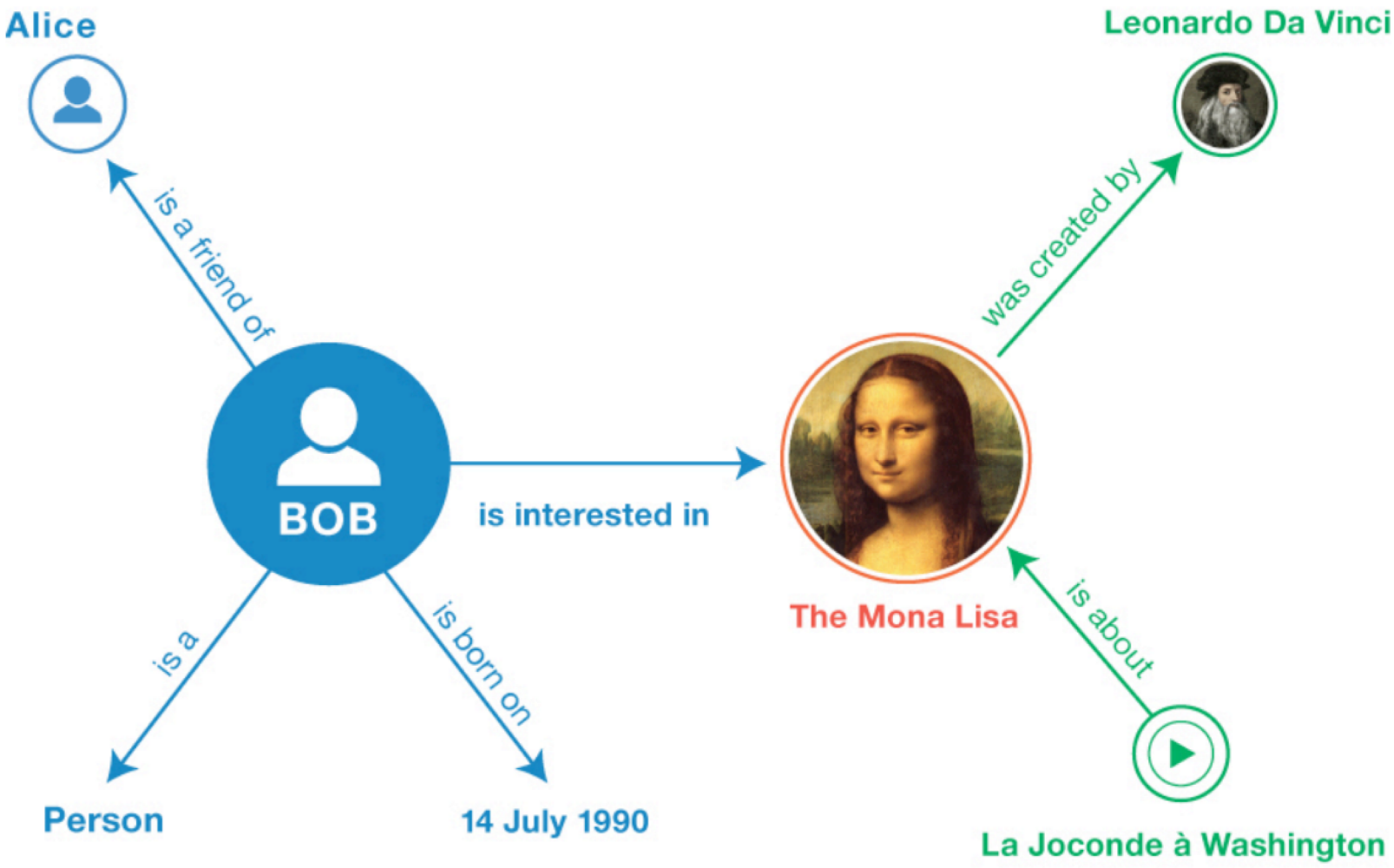
Reasoning in Knowledge Graphs using Embeddings

Joint work with H. Ren, W. Hamilton, R. Ying, J. You, M. Zitnik, D. Jurafsky

Jure Leskovec

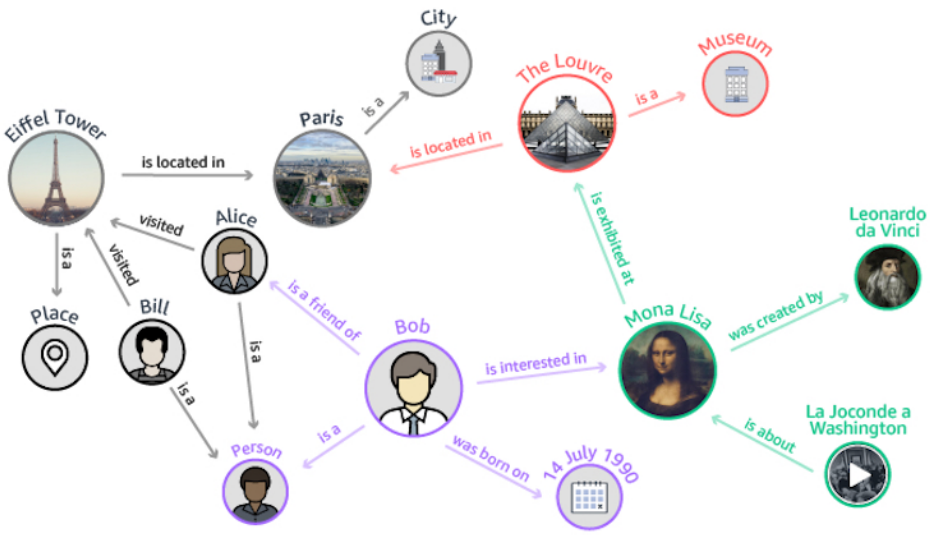
CHAN ZUCKERBERG
BIOHUB

Knowledge Graphs



Knowledge Graphs

- Knowledge Graphs are heterogenous graphs
  - Multiple types of entities and relations exist
- Facts are represented as triples (h, r, t)
  - ('Alice', 'friend_with', 'Bob')
  - ('Paris', 'is_a', 'City')
  - …



Traditional Tasks

Knowledge Graph Competion/Link Prediction

- Predict the missing head or tail for a given triple (h, r, t)
- Example:

Barack Obama BornIn United States

Barack Obama Nationality American

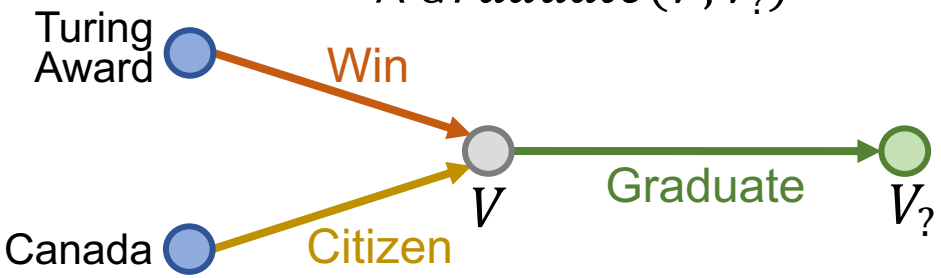
Our work: Beyond Link Prediction

Our goal: Reason over the knowledge graph using complex multi-hop queries

- Logical queries: Subset of first-order logic with existential quantifier (∃), conjunction (∧) and disjunction (∨)

"Where did all Canadian citizens with Turing Award graduate?"

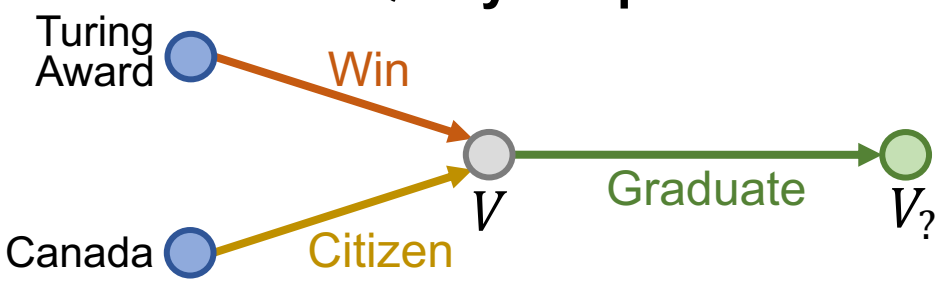
$$q = V_? \,.\, \exists\, V : Win(TuringAward, V) \wedge Citizen(Canada, V) \wedge Graduate(V, V_?)$$



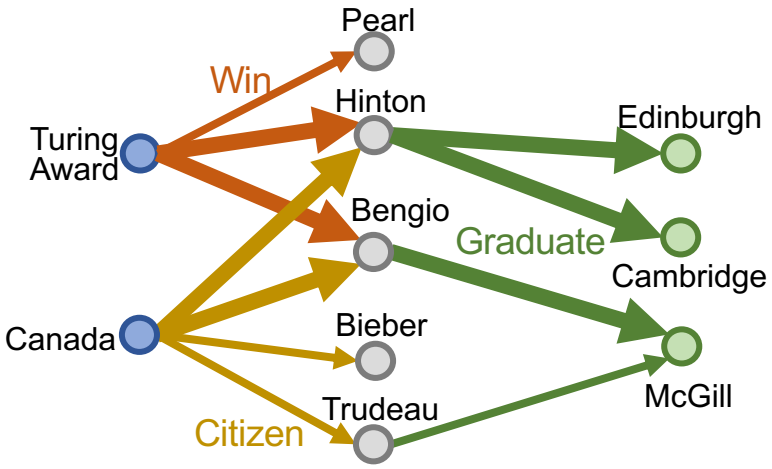
Answering Queries in KGs

"Where did Canadian citizens with Turing Award graduate?"

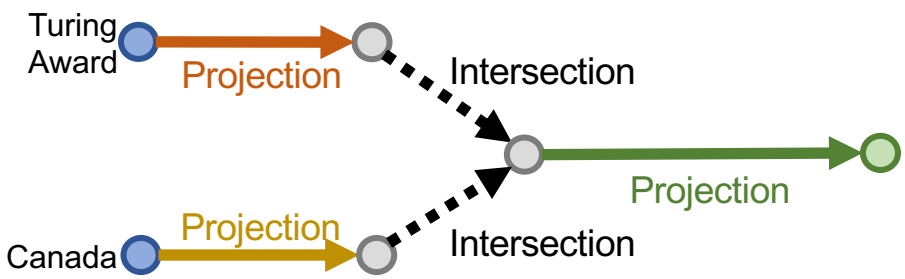
Query Graph



Knowledge Graph



Computation Graph



Each point corresponds to a set of entities

Why is it Hard?

