Adapting Neural Link Predictors for Data-Efficient Complex Query Answering

Pasquale Minervini
School of Informatics, and ELLIS
University of Edinburgh
Edinburgh Scotland
p.minervini@ed.ac.uk

Erik Arakelyan
Department of Computer Science
University of Copenhagen
Copenhagen Denmark
erik.a@di.ku.dk

Daniel Daza
Faculty of Science
Vrije Universiteit Amsterdam
Amsterdam Netherlands
d.dazacruz@vu.nl

Michael Cochez
Learning and Reasoning group
Vrije Universiteit Amsterdam
Amsterdam Netherlands
m.cochez@vu.nl

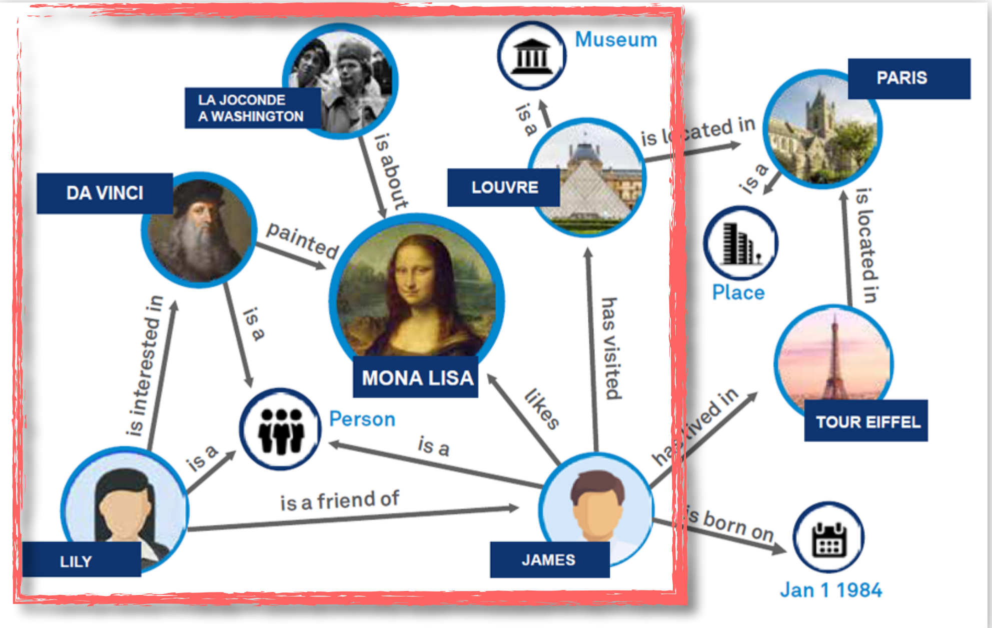
Isabelle Augenstein
Department of Computer Science
University of Copenhagen
Copenhagen Denmark
augenstein@di.ku.dk

About Knowledge Graphs and Complex Query Answering

Knowledge Graphs

- Knowledge Graphs - A Knowledge Graph (KG) is a knowledge base representing the relationships between entities in a relational graph structure

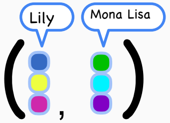
- The flexibility of this knowledge representation formalism allows KGs to be widely used in various domains.

