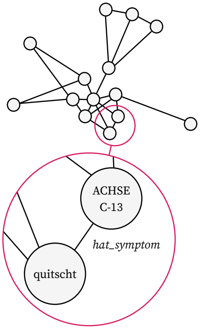
LAVIS
RheinMain University of Applied Sciences

Inductive Linking and Ranking in Knowledge Graphs of Varying Scale

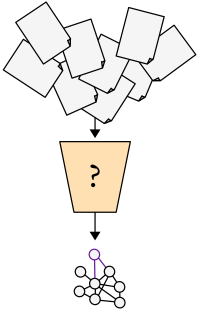
Workshop on Text Mining and Generation

September 19, 2022

- Topic: information extraction
- Discover and extract
  - **entities** and their **relations**
  - from **natural text**
- Possible sources: Issue tracking systems, insurance claims, customer inquiries, . . .
- Our *industrial* reality:
  - Unstructured text in abundance
  - Scarce or no structured data

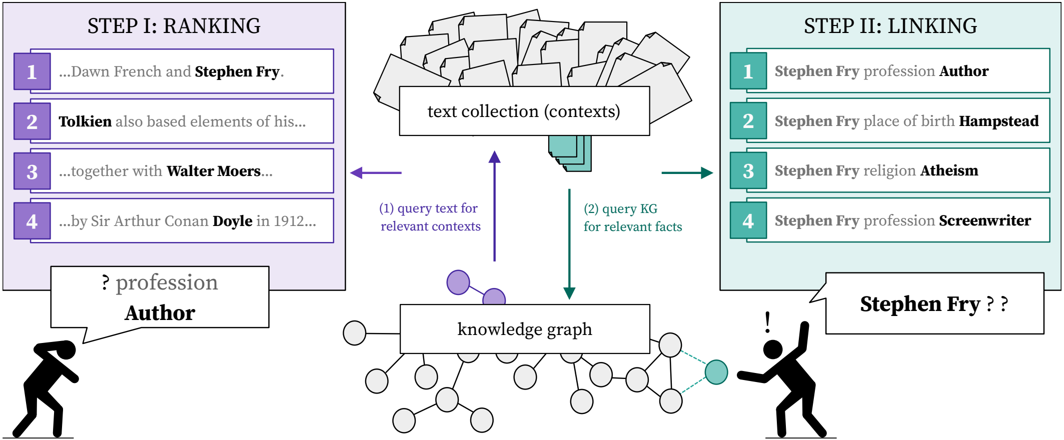


- Constraints and tools
  - **Scarce** graph data
  - **Noisy**, inconcise text
  - **Generic** knowledge not tailored to domain
  - **Neural** machine learning approaches
- It is not possible to try and compare models
  - Industry data needs to be labelled (expensive)
  - Even if labelled: usually confidential
  - Research benchmarks unsuitable [1, 2, 3, 4]
- **Contribution:**
  - A benchmark which reflects our industry use-cases



Benchmark for Ranking and Linking

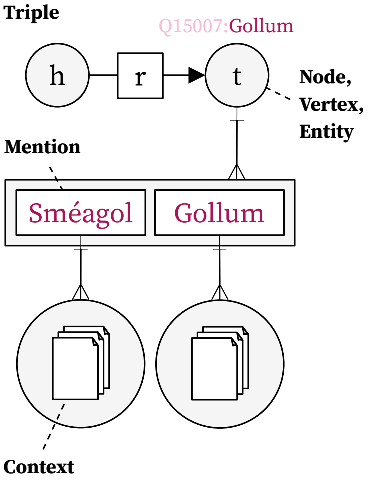
- Introducing Inductive Reasoning with Text (IRT) benchmarks
- Goals to resemble industry situation:
  - Study **graph scarcity** by varying sample size
  - Scattered, inconcise text with **incidental mentions**
  - Unknown entities are assumed to be **volatile**
- Using open data: Wikidata, Freebase, and Wikipedia
- Two versions IRT1 [5] & **IRT2** [6]


STEP I: RANKING
1 ...Dawn French and Stephen Fry.
2 Tolkien also based elements of his...
3 ...together with Walter Moers...
4 ...by Sir Arthur Conan Doyle in 1912...
? profession
Author
text collection (contexts)
(1) query text for relevant contexts
(2) query KG for relevant facts
knowledge graph
STEP II: LINKING
1 Stephen Fry profession Author
2 Stephen Fry place of birth Hampstead
3 Stephen Fry religion Atheism
4 Stephen Fry profession Screenwriter
Stephen Fry ? ?
!