- Heterogeneity: Lack of schema, or quite large schema (65K for DBpedia)
- Noise and incompleteness
- Uncertainty
- Massive size
- Fast query time

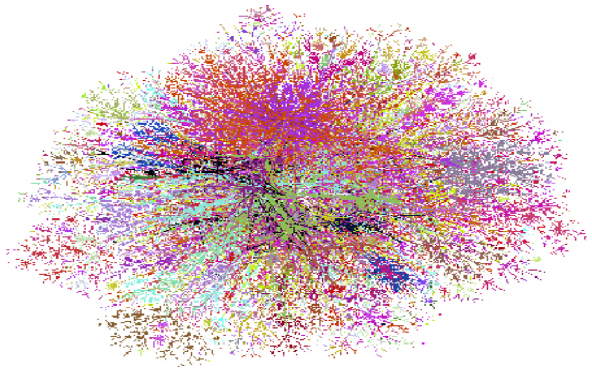
PubID	Publisher	PubAddress
03-4472822	Random House	123 4th Street, New York
04-7733903	Wiley and Sons	45 Lincoln Blvd, Chicago
03-4859223	O'Reilly Press	77 Boston Ave, Cambridge
03-3920886	City Lights Books	99 Market, San Francisco

AuthorID	AuthorName	AuthorBDay
345-28-2938	Haile Selassie	14-Aug-92
392-48-9965	Joe Blow	14-Mar-15
454-22-4012	Sally Hemmings	12-Sept-70
663-59-1254	Hannah Arendt	12-Mar-06

ISBN	AuthorID	PubID	Date	Title
1-34532-482-1	345-28-2938	03-4472822	1990	Cold Fusion for Dummies
1-38482-995-1	392-48-9965	04-7733903	1985	Macrame and Straw Tying
2-35921-499-4	454-22-4012	03-4859223	1952	Fluid Dynamics of Aquaducts
1-38278-293-4	663-59-1254	03-3920886	1967	Beads, Baskets & Revolution

Relational Data (Structured)

vs.

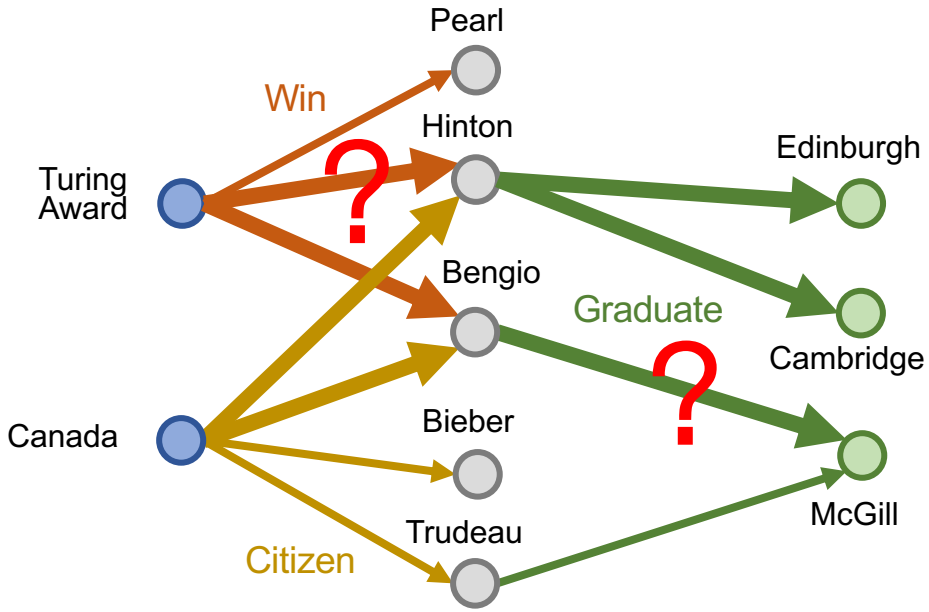


Heterogeneous Graph Data (Semi-structured)

Why is it Hard?

Key challenge: Big graphs and queries can involve noisy and unobserved data!

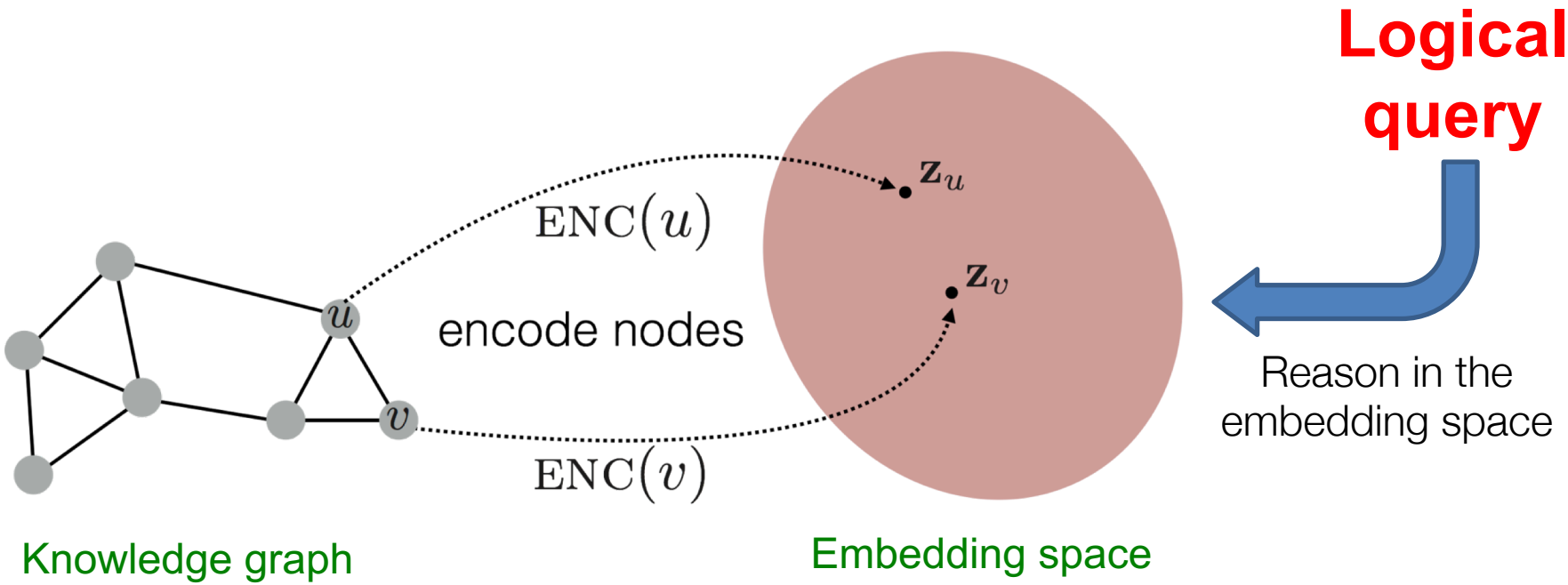
Some links might be noisy or missing



Problem: Naïve link prediction and graph template matching are too expensive

Our Idea: **Query2Box**

Use representation learning to map a graph into a Euclidean space and learn to reason in that space



Our Idea: **Query2Box**

Idea:

- **1)** Embed nodes of the graph
- **2)** For every logical operator learn a spatial operator

So that:

- **1)** Take an arbitrary logical query. Decompose it into a set of logical operators (∃,∧,∨)
- **2)** Apply a sequence of spatial operators to embed the query
- **3)** Answers to the query are entities close to the embedding of the query

Our Idea: **Query2Box**

Idea:

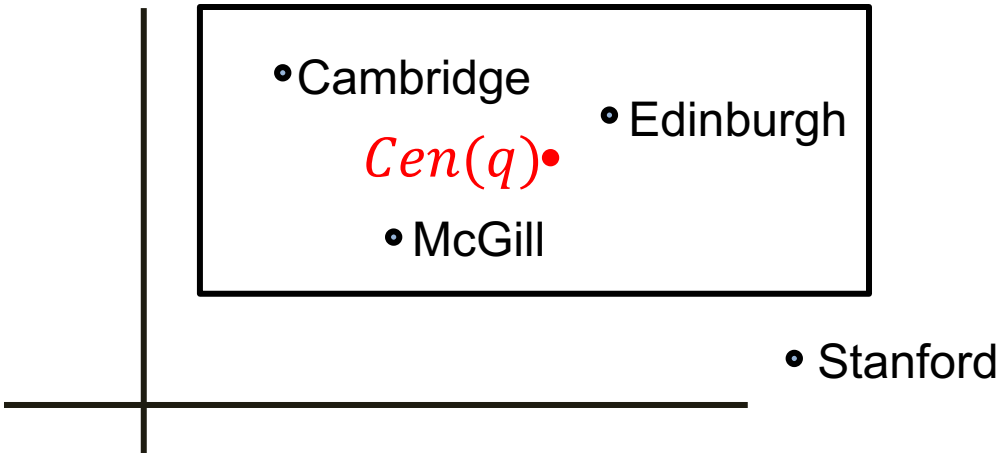
- **1)** Embed nodes of the graph
- **S**
- ose
- **3)** Answers to the query are entities close to the embedding of the query

Key insight:
Represent query as a box.
Operations (union, intersection) are well defined over boxes.