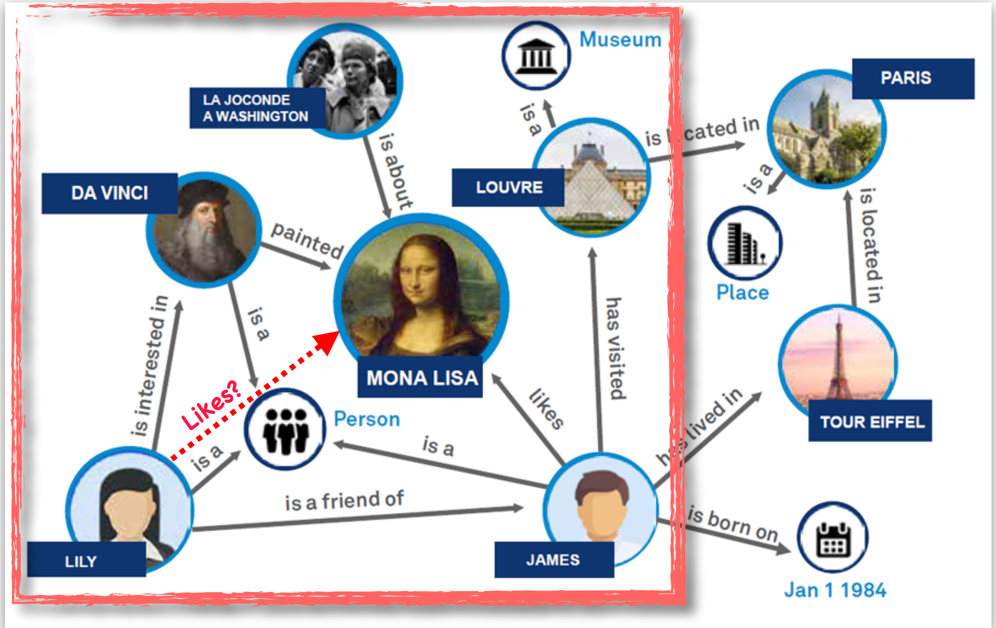


Link Prediction

- What works did L. Da Vinci paint ?
  - **Can be answered with traversal !**

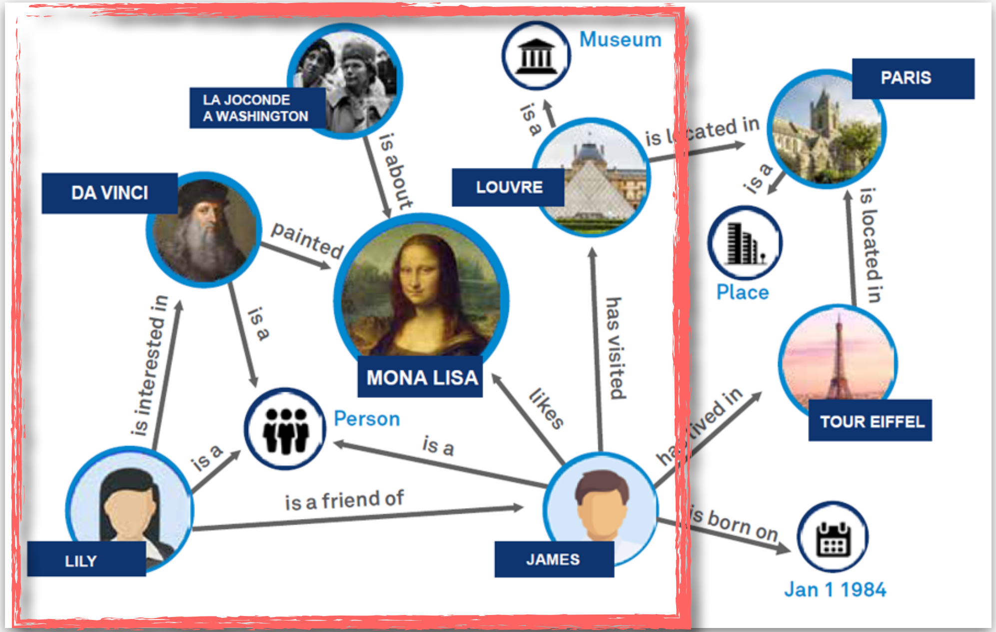
- What does Lily like ?
  - Cannot be directly answer with traversal.
  - Need to use a neural function to predict the probability

$P(Lily \xrightarrow{\text{Likes}} \text{Mona Lisa}) \propto \phi_{\text{Likes}}$ 



Complex Query Answering

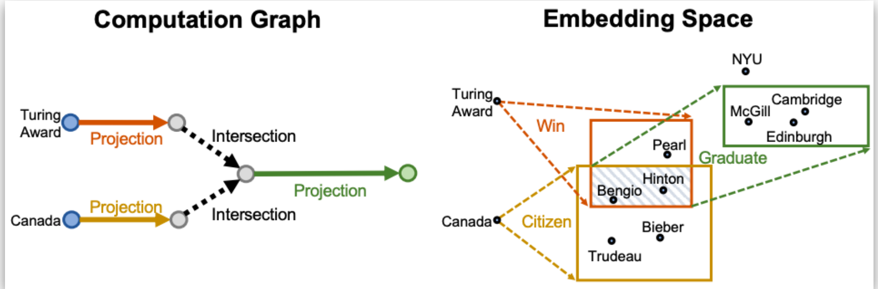
- Complex queries involve answering multi-hop questions that include logical conjunctions ($\land$), disjunctions ($\lor$) and negations ($\neg$)
- Which people are likely to have visited the Louvre given that they are interested in Da Vinci or like Mona Lisa ?
- Which German person produced the music for the film Constantine?

