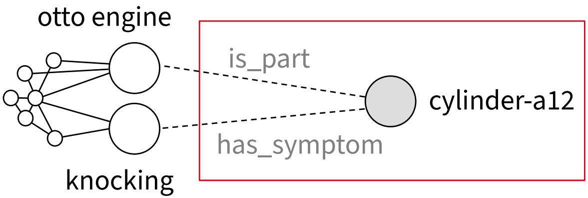
Open-World Knowledge Graph Completion Benchmarks for Knowledge Discovery

Felix Hamann, Adrian Ulges, Dirk Krechel, Ralph Bergmann

July 28, 2021

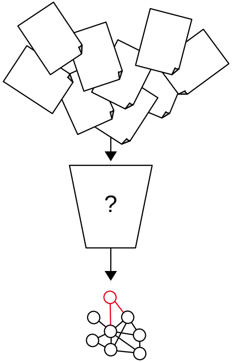
- KGs are an important part of industrial knowledge management
- Usually knowledge engineers hand-craft these KGs
- A common task: identify and link a new domain entity



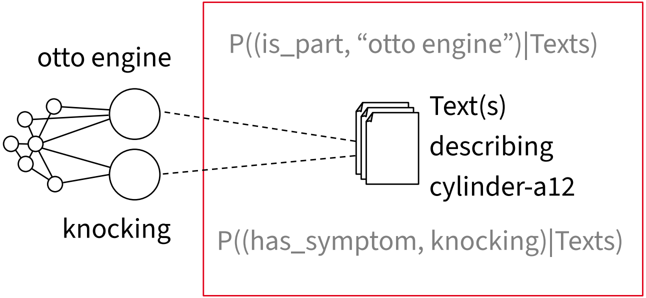
Knowledge Discovery: Challenges

- Expensive: Domain experts are required
- Tedious: Uncomfortable KG maintenance
- Unguided: Disregards accumulated (unstructured) knowledge

- Proposal: Use textual information to predict entity relations
  e.g. from an issue tracker



- **Open-World Knowledge Graph Completion** (OW-KGC)
- Predict links of *unseen* (i.e. open-world) entities
- Use text data for inference



Closing the Gap

LAVIS | RheinMain University of Applied Sciences

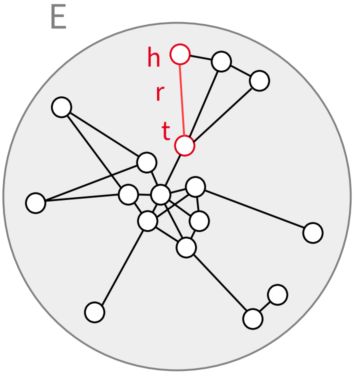
- Several benchmarks exist [1, 2, 3]
  - Open-world entities are randomly drawn
  - Concise single-sentence descriptions
- Unrealistic?
  1. There is fixed **world knowledge**
     e.g. all mechanical parts suffer wear and tear
  2. The unstructured text corpora only offer **incidental mentions**
     (but there may be many of those + noise)

LAVIS | RheinMain University of Applied Sciences

1. **Benchmark construction**
   - Formulate split criteria for open-world/closed-world splits
   - Sample textual information for these datasets
   - Try it on current KGC benchmark datasets
2. **Model approach**
   - A neural, multi-context approach to OW-KGC
   - Studies and experiments

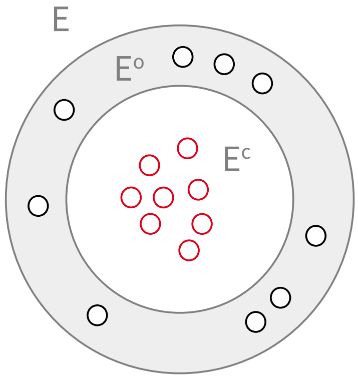
Benchmark Construction (IRT)

- Reference implementations:
  - **IRT-FB** based on FB15k-237 [4]
  - **IRT-CDE** based on CoDEx [5]
- Graph: $G = (E, R, T)$
- Triple-set $(h, r, t) \in T \subset E \times R \times E$