Embedding Queries

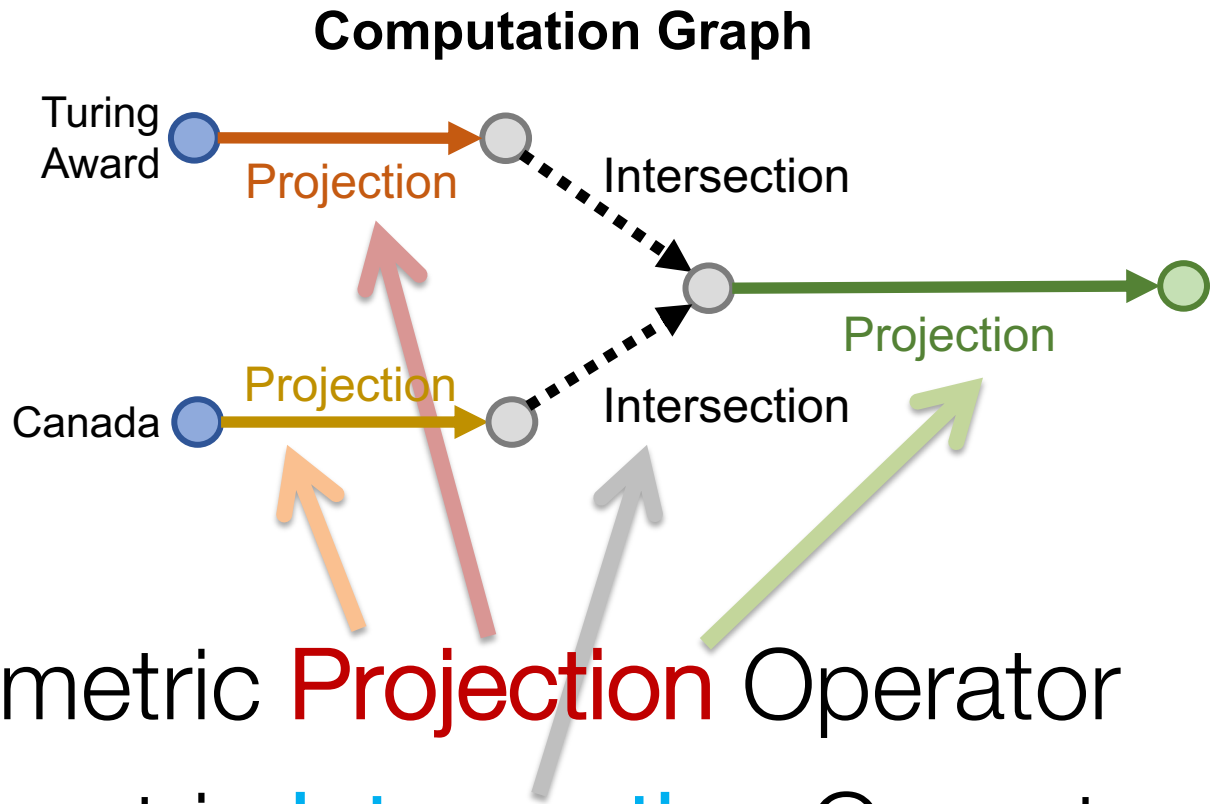
Query2Box embedding:

Embed queries with hyper-rectangles (boxes): $\mathbf{q} = (Cen(q), Off(q))$.



[Probabilistic Embedding of Knowledge Graphs with Box Lattice Measures. Vilnis, et al., ACL 2018]

Embedding Queries

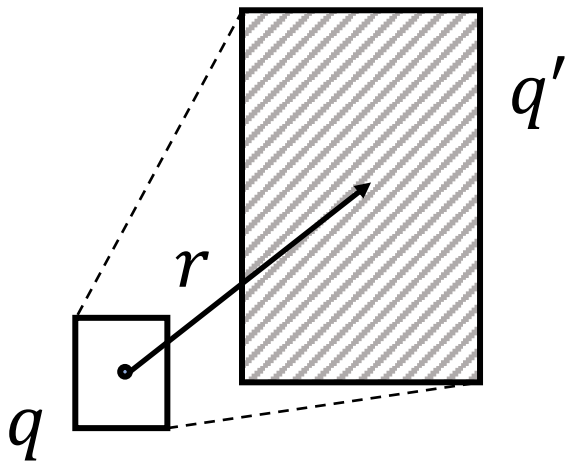


- Geometric Projection Operator
- Geometric Intersection Operator

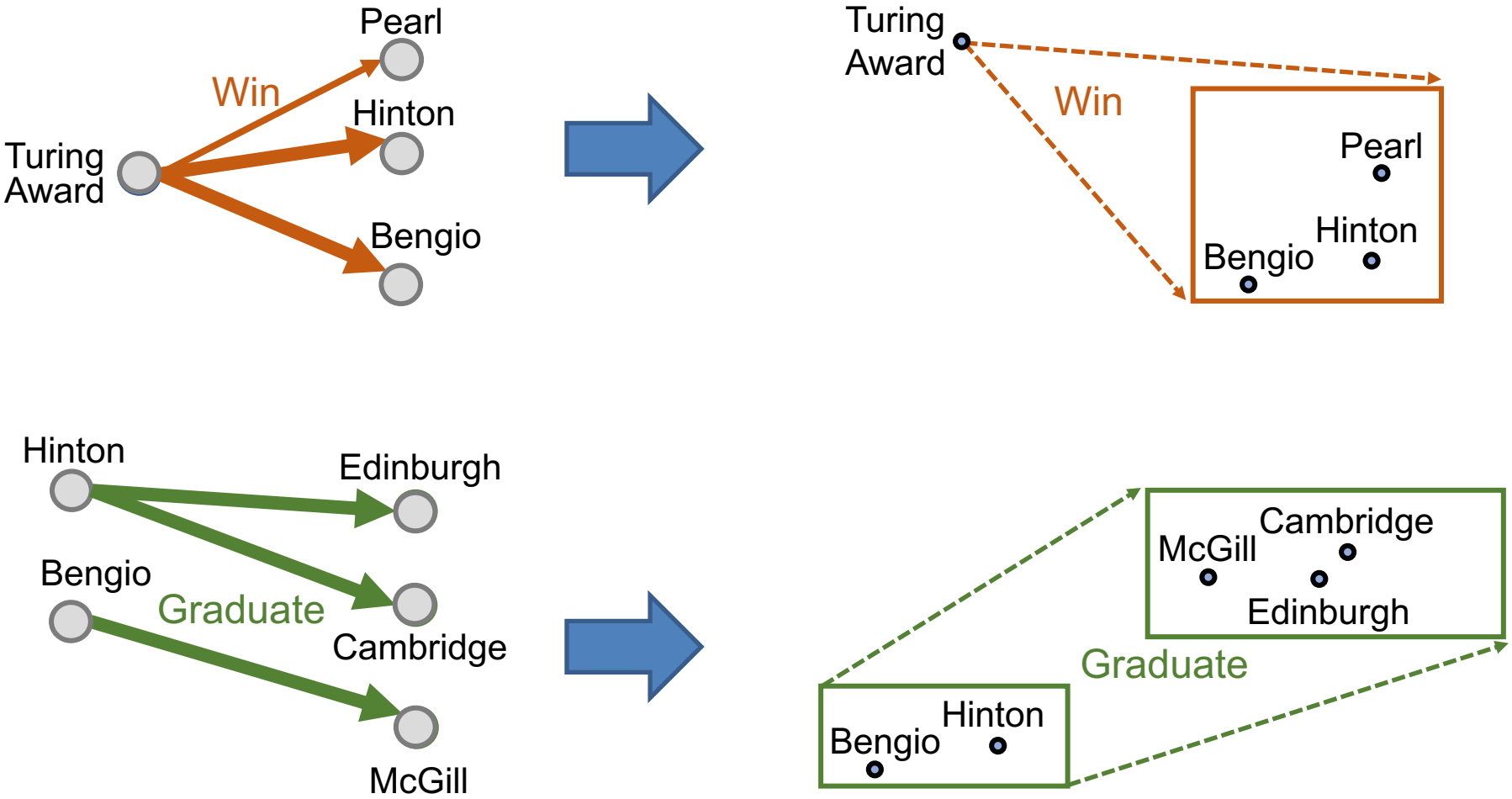
Projection Operator

Geometric Projection Operator $\mathcal{P}$

- $\mathcal{P}$: Box × Relation → Box



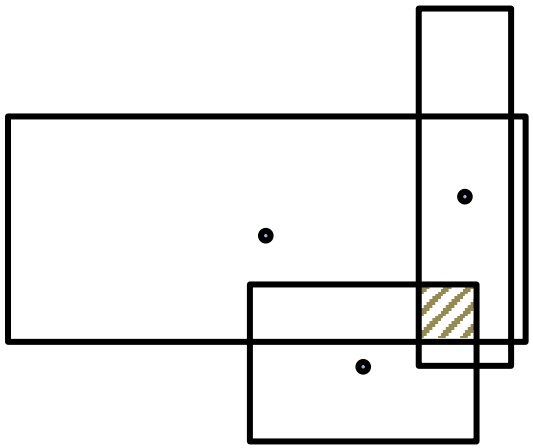
Projection Operator: Example



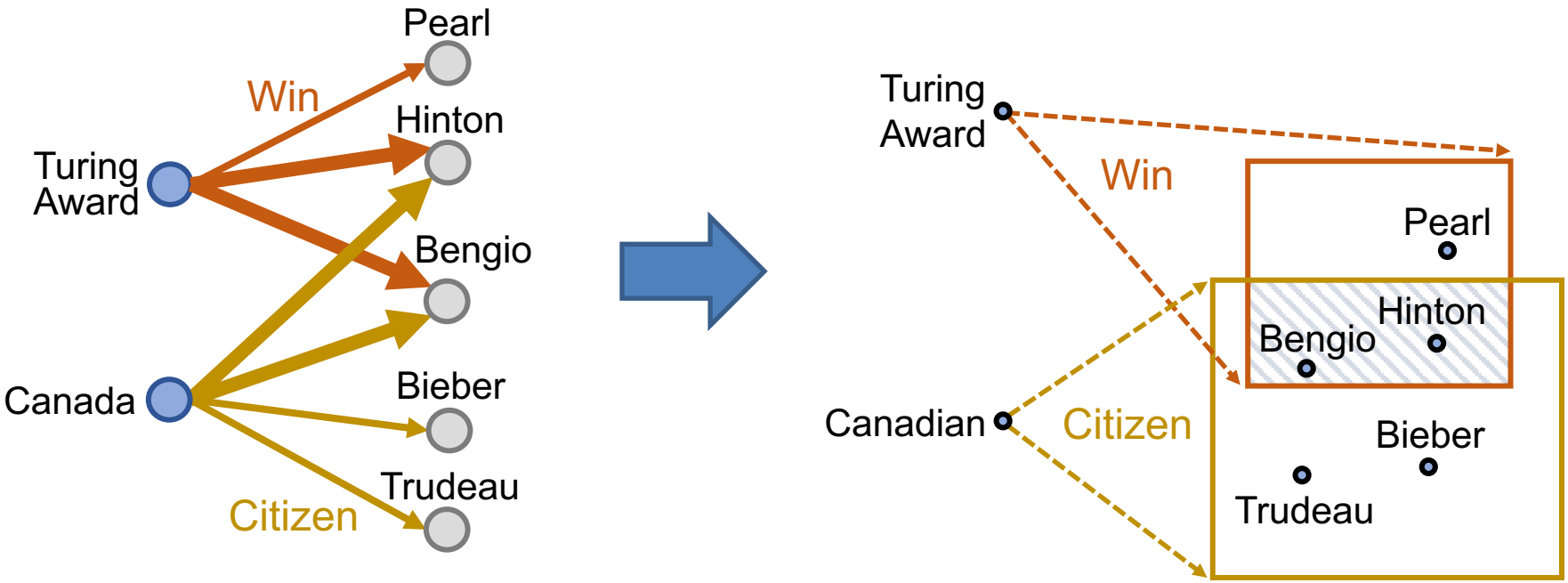
Intersection Operator

Geometric Intersection Operator $\mathcal{I}$

- $\mathcal{I}$: Box × ⋯ × Box → Box
  - The new center is a weighted average
  - The new offset shrinks



