												

Triple classification results

Table 6: Comparison of triple classification performance on CODEX by negative generation strategy.

	CODEX-S						CODEX-M					
	Uniform		Relative freq.		Hard neg.		Uniform		Relative freq.		Hard neg.	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
RESCAL	0.972	0.972	0.916	0.920	**0.843**	**0.852**	0.977	0.976	0.921	0.922	0.818	0.815
TransE	0.974	0.974	0.919	0.923	0.829	0.837	**0.986**	**0.986**	0.932	0.933	0.797	0.803
ComplEx	**0.975**	**0.975**	**0.927**	**0.930**	0.836	0.846	0.984	0.984	0.930	0.933	0.824	0.818
ConvE	0.972	0.972	0.921	0.924	0.841	0.846	0.979	0.979	**0.934**	**0.935**	**0.826**	**0.829**
TuckER	0.973	0.973	0.917	0.920	0.840	0.846	0.977	0.977	0.920	0.922	0.823	0.816

Accuracy drops up to 19 points on hard negative examples compared to randomly generated negatives

Lots of room for improvement!

CoDEx

How was CoDEx designed and collected?

What KGC tasks can I test on CoDEx?

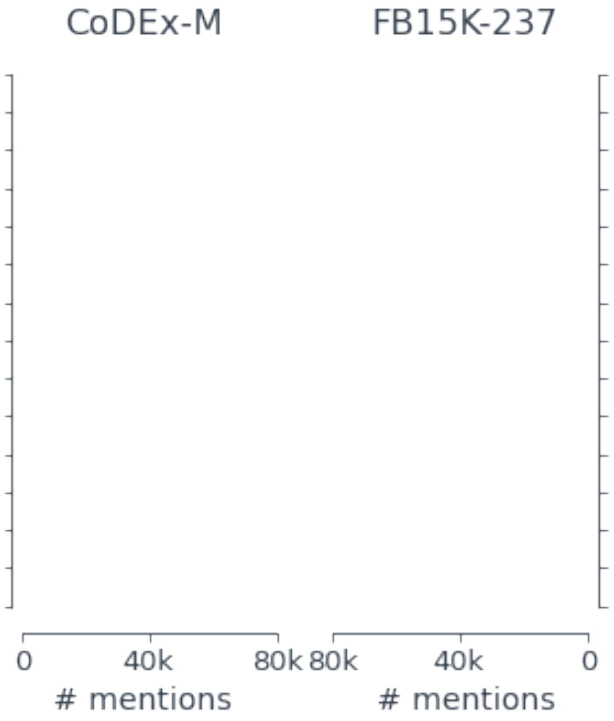
How does CoDEx compare to existing KGC benchmarks?

How does CoDEx compare to existing benchmarks?

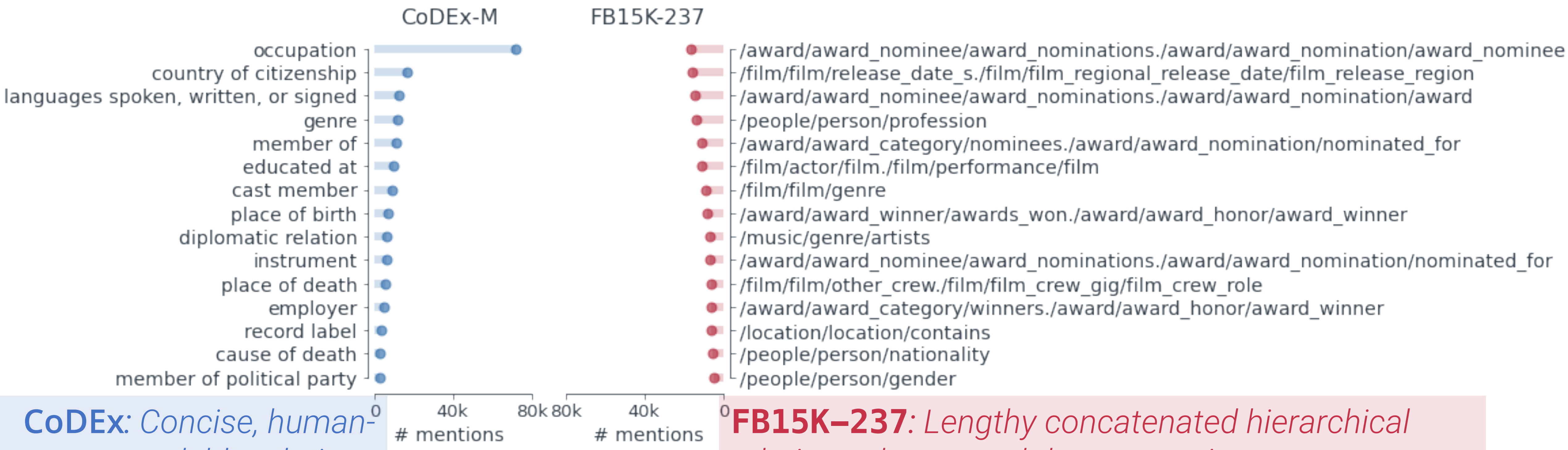
CoDEx **vs** Freebase

FB15K–237
[Toutanova and Chen 2015]