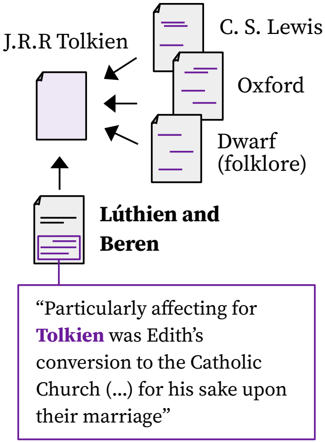
Datamodel

- KG: $\mathcal{G} = (\mathcal{V}, \mathcal{R}, \mathcal{T}, \mathcal{M}, \mathcal{C})$
- $(h, r, t) \in \mathcal{T} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$
- Get a mention: $M : \mathcal{V} \mapsto \mathcal{P}(\mathcal{M})$
- Get contexts: $C : \mathcal{M} \mapsto \mathcal{P}(\mathcal{C})$

$C($GOLLUM$) = \{$ *"In 2014, the Turkish physician Bilgin Çiftçi shared an image comparing Turkish President Recep Tayyip Erdoğan to* GOLLUM*."* $, \ldots \}$



- Goal: Gather mentions and associated text contexts
- We assume a **weak link** between mentions and text
  1. Gather mentions using hyperlink descriptions [7]
  2. Sample sentences from backlinked pages
- J. R. R. TOLKIEN *religion* ?



LAVIS | RheinMain University of Applied Sciences

- Goal: Emulate world knowledge by selecting **concept entities**
- Selection criterium: Disproportion of heads and tails

$$ratio(r) = \frac{\min\left(\mathrm{dom}(r), \mathrm{rg}(r)\right)}{\max\left(\mathrm{dom}(r), \mathrm{rg}(r)\right)}$$

Relation	Ratio	Heads	Tails
Language	0.006	9,816	62
Occupation	0.02	13,145	375
Influenced by	0.8	514	590
Spouse	1	804	804

Relation sub-selection taken from the CodEx-M benchmark [8] to construct IRT1-CDE and IRT2

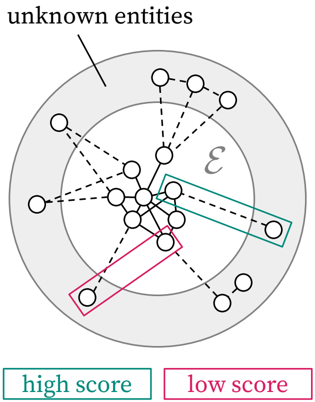
Graph Scarcity

- Goal: Study model performance for scarce graphs
- Four (limited) views on the same data
- Hand-selected subsets of upstream dataset

	Tiny	**Small**	**Medium**	**Large**
Relations	5	12	45	45
Entities	1,174	2,887	3,592	9,952
Training Triples	2,928	7,527	26,335	102,289
Training Contexts	9m	15m	17m	18m

Inductive Knowledge Graph Completion

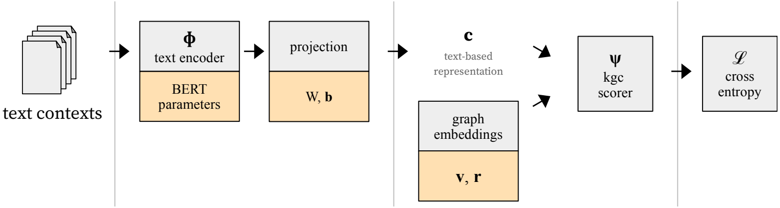
- Given a KG with (h, r, t) triples
- Predict $(q, r, ?)$ and $(?, r, q)$
- Ranking by scoring all possible triples
- Transductive scenario: “classic” KGC
- Inductive scenario:
  - Query entity $q \notin \mathcal{E}$
  - Auxiliary information is text: $C(M(q))$