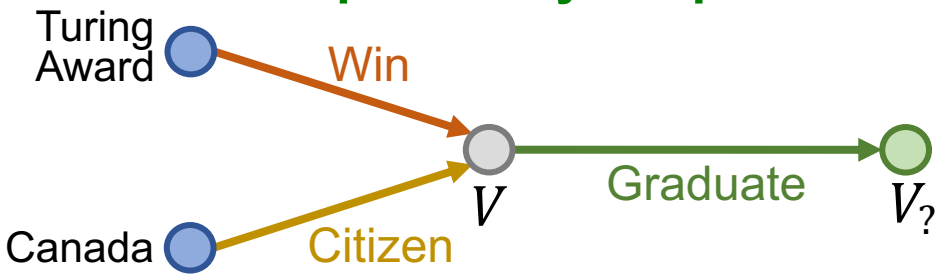
Intersection Operator: Example



Example

"Where did Canadian citizens with Turing Award graduate?"

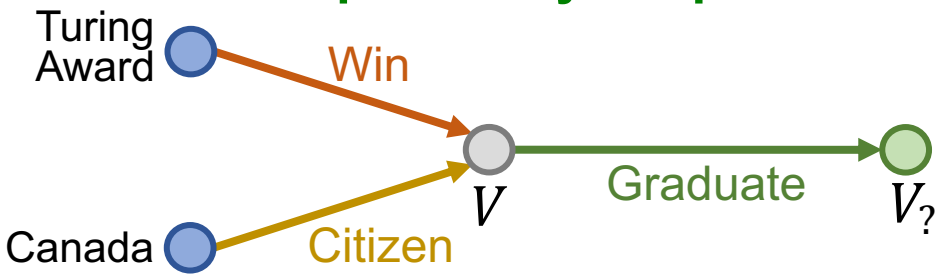
Dependency Graph



Example

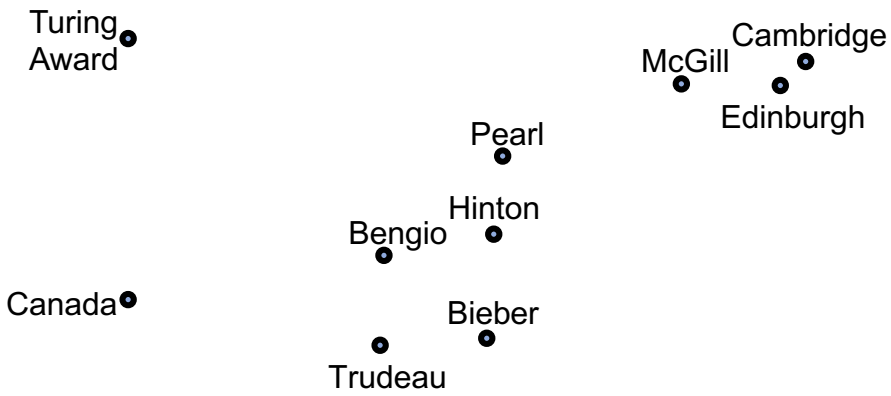
"Where did Canadian citizens with Turing Award graduate?"

Dependency Graph



Computation Graph

Embedding Process

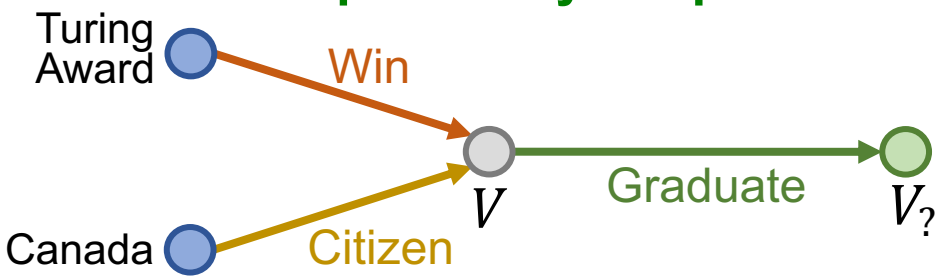


Each point corresponds to a set of entities

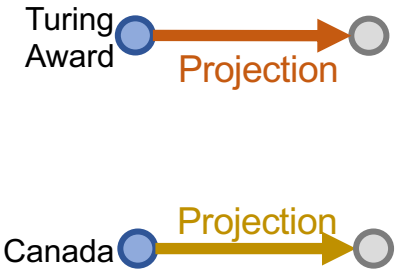
Example

"Where did Canadian citizens with Turing Award graduate?"

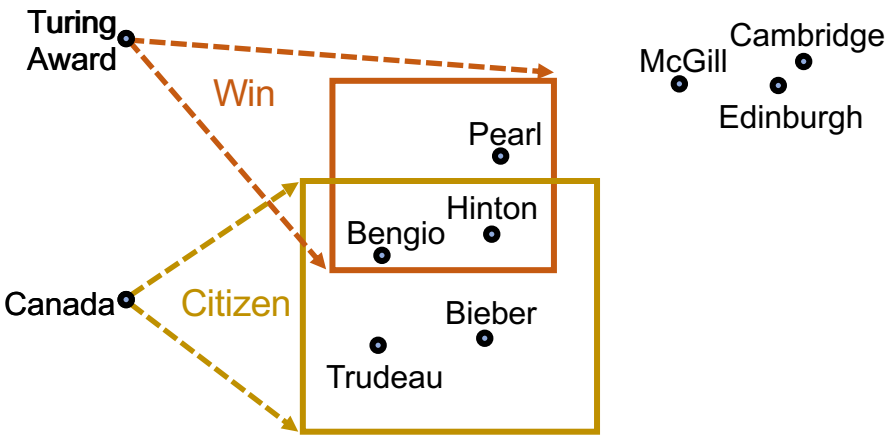
Dependency Graph



Computation Graph



Embedding Process

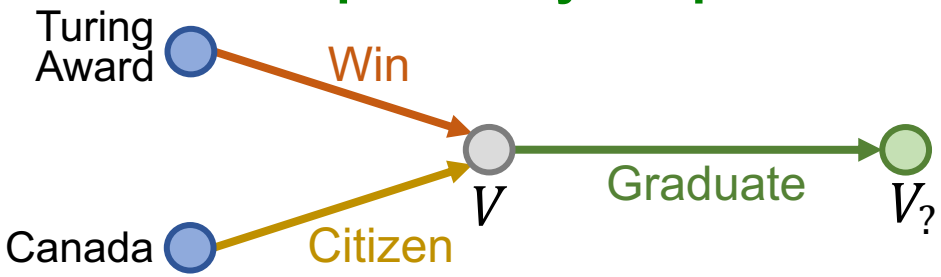


Each point corresponds to a set of entities

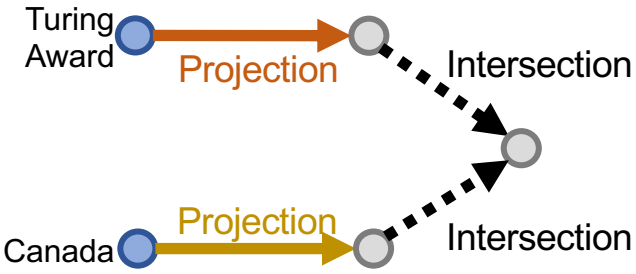
Example

"Where did Canadian citizens with Turing Award graduate?"

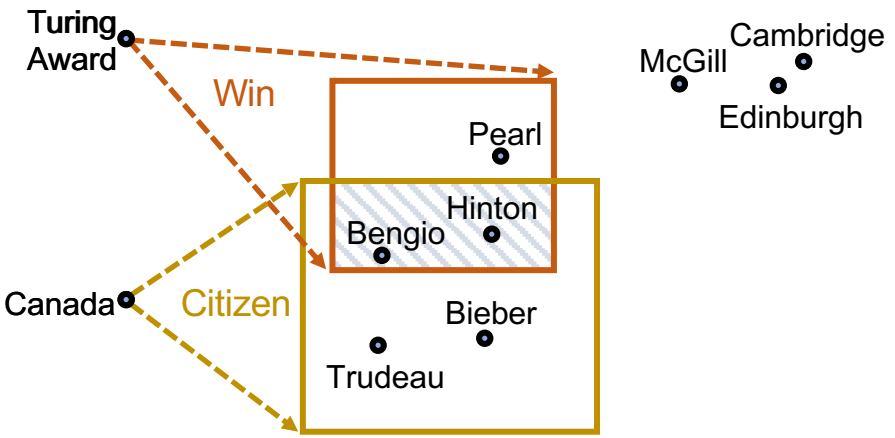
Dependency Graph



Computation Graph



Embedding Process

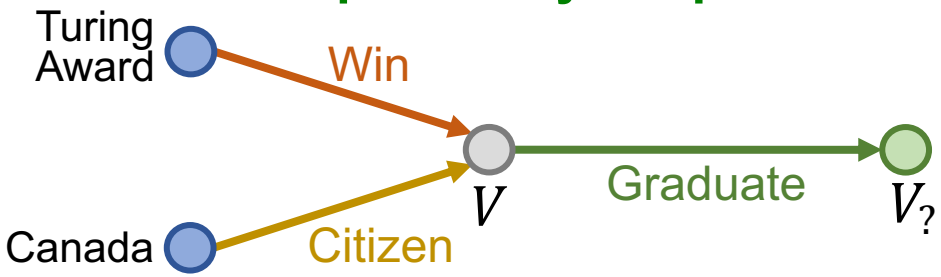


Each point corresponds to a set of entities

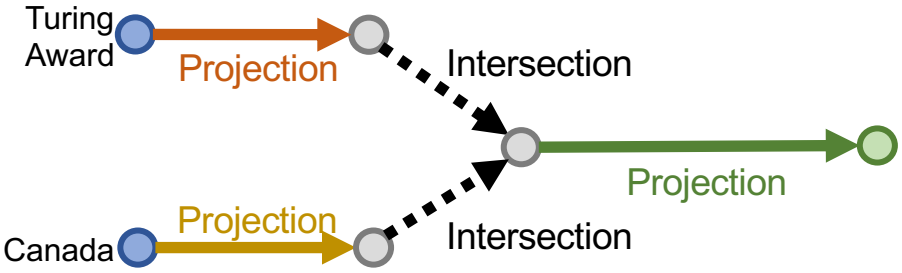
Example

"Where did Canadian citizens with Turing Award graduate?"