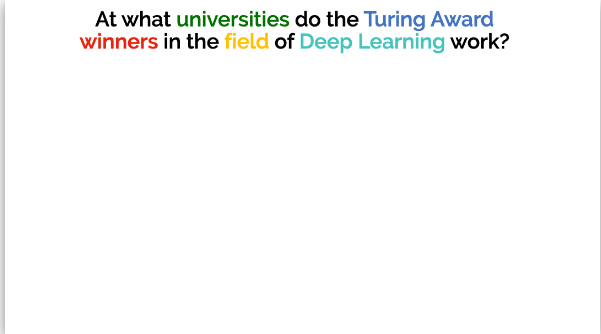


How is this Done ?

- Query Embedding based methods.
  - GQE - Hamilton et al. [NeurIPS 2018]
  - Query2Box - Ren et al. [NeurIPS 2020]
  - BetaE -Ren et al. [ICLR 2020]
  - etc.

- Symbolic and hybrid methods
  - ***CQD - Arakelyan et al. [ICLR 2020]***
  - GNN-QE - Zhu et al. [ICML 2022]
  - EmQL - Sun et al. [NeurIPS 2020]
  - Etc.





Zhu et al. [ICML 2022]

Query Answering as Optimization

Proposed solution: train a neural model ϕ for answering atomic (simple) queries (e.g. "which people are German?"), and cast the query answering task as an *optimization problem*

$$\mathcal{Q} \equiv ?T : \text{Country}(\text{Germany}, T) \wedge \text{music}(\text{Constantine}, T)$$

Query Answering as Optimisation

Proposed solution: train a neural model ϕ for answering atomic (simple) queries (e.g. "which people are German?"), and cast the query answering task as an *optimization problem*

$$\mathcal{Q} \equiv ?T : \text{Country}(\text{Germany}, T) \wedge \text{music}(\text{Constantine}, T)$$

Optimisation problem

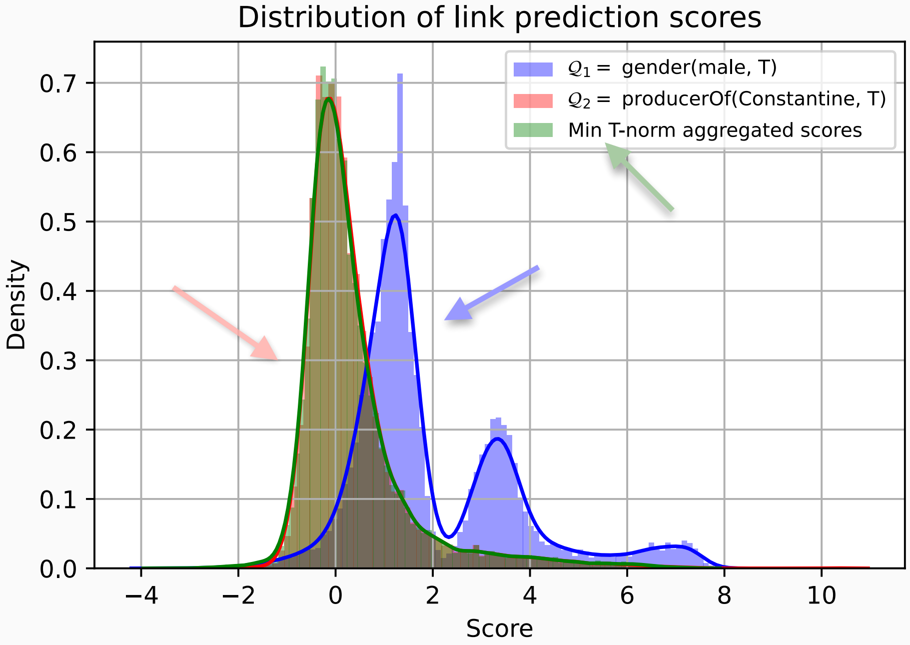
$$\arg\max_{T \in \mathcal{E}}$$

Query Answering as Optimisation

$$\mathcal{Q} \equiv ?T : \text{Country}(\text{Germany}, T) \wedge \text{music}(\text{Constantine}, T)$$