- Goal: Predict **missing links** for **unknown entities**
- For modern neural approaches:
  - Train graph embeddings $\mathbf{v}, \mathbf{r}$ and text representations $\mathbf{c}$
    $\mathbf{c}, \mathbf{r}, \mathbf{v} \in \mathbb{C}^d$
  - Combine a neural link prediction model ψ
    $s(h, r, t) = \psi(\mathbf{v}_h, \mathbf{r}, \mathbf{v}_t)$
    SOTA: triple scorer or GNNs [9, 10, 11]
  - With a neural text encoder ϕ
    $\mathbf{c} = \phi(c), c \in \mathcal{C}$
    SOTA: large, pre-trained attention models [12, 13, 14]

- Key idea: Use text representation in the graph embedding space

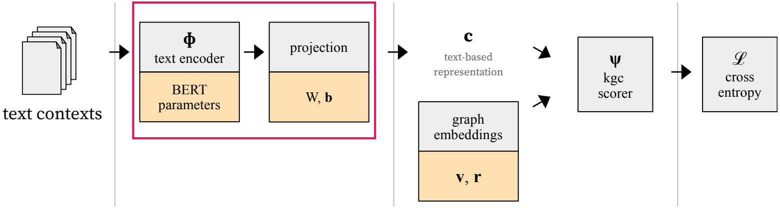
$$s(q,r,t) = \psi\big(\underbrace{W\phi(c_q)_{\text{CLS}} + \mathbf{b}}_{\mathbf{c}}, \mathbf{r}, \mathbf{v}_t\big), c_q \in C(M(q))$$



$\psi(\phi(c_q)_{\text{CLS}}, \mathbf{r}, \phi(c_t)_{\text{CLS}})$ also possible [1], but not studied here

- Key idea: Use text representation in the graph embedding space

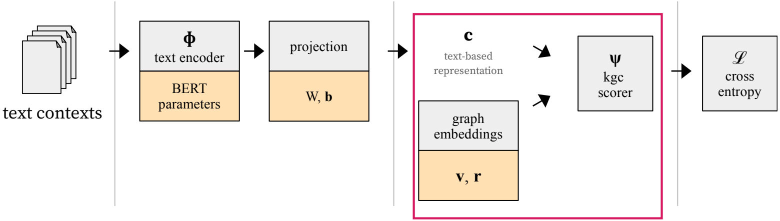
$$s(q, r, t) = \psi\left(\boxed{\underbrace{W\phi(c_q)_{\mathrm{CLS}} + \mathbf{b}}_{\mathbf{c}}}, \mathbf{r}, \mathbf{v}_t\right), c_q \in C(M(q))$$



$\psi(\phi(c_q)_{\mathrm{CLS}}, \mathbf{r}, \phi(c_t)_{\mathrm{CLS}})$ also possible [1], but not studied here

- Key idea: Use text representation in the graph embedding space

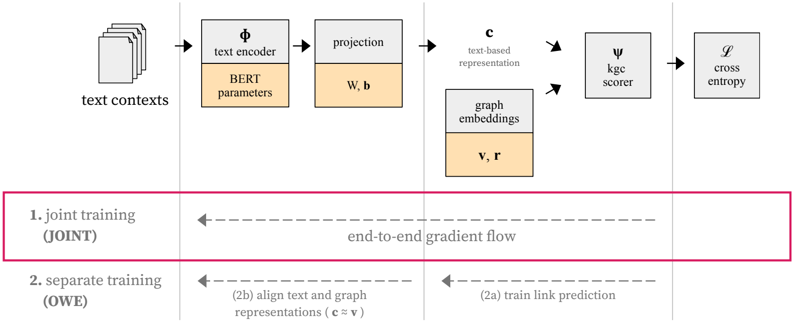
$$s(q,r,t) = \boxed{\psi(\mathbf{c}, \mathbf{r}, \mathbf{v}_t)}, c_q \in C(M(q))$$