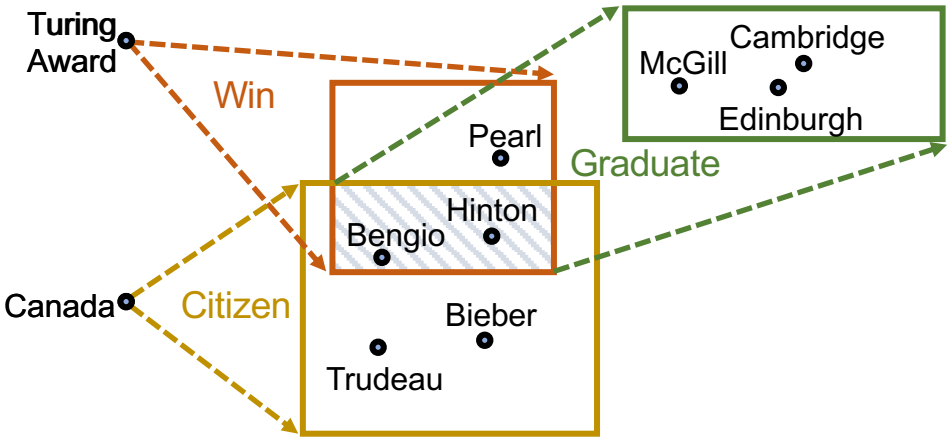
Dependency Graph



Computation Graph



Embedding Process



Each point corresponds to a set of entities

How to Handle Disjunction

So far we can handle Conjunctive queries

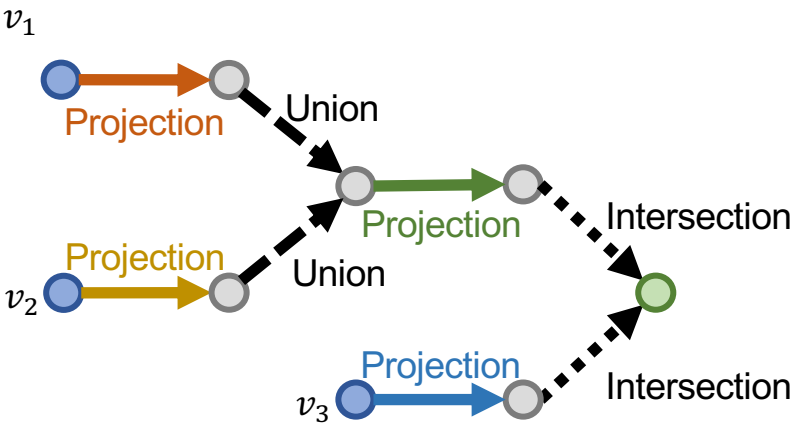
Can we learn a geometric disjunction operator?

- Theorem (paraphrased): For a KG with M nodes, we need embedding dimension of M to handle disjunction

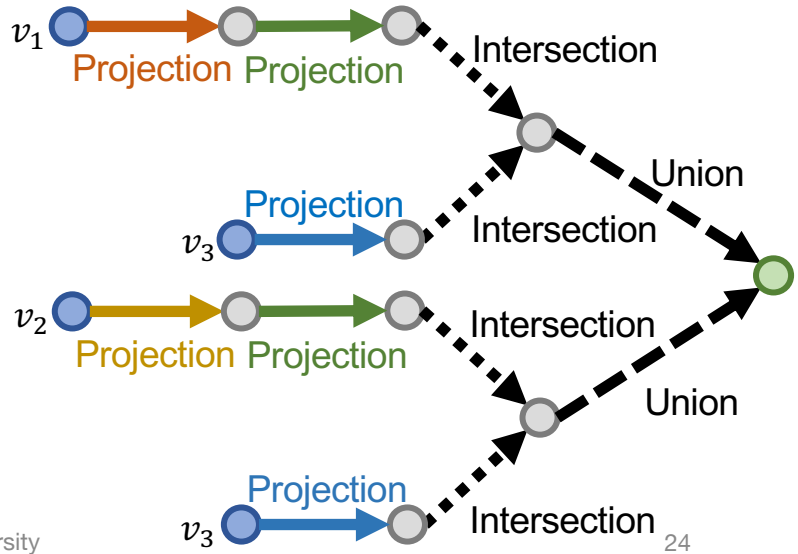
Disjunctive Normal Form

- Any query with AND and OR can be transformed into equivalent Disjunctive Normal Form (disjunction of conjunctive queries).

Original Computation Graph



Converted Computation Graph



Disjunctions: Solution

Given an arbitrary AND-OR query

- 1) Transform it into an DNF
- 2) Answer each conjunctive query
- 3) Overall answer is the union of conjunctive query answers

Benefits of **Query2Box**

Scalability and efficiency:

- Any query can be reduced to a couple of matrix operations and a single k-nearest neighbor search

Generality:

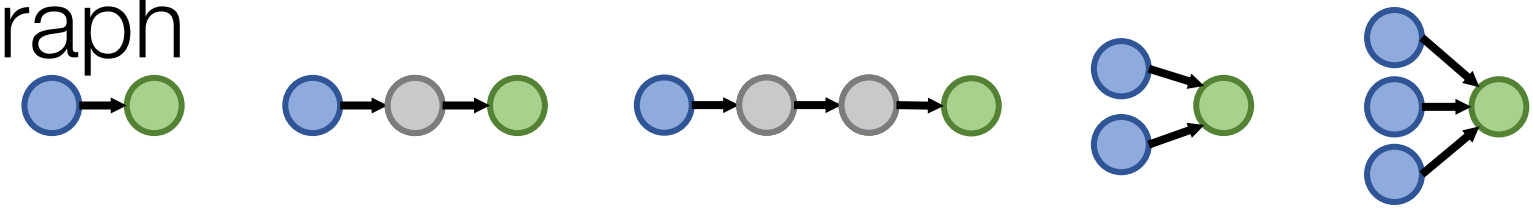
- We can answer any query (even those we have never seen before)

Robustness to noise:

- Graph can contain missing and noisy relationships

Query2Box : Model Training

Training examples: Queries on the graph



- Positives: Path with a known answer
- Negatives: Random nodes of the correct answer type
- **Goal:** Find embeddings and operators so that that queries give correct answers

Experimental Setup

We essentially learn to “memorize” the answers to queries

- We embed entities so that our geometric operators give correct answers

Questions:

- Does our method generalize to new unseen queries?
- Does our method generalize to new query structures?
- Can method handle missing relations?

Experimental Setup

Training on an incomplete graph:

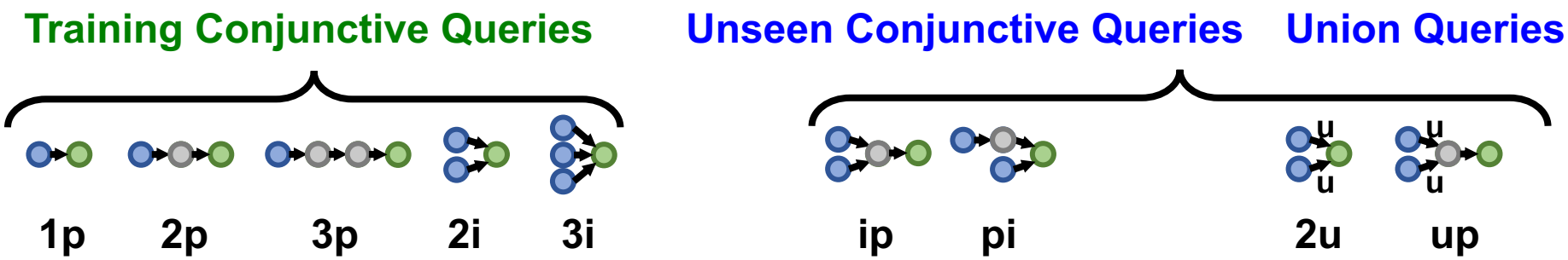
- **Test queries are not answerable in the training graph**
  - Every test query has at least one missing edge
- Method has to (implicitly) input missing edges
  - Note: Query template matching would have accuracy of random guessing

KG and Query Statistics

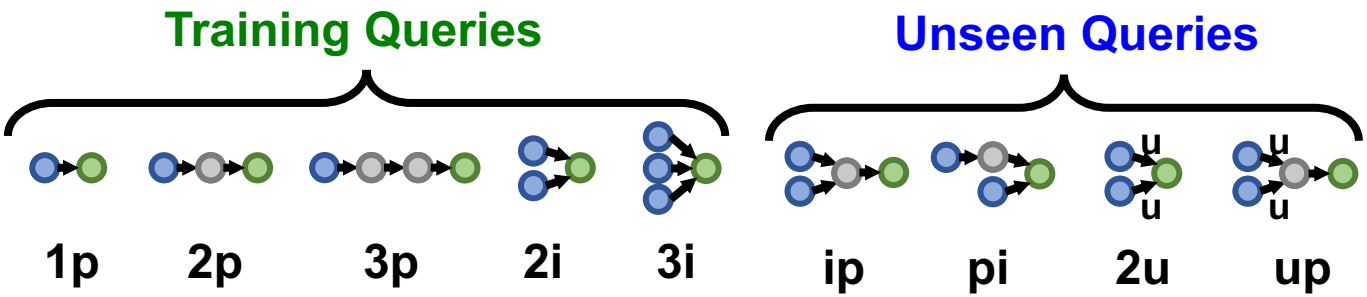
- **Freebase:** FB15K, FB15K-237

Dataset	Entities	Relations	Training Edges	Validation Edges	Test Edges	Total Edges
FB15k	14,951	1,345	483,142	50,000	59,071	592,213
FB15k-237	14,505	237	272,115	17,526	20,438	310,079