$$\underbrace{\arg\max_{T \in \mathcal{E}}}_{\text{Optimisation problem}} \left[\underbrace{\phi_{\text{Country}}\left(\mathbf{e}_T, \mathbf{e}_G\right)}_{\text{Likelihood that T is German}} \; \underbrace{\top}_{\text{Continuous relaxation of the logical AND (t-norm)}} \; \underbrace{\phi_{\text{music}}\left(\mathbf{e}_C, \mathbf{e}_T\right)}_{\text{Likelihood that T composed from the film Constantine}} \right]$$

Limitations



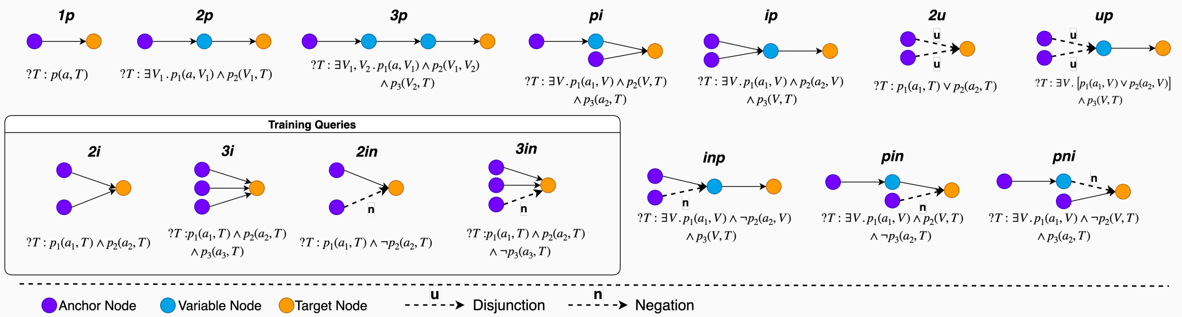
- Limitations
  - The neural link predictor used is not explicitly optimised for the complex query answering task, implying that its scores are not calibrated to interact together
  - Logical negations are not supported

CQD$^{\mathscr{A}}$

To overcome this limitations we propose CQD$^{\mathscr{A}}$, a parameter-efficient score adaptation model optimized to re-calibrate neural link predictor scores for complex query answering task.

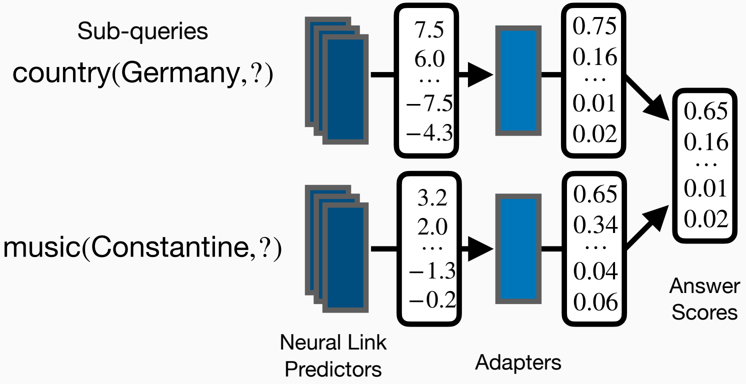
We evaluate the method on an existing benchmark of *3* Knowledge Graphs covering diverse domains while also analyzing the method in terms of generalization, data and parameter efficiency.

In our experiments, CQD$^{\mathscr{A}}$ produces more accurate results than current state-of-the-art methods, improving from 34.4 to 35.1 Mean Reciprocal Rank values averaged across all datasets and query types while using $\leq 30\,\%$ of the available training query types.