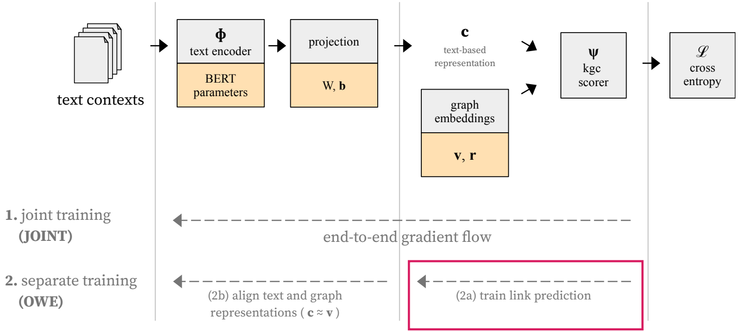
$\psi(\phi(c_q)_{\text{CLS}}, \mathbf{r}, \phi(c_t)_{\text{CLS}})$ also possible [1], but not studied here

IKGC: JOINT Model

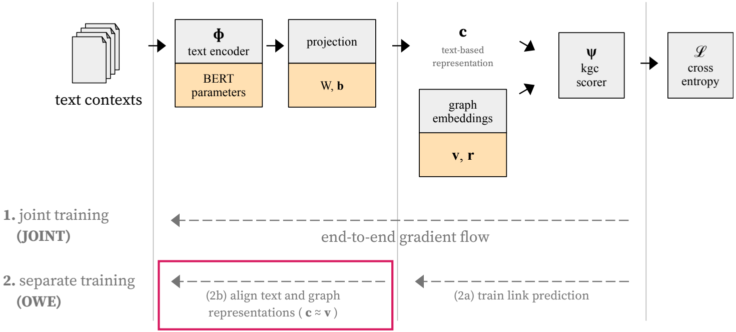
- End-to-end training (JOINT)
- Train closed-world embedding using text
- Cross-entropy loss



- Two-step training (OWE)
- Train reference embedding on known entities
- Mean squared error loss



- Two-step training (OWE)
- Train reference embedding on known entities
- Mean squared error loss



Linking Results using both approaches (Scorer = ComplEx [10])

Given text, predict relations and target entities

		HITS@10				MRR			
Model	**Inst.**	**Tiny**	**Small**	**Med.**	**Large**	**Tiny**	**Small**	**Med.**	**Large**
BOW		53.82	55.18	46.43	**71.38**	33.63	34.62	29.81	**50.61**
JOINT	single	72.06	70.20	47.14	65.75	50.61	45.95	33.72	48.29
JOINT	multi	73.56	74.27	53.77	65.12	51.28	**52.39**	37.50	45.26
OWE	single	74.09	**74.33**	61.98	64.27	50.25	50.57	40.60	42.69
OWE	multi	**75.39**	71.49	**64.41**	66.36	**53.06**	47.17	**43.25**	45.51

Linking Results using both approaches (Scorer = ComplEx [10])

Given text, predict relations and target entities

		HITS@10				MRR			
Model	**Inst.**	**Tiny**	**Small**	**Med.**	**Large**	**Tiny**	**Small**	**Med.**	**Large**
BOW		53.82	55.18	46.43	**71.38**	33.63	34.62	29.81	**50.61**
JOINT	single	72.06	70.20	47.14	65.75	50.61	45.95	33.72	48.29
JOINT	multi	73.56	74.27	53.77	65.12	51.28	**52.39**	37.50	45.26
OWE	single	74.09	**74.33**	61.98	64.27	50.25	50.57	40.60	42.69
OWE	multi	**75.39**	71.49	**64.41**	66.36	**53.06**	47.17	**43.25**	45.51