- **Queries:**



Experiments Freebase



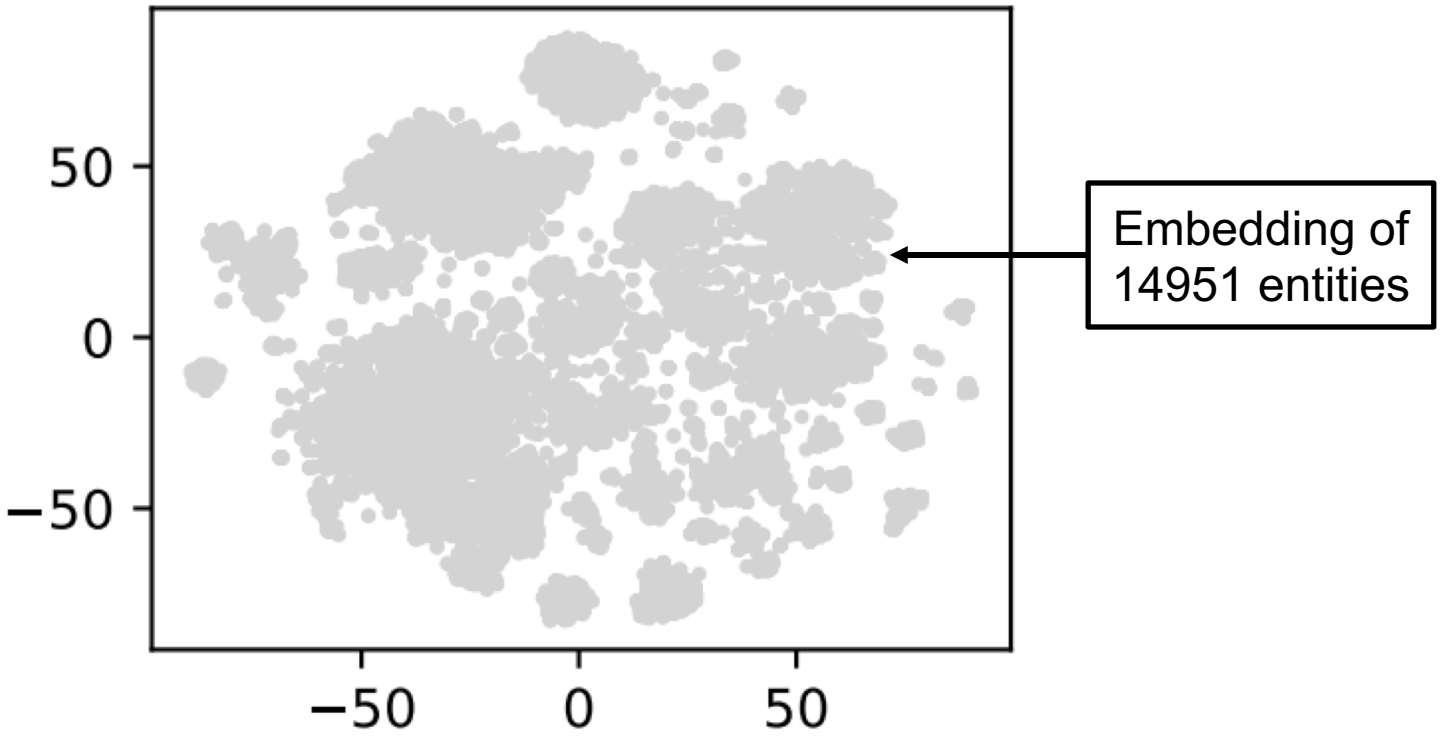
Method	Avg	1p	2p	3p	2i	3i	ip	pi	2u	up
Q2B	**0.484**	**0.786**	**0.413**	**0.303**	**0.593**	**0.712**	**0.211**	**0.397**	**0.608**	**0.330**
Point-based	0.386	0.636	0.345	0.248	0.515	0.624	0.151	0.31	0.376	0.273
embedding	0.384	0.63	0.346	0.250	0.515	0.611	0.153	0.32	0.362	0.271

Table 4: H@3 on test set for QUERY2BOX vs. GQE on FB15k.

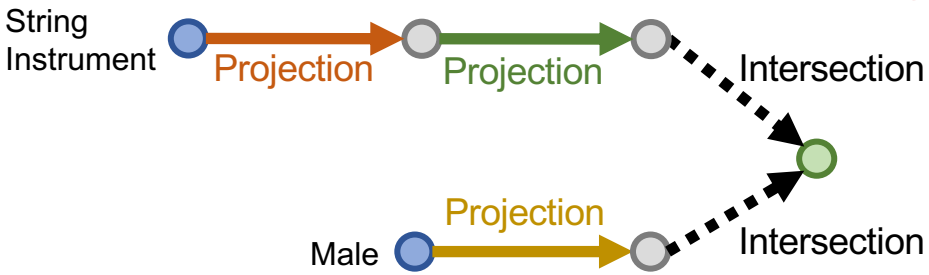
Observations:

- On "training" queries: +20% H@3
- On new conjunctive query structures: +15%
- On disjunctive queries: +36%

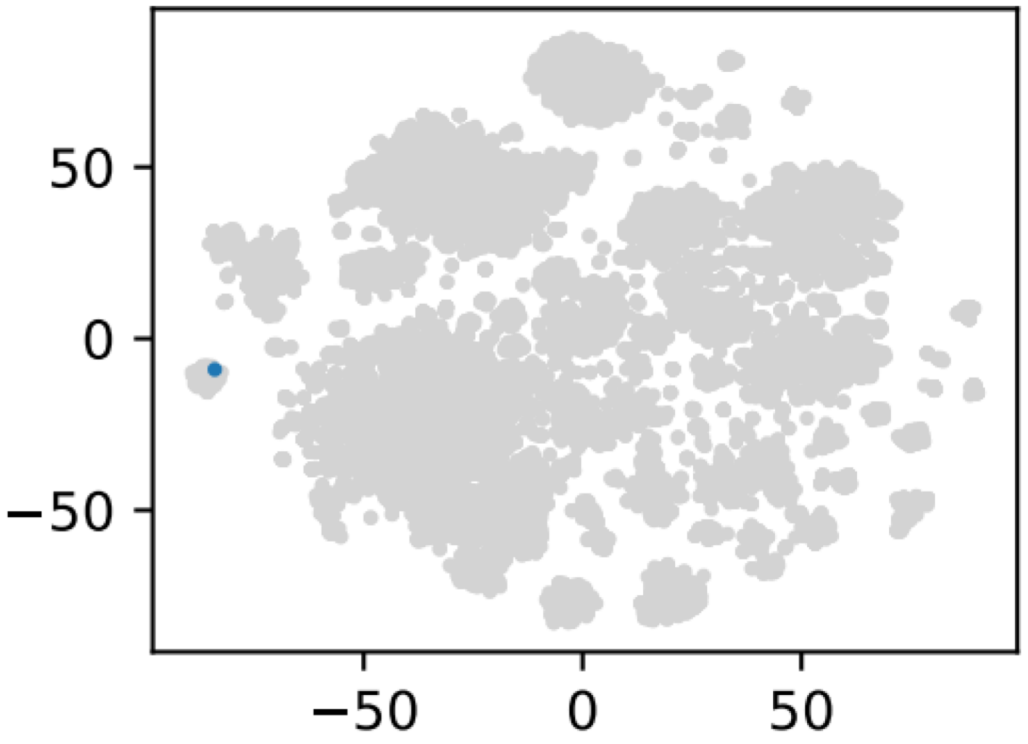
FB15k: Embedding Space



"List male instrumentalists who play string instruments"



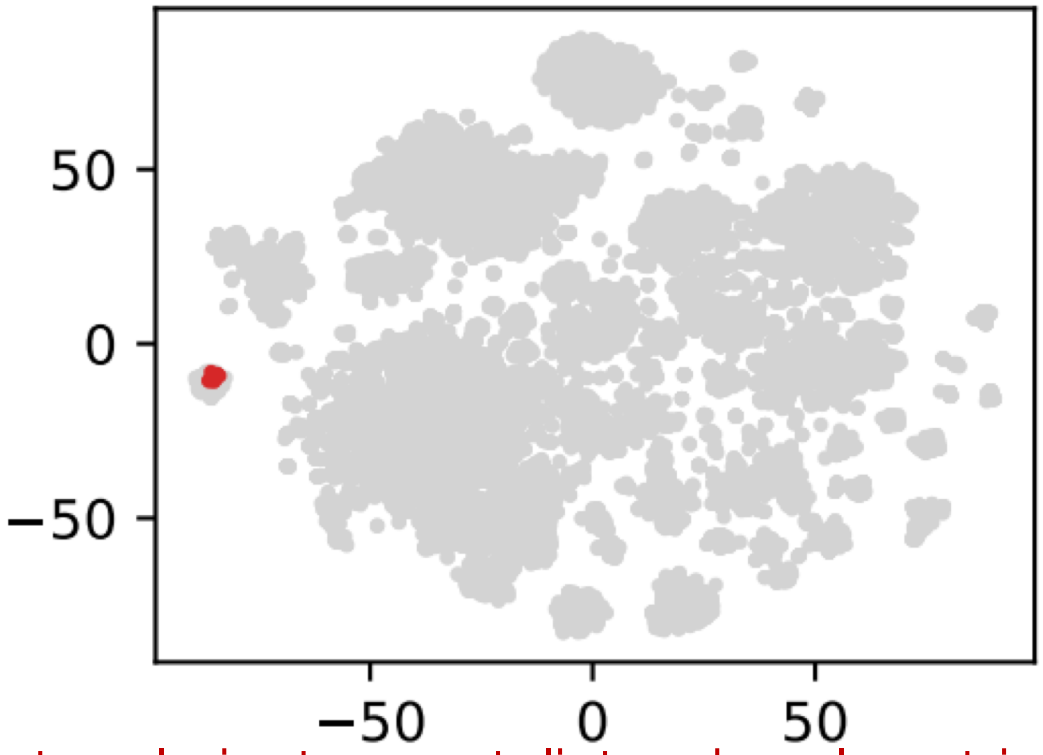
FB15k: Embedding Space



"List male instrumentalists who play string instruments"