Qualitative comparison



Qualitative comparison



CoDEx: *Concise, human-readable relations*

FB15K–237: *Lengthy concatenated hierarchical relations, deprecated documentation*

CoDEx covers a wider selection of topics and is easier to interpret

CoDEx

Quantitative comparison

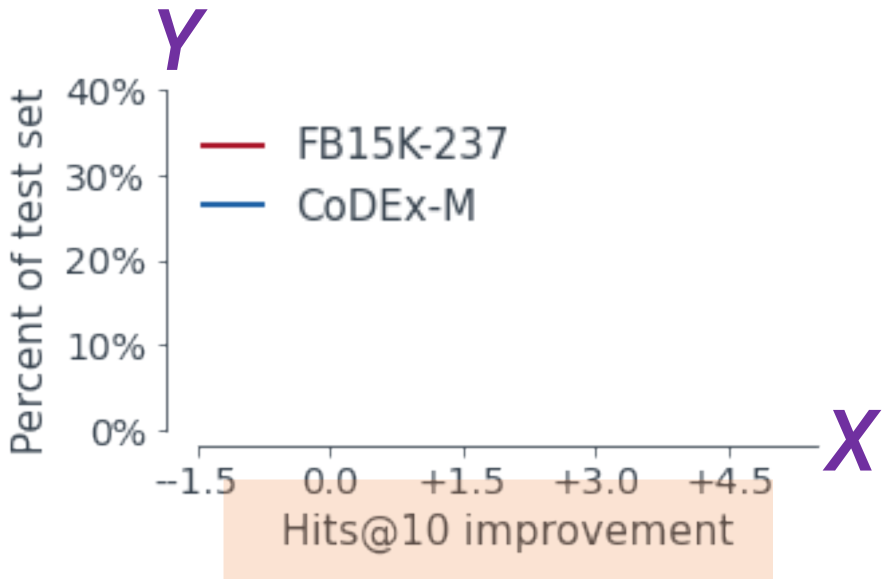
Link prediction

Compare simple non-learning link prediction baseline to SOTA KGC model per dataset, compute improvement over baseline

FB15K–237: *RESCAL [Nickel et al 2011]*

CoDEx: *ComplEx [Trouillon et al 2016]*

Quantitative comparison



*Read as: "Learned KGC model improves ≤ **X** pts over baseline for **Y**% of test set"*

Link prediction

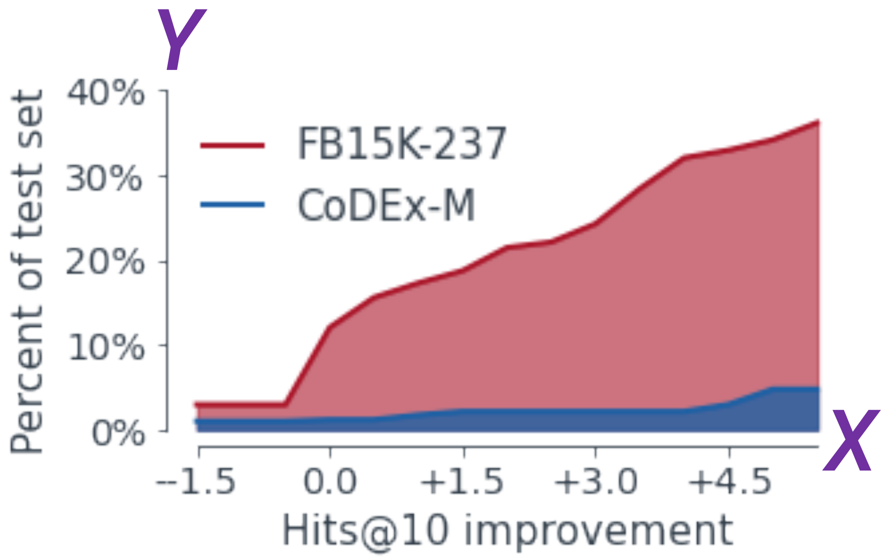
Compare simple non-learning link prediction baseline to SOTA KGC model per dataset, compute improvement over baseline

FB15K–237: *RESCAL [Nickel et al 2011]*

CoDEx: *ComplEx [Trouillon et al 2016]*

CoDEx

Quantitative comparison



*Read as: "Learned KGC model improves ≤ **X** pts over baseline for **Y**% of test set"*

FB15K–237 easier?