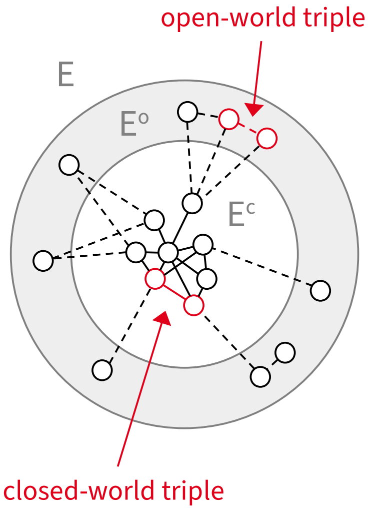
Open-World/Closed-World -Split

- Graph: $G = (E, R, T)$
- Triple-set $(h, r, t) \in T \subset E \times R \times E$
- Split entities:
  - **closed-world**: E^c
  - **open-world**: $E^o = E \backslash E^c$



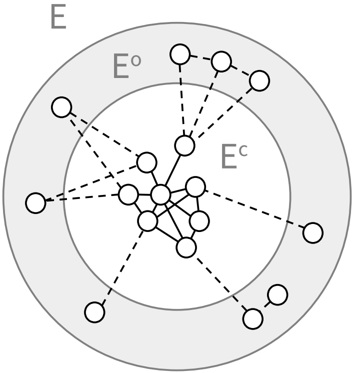
Open-World/Closed-World -Split

- Graph: $G = (E, R, T)$
- Triple-set $(h, r, t) \in T \subset E \times R \times E$
- Entity-partition: E^c, E^o
- Split triples:
  - **closed-world:** T^c (model training)
  - **open-world:** T^o (validation/test)



Open-World/Closed-World -Split

- Graphs:
  - $G^c = (E^c, R, T^c)$ (closed-world)
  - $G^o = (E, R, T^o)$ (open-world)
- Constraints:
  - $T^c \cap T^o = \emptyset$ (no test leakage)
  - $E^c \cap E^o = \emptyset$ (zero-shot)



Concept Entities

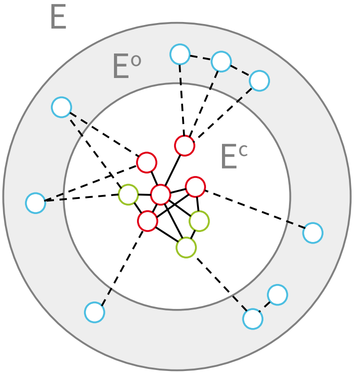
- Goal: Emulate world knowledge by selecting **concept entities**
- Selection criterium: Disproportion of heads and tails

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

- For example:
  - 353 states on 7 continents: $\frac{7}{353} \simeq 0.0198$
  - 157 headquarters located in 66 cities: $\frac{66}{157} \simeq 0.4203$

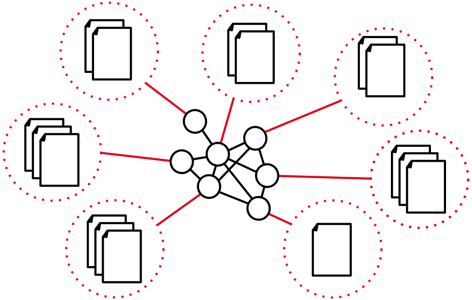
Creating the split

- concept entities: add to E^c and T^c
- open-world entities: while $|T^o|$ too small
  - Select randomly from remaining $E \backslash (E^c \cup E^o)$ and add to E^o and T^o
- remaining entities: add to E^c and T^c



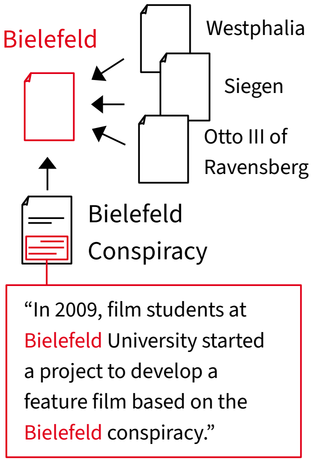
Associate Text

- Models infer links for open-world entities using text
- Required:
  - Incidental **mentions** of the entities
  - Multiple **contexts** of these mentions



Wikipedia Link Graph

- **Mentions:**
  From Wikipedia link-graph
- **Contexts:**
  Use only back-linking pages
- **Samples:**
  Select sentences randomly
  we take up to 30



- Reference dataset statistics

	IRT-FB	IRT-CDE
entities	14541	17050
triples	310116	206205
concept entitities	2389	2548
open-world entities	2377	4959

Model Approach

- Model:

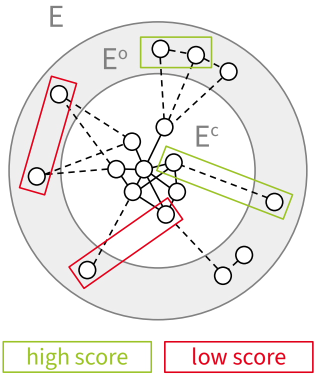
$$\phi : (E \cup C) \times R \times (E \cup C) \mapsto \mathbb{R}$$