String Instrument

FB15k: Embedding Space



# of string instruments: 10

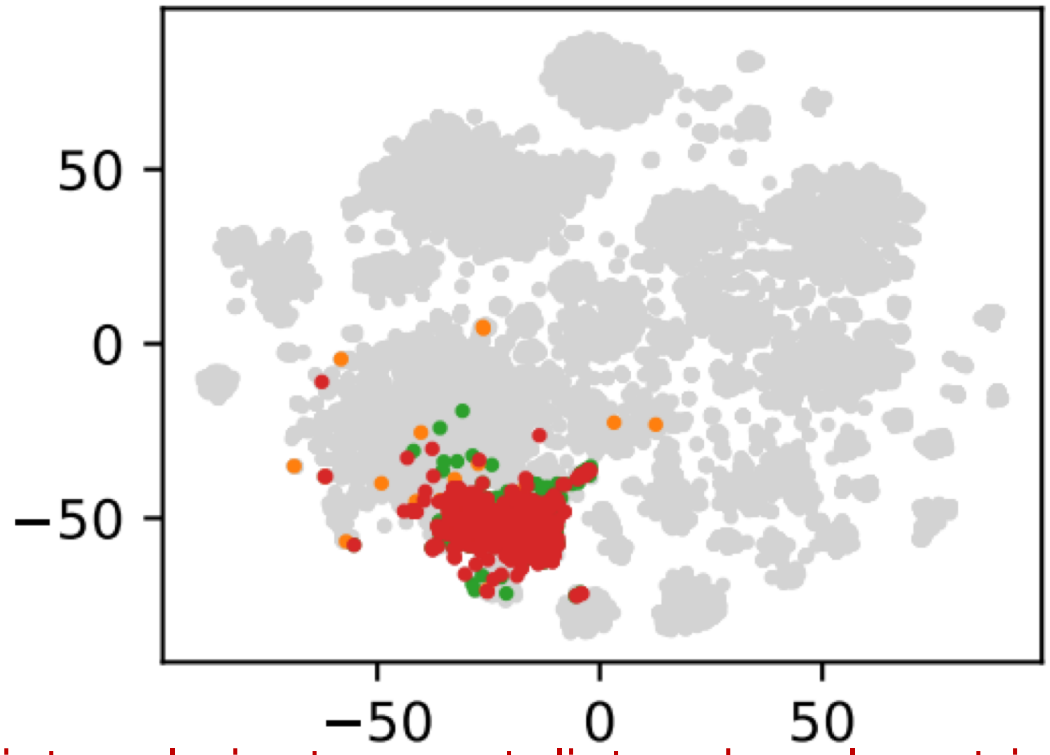
TP
FN
FP
TN

TPR: 100%
FPR: 0%

"List male instrumentalists who play string instruments"

String Instrument → Projection

FB15k: Embedding Space



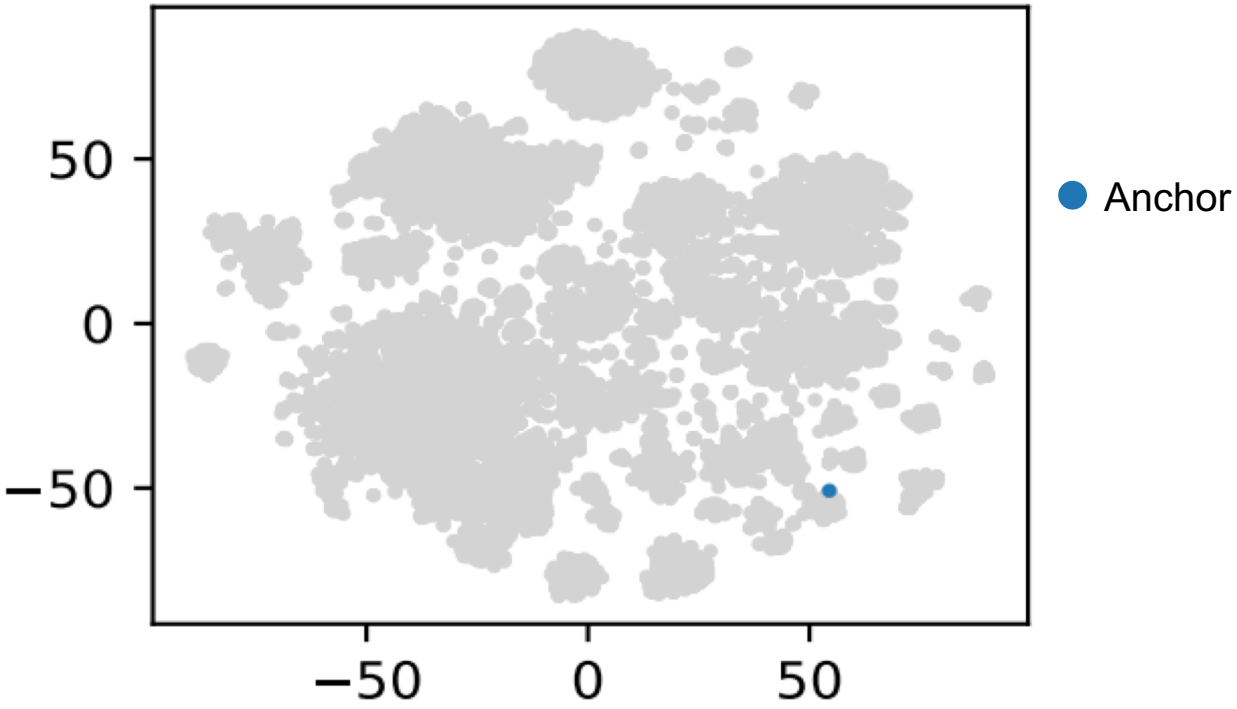
# of instrumentalists: 472

TP
FN
FP
TN

TPR: 98.4%
FPR: 0.01%

"List male instrumentalists who play string instruments"

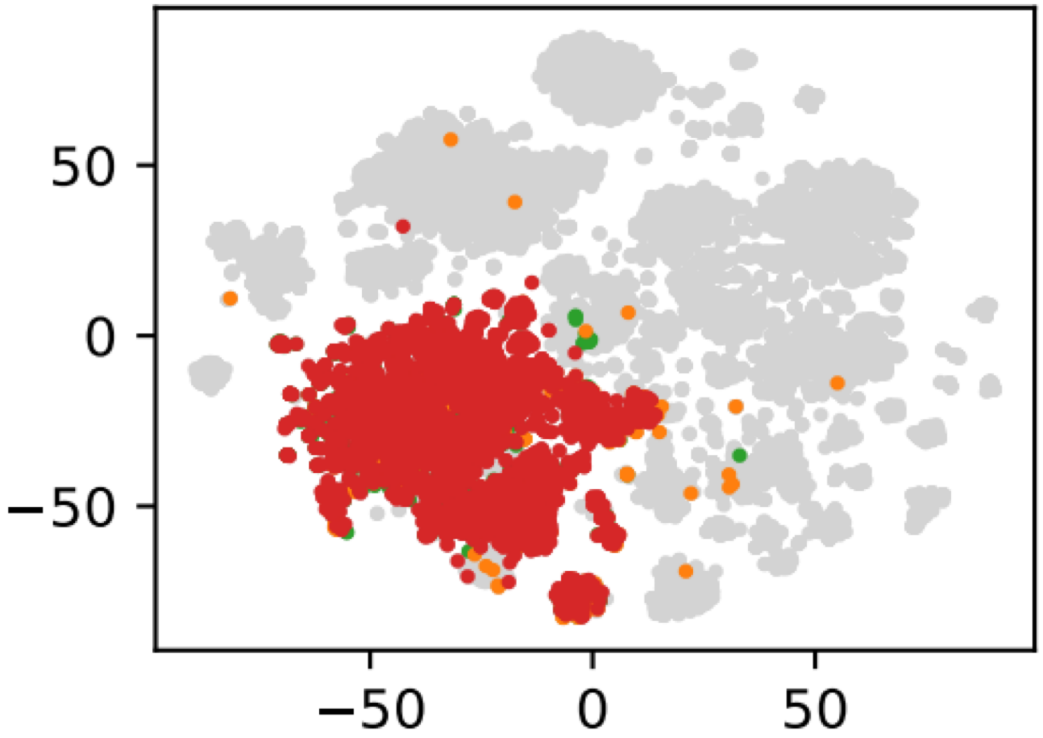
FB15k: Embedding Space



"List male instrumentalists who play string instruments"

Male

FB15k: Embedding Space



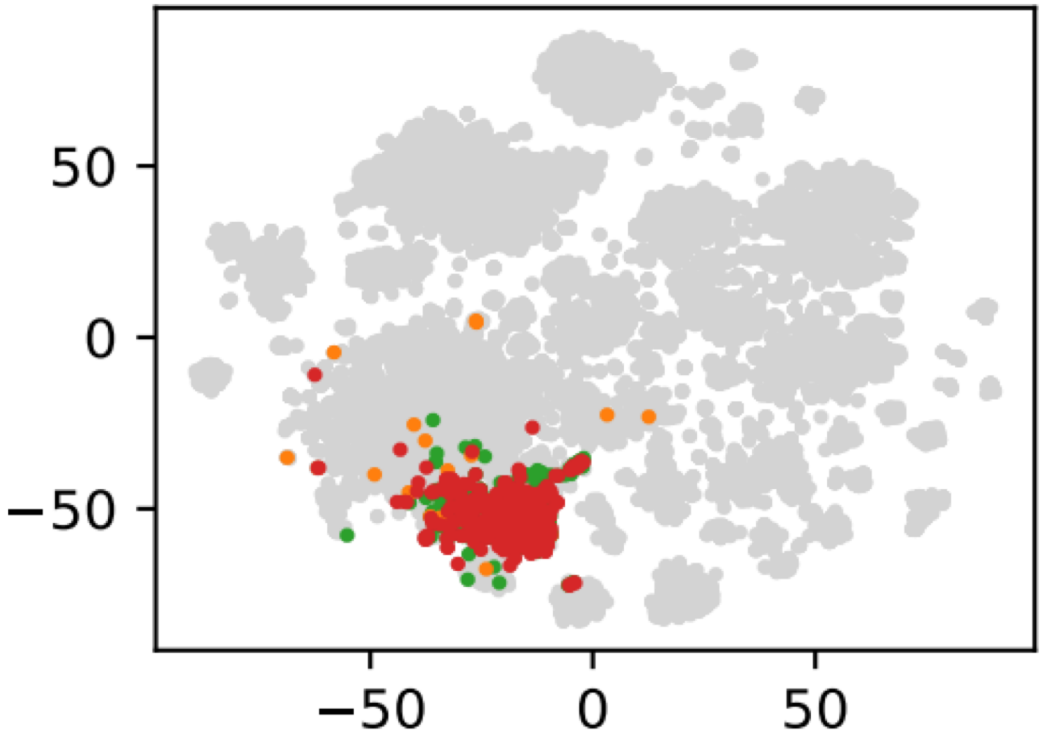
# of men: 3555

TP
FN
FP
TN

TPR: 97.9%
FPR: 0.01%

"List male instrumentalists who play string instruments"