- *Baseline performs better than KGC model for 10%*
- *Baseline within 5 pts of KGC model for 40%*

FB15K–237: *very biased toward high-degree entities ("USA", "male")*

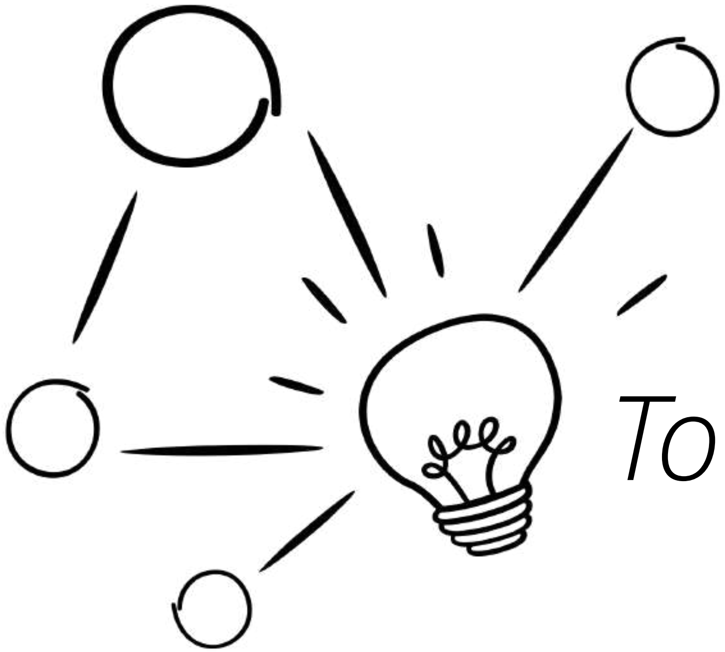
CoDEx: *Designed to avoid this!*

CoDEx

How was CoDEx designed and collected?

What KGC tasks can I test on CoDEx?

How does CoDEx compare to existing KGC benchmarks?

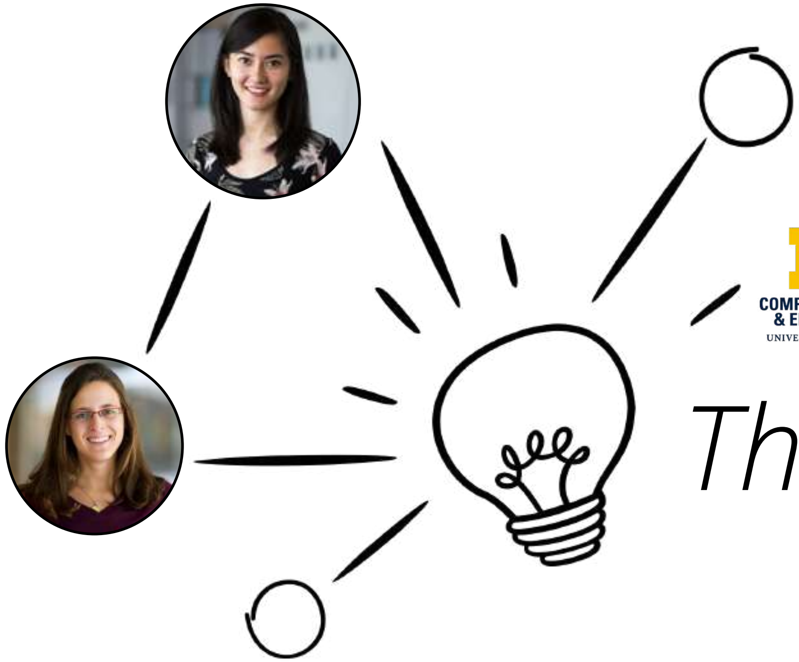


To recap, we:

Introduced and described CoDEx

Benchmarked KGC models on CoDEx for two tasks

Showed its value over a popular KGC dataset



COMPUTER SCIENCE
& ENGINEERING
UNIVERSITY OF MICHIGAN

Thanks for watching!

tsafavi@umich.edu

@tararootcake

tsafavi.github.io

github.com/tsafavi/codex