Which people are German and produced the music for the film Constantine?

$$\mathscr{Q} \equiv ?T : \text{Country}(\text{Germany}, T) \wedge \text{music}(\text{Constantine}, T)$$



Calibration of atomic scores

- Processes
  - Transformation is ***monotonic***, which is desirable for maintaining the capability to interpret intermediate scores
  - This is parameter efficient !

Optimisation problem

Continuous relaxation of the logical AND (t-norm)

$$\arg\max_{T \in \mathscr{E}} \left[\phi_{\text{Country}}\left(\mathbf{e}_T, \mathbf{e}_G\right) \top \phi_{\text{music}}\left(\mathbf{e}_C, \mathbf{e}_T\right) \right]$$

$$\arg\max_{T \in \mathscr{E}} \left[\rho_\theta\left(\phi_{\text{Country}}\left(\mathbf{e}_T, \mathbf{e}_G\right)\right) \top \rho_\theta\left(\phi_{\text{music}}\left(\mathbf{e}_C, \mathbf{e}_T\right)\right) \right]$$

Affine adaptation function
$\rho_\theta(\phi_p(\mathbf{e}_V, \mathbf{e}_{V'})) = \phi_p(\mathbf{e}_V, \mathbf{e}_{V'})(1 + \alpha) + \beta\,.$

The parameters θ can be conditioned on the representation of the predicate p and the entities V and V', i.e. $\theta = \psi(\mathbf{e}_V, \mathbf{e}_p, \mathbf{e}_{V'})$

Training

- For training the score calibration component in, we first compute how likely each entity $a' \in \mathcal{E}$ is to be an answer to the query $\mathcal{Q}$.

$$\underbrace{\text{score}(\mathcal{Q}, A \leftarrow a')}_{\text{Query Answer assignment}} = \underbrace{\max_{S} \text{score}(\mathcal{Q}, S)}_{\text{Optimization Task}}, \text{ where } A \leftarrow a' \in S\,.$$

- We optimize the adaptation parameters by gradient descent on the likelihood of the true answers on a dataset of query-answer pairs by using a ***1-vs-all cross-entropy loss***

$$\mathcal{L}(\mathcal{D}) = \sum_{(\mathcal{Q}_i, a_i) \in \mathcal{D}} - \text{score}(\mathcal{Q}_i, A \leftarrow a_i) + \log\left[\sum_{a' \in \mathcal{E}} \exp\left(\text{score}(\mathcal{Q}_i, A \leftarrow a')\right)\right].$$

COPENLU NATURAL LANGUAGE UNDERSTANDING RESEARCH GROUP

THE UNIVERSITY of EDINBURGH

VU

Results

Model	avg_p	avg_n	1p	2p	3p	2i	3i	pi	ip	2u	up	2in	3in	inp	pin	pni
								FB15K								
GQE	28.0	-	54.6	15.3	10.8	39.7	51.4	27.6	19.1	22.1	11.6	-	-	-	-	-
Q2B	38.0	-	68.0	21.0	14.2	55.1	66.5	39.4	26.1	35.1	16.7	-	-	-	-	-
BetaE	41.6	11.8	65.1	25.7	24.7	55.8	66.5	43.9	28.1	40.1	25.2	14.3	14.7	11.5	6.5	12.4
CQD-CO	46.9	-	**89.2**	25.3	13.4	74.4	78.3	44.1	33.2	41.8	21.9	-	-	-	-	-
CQD-Beam	68.4	-	**89.2**	65.3	29.7	76.1	79.3	70.6	70.6	72.3	59.4	-	-	-	-	-
ConE	49.8	14.8	73.3	33.8	29.2	64.4	73.7	50.9	35.7	55.7	31.4	17.9	18.7	12.5	9.8	15.1
GNN-QE	72.8	38.6	88.5	**69.3**	**58.7**	**79.7**	**83.5**	69.9	70.4	**74.1**	**61.0**	44.7	41.7	**42.0**	30.1	34.3
CQDA	70.4	42.8	**89.2**	64.5	57.9	76.1	79.4	**70.0**	**70.6**	68.4	57.9	**54.7**	**47.1**	37.6	**35.3**	24.6