FB15k: Embedding Space

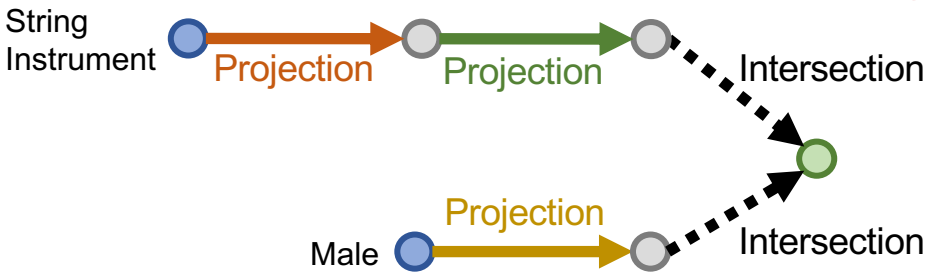


# of answers: 396

TP
FN
FP
TN

TPR: 99.4%
FPR: 0.01%

"List male instrumentalists who play string instruments"



Query2Box: Summary

- **Query2Box:**
  - Embed the query as a box
  - Logical operations become spatial operations
- Composability of queries:
  - Generalize well to unseen, extrapolated queries
  - Explicitly training for composability is important
- Instance vs. multi-hop generalization

Conclusion

- Box embeddings for answering logical queries on Knowledge graphs
- Handle union and intersection
- Generalize well to unseen, extrapolated queries
- Future work: Handle negation, other geometric model

Open Graph Benchmark

- On-going effort for large-scale realistic benchmark datasets for graph ML.

Webpage: https://ogb.stanford.edu/
Github: https://github.com/snap-stanford/ogb
Paper: https://arxiv.org/abs/2005.00687

Why a New Benchmark?

1) **Current focus is on small graphs or small sets of graphs from just a handful of domains:**
   - Datasets are too small:
     - MUTAG graph classification is just 188 graphs
   - Hard to reliably and rigorously evaluate algorithms

2) **Lack of common benchmark datasets for comparing different methods:**
   - Every paper design its own, custom train/test splits
   - Performance across papers is not comparable

3) **Dataset splits follow conventional random splits:**
   - Unrealistic for real-world applications
   - Accuracies are over-optimistic under conventional splits

ML with Graphs

To properly track progress and identify issues with current approaches it is critical for our community to...

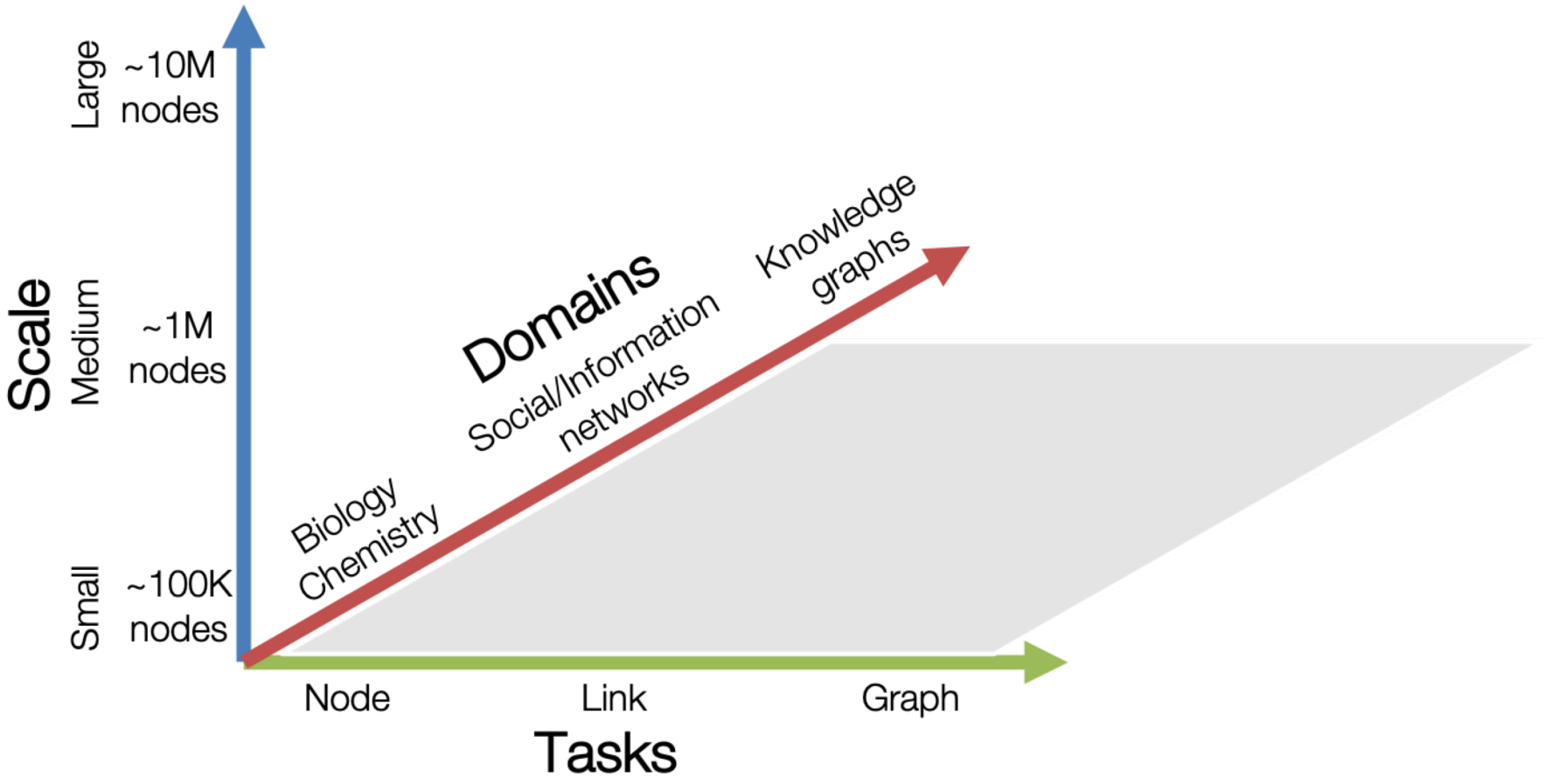
…develop diverse, challenging, and realistic benchmark datasets for machine learning on graphs

The Open Graph Benchmark

OGB is a set of benchmarks for graph ML:

1. **Ready-to-use datasets for key tasks on graphs:**
   - Node classification, link prediction, graph classification
2. **Common codebase to load, construct & represent graphs:**
   - Popular deep frameworks, e.g., DGL, PyTorch Geometric
3. **Common codebase with performance metrics for fast model evaluation and comparison:**
   - **Meaningful data splits focusing on generalization**

OGB Datasets are Diverse



Open Graph Benchmark

https://ogb.stanford.edu

ogb@cs.stanford.edu

Core development team

W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, J. Leskovec

Steering committee

Regina Barzilay, Peter Battaglia, Yoshua Bengio, Michael Bronstein, Stephan Günnemann, Will Hamilton, Tommi Jaakkola, Stefanie Jegelka, Maximilian Nickel, Chris Re, Le Song, Jian Tang, Max Welling, Rich Zemel

PhD Students

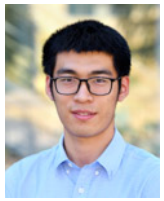

Alexandra Porter


Camilo Ruiz

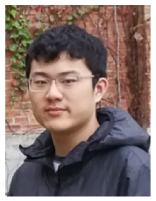

Claire Donnat

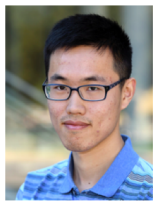

Emma Pierson


Weihua Hu


Jiaxuan You


Bowen Liu

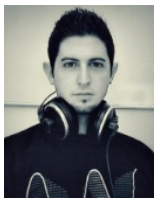

Hongyu Ren

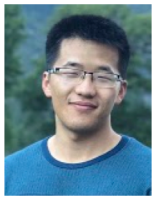

Rex Ying

Post-Doctoral Fellows


Baharan Mirzasoleiman


Marinka Zitnik

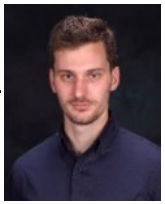

Michele Catasta


Pan Li


Shantao Li


Maria Brbic

Research Staff


Adrijan Bradaschia


Rok Sosic

Industry Partnerships

Funding

Collaborators

Dan Jurafsky, Linguistics, Stanford University
David Grusky, Sociology, Stanford University
Stephen Boyd, Electrical Engineering, Stanford University
David Gleich, Computer Science, Purdue University
VS Subrahmanian, Computer Science, University of Maryland
Marinka Zitnik, Medicine, Harvard University
Russ Altman, Medicine, Stanford University
Jochen Profit, Medicine, Stanford University
Eric Horvitz, Microsoft Research
Jon Kleinberg, Computer Science, Cornell University
Sendhill Mullainathan, Economics, Harvard University
Scott Delp, Bioengineering, Stanford University
James Zou, Medicine, Stanford University

STANFORD INFOLAB

References

- Embedding Logical Queries on Knowledge Graphs. W. Hamilton, P. Bajaj, M. Zitnik, D. Jurafsky, J. Leskovec. *NIPS*, 2018.

- Query2box: Reasoning over Knowledge Graphs in Vector Space Using Box Embeddings. H. Ren, W. Hu, J Leskovec, ICLR 2020.