- Tail-prediction:

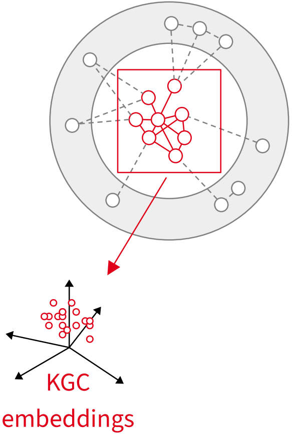
$$t^* = \underset{t' \in E \cup C}{\text{argmax}} \ \phi(h, r, t')$$

- For example:

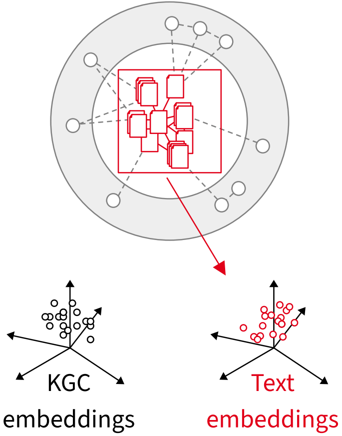
$$\phi(\{\text{“north american actor”}\}, \textit{profession}, ?)$$



- We employ the pipeline approach of [3]:
  1. Train a KGC model on T^c
     (we use DistMult [6])

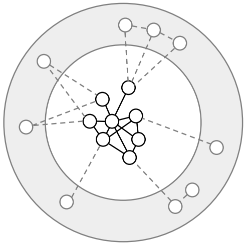


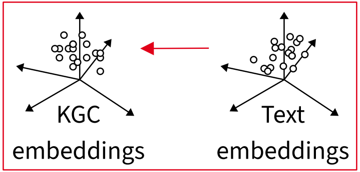
- We employ the pipeline approach of [3]:
  1. Train a KGC model on T^c
     (we use DistMult [6])
  2. Obtain text embeddings
     (we use BERT [7])



Pipeline Approach: Overview

- We employ the pipeline approach of [3]:
  1. Train a KGC model on T^c
     (we use DistMult [6])
  2. Obtain text embeddings
     (we use BERT [7])
  3. Learn a **projection** of the text embedding space to the KGC embedding space

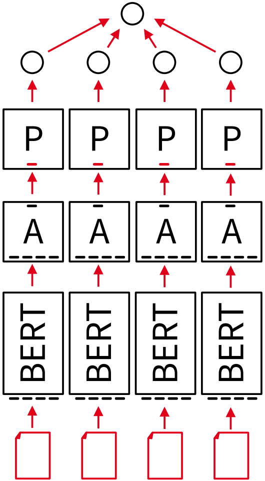




Two-Step Approach: Different Modes for the Model

For a single entity:

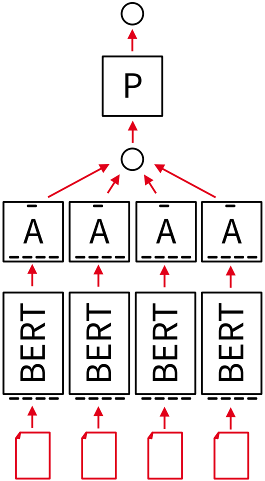
- **Single-context:**
  - **A:** Take CLS- or max-pooled -token(s)
  - **P:** Project n embeddings independently
  - Average projections for inference



Two-Step Approach: Different Modes for the Model

For a single entity:

- **Single-context:**
  - **A:** Take CLS- or max-pooled -token(s)
  - **P:** Project *n* embeddings independently
  - Average projections for inference
- **Multi-context:**
  - **A:** Max-pool all CLS-token
  - **P:** Project single embedding
  - Select single embedding for inference



Two-Step Approach: Different Modes for the Data

- **Marked:** Guide the model to better recocgnise what to look for
  - “[CLS] The quick brown [BEG] fox [END] jumps over the lazy dog . [SEP]”
- **Masked:** Focus on the context and not the mention identity
  - “[CLS] The quick brown [MASK] jumps over the lazy dog . [SEP]”
- **Clean:** Neither withhold any information nor guide the model
  - “[CLS] The quick brown fox jumps over the lazy dog . [SEP]”