Results

	FB15K-237															
GQE	16.3	-	35.0	7.2	5.3	23.3	34.6	16.5	10.7	8.2	5.7	-	-	-	-	-
Q2B	20.1	-	40.6	9.4	6.8	29.5	42.3	21.2	12.6	11.3	7.6	-	-	-	-	-
BetaE	20.9	5.5	39.0	10.9	10.0	28.8	42.5	22.4	12.6	12.4	9.7	5.1	7.9	7.4	3.5	3.4
CQD-CO	21.8	-	**46.7**	9.5	6.3	31.2	40.6	23.6	16.0	14.5	8.2	-	-	-	-	-
CQD-Beam	25.3	-	**46.7**	13.3	7.9	34.4	48.3	27.1	20.4	17.6	11.5	-	-	-	-	-
ConE	23.4	5.9	41.8	12.8	11.0	32.6	47.3	25.5	14.0	14.5	10.8	5.4	8.6	7.8	4.0	3.6
GNN-QE	26.8	10.2	42.8	**14.7**	**11.8**	**38.3**	**54.1**	**31.1**	18.9	16.2	**13.4**	10.0	**16.8**	**9.3**	7.2	**7.8**
CQDA	25.7	10.7	**46.7**	13.6	11.4	34.5	48.3	27.4	**20.9**	**17.6**	11.4	**13.6**	16.8	7.9	**8.9**	5.8

Results

	NELL995															
GQE	18.6	-	32.8	11.9	9.6	27.5	35.2	18.4	14.4	8.5	8.8	-	-	-	-	-
Q2B	22.9	-	42.2	14.0	11.2	33.3	44.5	22.4	16.8	11.3	10.3	-	-	-	-	-
BetaE	24.6	5.9	53.0	13.0	11.4	37.6	47.5	24.1	14.3	12.2	8.5	5.1	7.8	10.0	3.1	3.5
CQD-CO	28.8	-	**60.4**	17.8	12.7	39.3	46.6	30.1	22.0	17.3	13.2	-	-	-	-	-
CQD-Beam	31.8	-	**60.4**	22.6	13.6	42.6	52.0	31.2	25.6	19.9	16.7	-	-	-	-	-
ConE	27.2	6.4	53.1	16.1	13.9	40.0	50.8	26.3	17.5	15.3	11.3	5.7	8.1	10.8	3.5	3.9
GNN-QE	28.9	9.7	53.3	18.9	14.9	42.4	52.5	30.8	18.9	15.9	12.6	9.9	14.6	11.4	6.3	6.3
CQDA	32.3	13.3	**60.4**	**22.9**	**16.7**	**43.4**	**52.6**	**32.1**	**26.4**	**20.0**	**17.0**	**15.1**	**18.6**	**15.8**	**10.7**	**6.5**