Linking Results using both approaches (Scorer = ComplEx [10])

Given text, predict relations and target entities

		HITS@10				MRR			
Model	**Inst.**	**Tiny**	**Small**	**Med.**	**Large**	**Tiny**	**Small**	**Med.**	**Large**
BOW		53.82	55.18	46.43	**71.38**	33.63	34.62	29.81	**50.61**
JOINT	single	72.06	70.20	47.14	65.75	50.61	45.95	33.72	48.29
JOINT	multi	73.56	74.27	53.77	65.12	51.28	**52.39**	37.50	45.26
OWE	single	74.09	**74.33**	61.98	64.27	50.25	50.57	40.60	42.69
OWE	multi	**75.39**	71.49	**64.41**	66.36	**53.06**	47.17	**43.25**	45.51

Linking Results using both approaches (Scorer = ComplEx [10])

Given text, predict relations and target entities

		HITS@10				MRR			
Model	**Inst.**	**Tiny**	**Small**	**Med.**	**Large**	**Tiny**	**Small**	**Med.**	**Large**
BOW		53.82	55.18	46.43	**71.38**	33.63	34.62	29.81	**50.61**
JOINT	single	72.06	70.20	47.14	65.75	50.61	45.95	33.72	48.29
JOINT	multi	73.56	74.27	53.77	65.12	51.28	**52.39**	37.50	45.26
OWE	single	74.09	**74.33**	61.98	64.27	50.25	50.57	40.60	42.69
OWE	multi	**75.39**	71.49	**64.41**	66.36	**53.06**	47.17	**43.25**	45.51

IKGC: Ranking Results on IRT2

- Idea: Use IKGC scores for ranking
- Pre-compute scores $s(c, r, t)$ for all t, r and c
- When asked $(?, r, t)$ order c by assigned score

		HITS@100			
		Tiny	**Small**	**Med.**	**Large**
BOW		2.86	4.29	6.42	14.83
JOINT	single	7.91	6.78	6.37	19.47
JOINT	multi	**13.28**	**16.17**	**14.38**	30.68
OWE	single	6.30	8.19	6.88	10.81
OWE	multi	9.98	13.00	6.36	**31.40**

Conclusions

- We present IRT2, a more realistic inductive benchmark
- Linking works well using both neural approaches
  - We recommend OWE as its much less costly
- Ranking promising but not ready for tooling
  - Future work: Learning to rank
- Benchmark and models for download
  - `https://github.com/lavis-nlp/irt2` (Benchmark and evaluation)
  - `https://github.com/lavis-nlp/irt2m` (Models and training)

Thank you!

References

[1] Daniel Daza, Michael Cochez, and Paul Groth. Inductive entity representations from text via link prediction. In *Proceedings of the Web Conference 2021*, pages 798–808, 2021.

[2] Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194, 2021.

[3] Haseeb Shah, Johannes Villmow, Adrian Ulges, Ulrich Schwanecke, and Faisal Shafait. An open-world extension to knowledge graph completion models. In *Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.

[4] Baoxu Shi and Tim Weninger. Open-world knowledge graph completion. *arXiv preprint arXiv:1711.03438*, 2017.

[5] Felix Hamann, Adrian Ulges, Dirk Krechel, and Ralph Bergmann. Open-world knowledge graph completion benchmarks for knowledge discovery. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 252–264. Springer, 2021.

[6] Felix Hamann, Maurice Falk, and Adrian Ulges. Inductive linking and ranking in knowledge graphs of varying scale. 2022.

[7] David Milne and Ian H Witten. Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 509–518, 2008.

[8] Tara Safavi and Danai Koutra. CoDEx: A Comprehensive Knowledge Graph Completion Benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8328–8350, 2020.

[9] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795, 2013.

[10] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, pages 2071–2080, 2016.

[11] Deepak Nathani, Jatin Chauhan, Charu Sharma, and Manohar Kaul. Learning attention-based embeddings for relation prediction in knowledge graphs. *arXiv preprint arXiv:1906.01195*, 2019.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[14] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.