Impact of aggregation

Mode	Inst.	Agg.	IRT-FB H@10	IRT-CDE H@10
baseline			17.08	16.11
marked	single	max†	19.75	26.22
marked	single	max	19.47	19.50
marked	single	cls	22.75	32.15
marked	multi	cls	26.86	36.18
clean	single	max†	20.29	25.88
clean	single	max	25.34	21.76
clean	single	cls	25.45	31.65
clean	multi	cls	29.60	31.39
masked	single	max†	18.61	25.00
masked	single	max	23.69	19.71
masked	single	cls	27.62	33.77
masked	multi	cls	**34.18**	**40.67**

Experiment Results: Modes

Different text modes

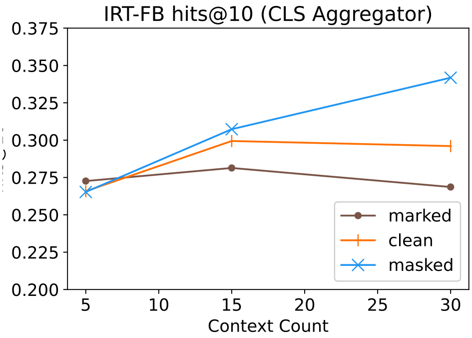
Mode	Inst.	Agg.	IRT-FB H@10	IRT-CDE H@10
baseline			17.08	16.11
marked	single	max†	19.75	26.22
marked	single	max	19.47	19.50
marked	single	cls	22.75	32.15
marked	multi	cls	26.86	36.18
clean	single	max†	20.29	25.88
clean	single	max	25.34	21.76
clean	single	cls	25.45	31.65
clean	multi	cls	29.60	31.39
masked	single	max†	18.61	25.00
masked	single	max	23.69	19.71
masked	single	cls	27.62	33.77
masked	multi	cls	**34.18**	**40.67**

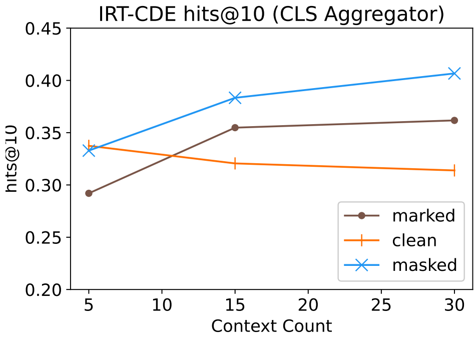
Experiment Results: Modes

Single- vs. multi-context

Mode	Inst.	Agg.	IRT-FB H@10	IRT-CDE H@10
baseline			17.08	16.11
marked	single	max†	19.75	26.22
marked	single	max	19.47	19.50
marked	single	cls	22.75	32.15
marked	multi	cls	26.86	36.18
clean	single	max†	20.29	25.88
clean	single	max	25.34	21.76
clean	single	cls	25.45	31.65
clean	multi	cls	29.60	31.39
masked	single	max†	18.61	25.00
masked	single	max	23.69	19.71
masked	single	cls	27.62	33.77
masked	multi	cls	**34.18**	**40.67**

Experiment Results: Context Sizes





LAVIS | RheinMain University of Applied Sciences

Concise vs. noisy text samples:

			IRT-FB H@10	IRT-CDE H@10
baseline	our text	1	7.22	8.09
	their text	1	**14.51**	**15.14**
cls agg.	our text	1	19.77	32.03
	their text	1	**30.25**	**45.73**
multi-ctx	our text	30	34.18	40.67

- **theirs/IRT-FB:** Wikidata descriptions assigned in FB15k-237-OWE [3]
- **theirs/IRT-CDE:** First sentence of associated Wikipedia page provided in CoDEx [5]

Thank you

- **Thank you!**
- Get the dataset: https://github.com/lavis-nlp/irt
- Get the models: https://github.com/lavis-nlp/irtm

Bibliography

[1] Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. Representation learning of knowledge graphs with entity descriptions. In *30th AAAI*, 2016.

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

[3] Haseeb Shah, Johannes Villmow, Adrian Ulges, Ulrich Schwanecke, and Faisal Shafait. An open-world extension to knowledge graph completion models. In *33rd AAAI*, 2019.

[4] Kristina Toutanova and Danqi Chen. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop for IJCNLP*, pages 57–66. ACL, July 2015.

[5] Tara Safavi and Danai Koutra. CoDEx: A Comprehensive Knowledge Graph Completion Benchmark. In *Proceedings of the 2020 EMNLP*, 2020.

[6] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

[7] 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.