THE UNIVERSITY of EDINBURGH

Model Modes

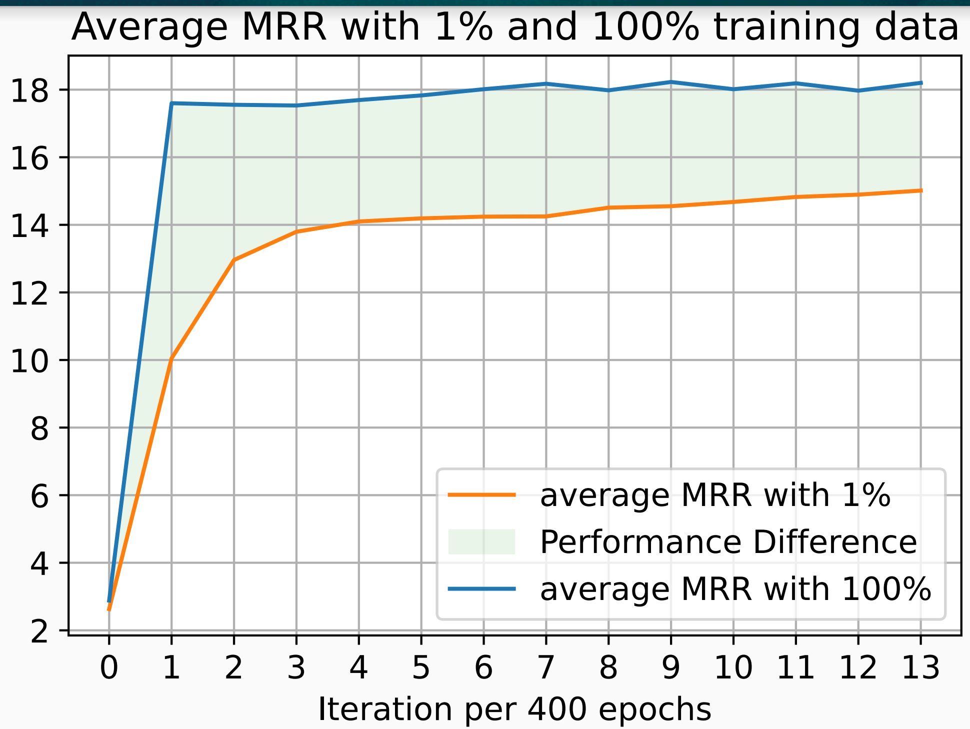
Model	**2p**	**2i**	3i	pi	ip	2u	up	2in	3in	inp	pin	pni
CQD	13.2	34.5	48.2	26.8	20.3	17.4	10.3	5.4	12.4	6.1	3.2	4.6
CQD_F	9.3	22.8	34.9	19.8	14.5	13.0	7.2	7.4	7.1	4.9	3.9	3.8
CQD^A_F	9.5	23.9	39.0	19.8	14.5	14.2	7.2	8.4	9.7	4.9	4.2	3.6
CQD_C	10.9	33.7	47.3	25.6	18.9	16.4	9.4	7.9	12.2	6.6	4.2	5.0
CQD_R	6.4	22.2	31.0	16.6	11.2	12.5	4.8	4.7	5.9	4.1	2.0	3.5
CQDA	13.2	35.0	48.5	27.3	20.7	17.6	10.5	13.2	14.9	7.4	7.8	5.5

Test MRR results for FOL queries on FB15K-237 using the following CQD extensions:CQD from Arakelyan et al. with the considered normalisation and negations; CQD_F, where we fine-tune ***all neural link predictor parameters*** in CQD; CQD^A_F, where we ***fine-tune all link predictor parameters*** in CQDA; CQD_R, where we learn a ***transformation*** for the entity and relation embeddings and we use it to ***replace*** the initial entity and relation representations; and CQD_C, where we learn a transformation for the entity and relation embeddings, and we ***concatenate*** it to the initial entity and relation representations.

Data Efficiency

Dataset	**Model**	**1p**	2p	3p	2i	3i	pi	ip	2u	up	2in	3in	inp	pin	pni
FB237, 1%	CQDA	***46.7***	***11.8***	***11.4***	***33.6***	41.2	***24.82***	***17.81***	***16.45***	***8.74***	***10.8***	***13.86***	5.93	***5.38***	***14.82***
	GNN-QE	36.82	8.96	8.13	33.02	***49.28***	24.58	14.18	10.73	8.47	4.89	12.31	6.74	4.41	4.09
	BetaE	36.80	6.89	5.94	22.84	34.34	17.12	8.72	9.23	5.66	4.44	6.14	5.18	2.54	2.94
FB237 2i, 1%	CQDA	***46.7***	***11.8***	***11.2***	***30.35***	40.75	***23.36***	***18.28***	***15.85***	***8.96***	***9.36***	***10.25***	***5.17***	***4.46***	***4.44***
	GNN-QE	34.81	5.40	5.17	30.12	***48.88***	23.06	12.65	9.85	5.26	4.26	12.5	4.43	0.71	1.98
	BetaE	37.99	5.62	4.48	23.73	35.25	15.63	7.96	9.73	4.56	0.15	0.49	0.62	0.10	0.14

Data Efficiency



- Average test MRR score (y-axis) of CQDA using ***1% and 100%*** of the training queries from FB15K-237 throughout the training iterations (x-axis).

Summary

- We propose a novel method for Complex Query Answering
- The Method is able to answer complex EPFO queries
- We are able to obtain SOTA results on Complex query answering benchmarks
- The method show great generalization capabilities
    - data and parameter efficiency

Thanks!

Paper: Arxiv

Chat with the paper

Codebase

Contact us:

Erik Arakelyan
Dr. Pasquale Minervini

Questions ?