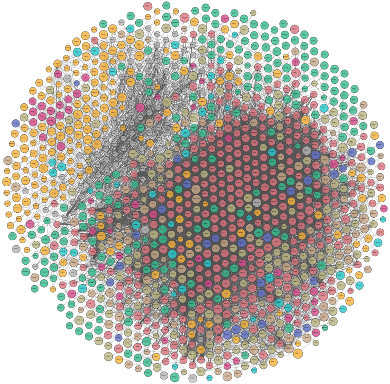
EAGER: Embedding-Assisted Entity Resolution for Knowledge Graphs

Jonathan Schuchart & Daniel Obraczka

Entity Resolution is important...



https://lod-cloud.net

- Knowledge Graphs store real-world information in machine-readable way
- Integrating multiple data sources is essential for complex information needs such as
  - Semantic Search
  - Question Answering
  - Recommender Systems
  - ...

…but it is a difficult task

Integrating Knowledge Graphs is hard:

- usually heterogeneous schemata
- different standards of data representation
- dissimilar levels of quality

Related Work

Conventional Methods

Conventional ER has a long history and many different approaches

- usually based learning a classifier or similarity measure
- geared towards tabular data
- Magellan (Konda et al. 2016: Magellan: Toward building entity matching management systems)
- DeepMatcher (Mudgal et al. 2018: Deep learning for entity matching: A design space exploration)

ScaDS.AI DRESDEN LEIPZIG

Embedding Based Methods

More recently, embedding based methods, especially for KGs have been introduced

- based on embedding two nodes+relations of KGs into a shared embedding space
- using a similarity measure for ranking potential matches
- BootEA (Sun, Z. et al. 2018: Bootstrapping entity alignment with knowledge graph embedding)
- MultiKE (Zhang, Q. et al. 2019: Multi-view knowledge graph embedding for entity alignment)
- RDGCN (Wu et al. 2019: Relation-aware entity alignment for heterogeneous knowledge graphs)

EAGER

Overview

Embedding **A**ssisted Knowledge **G**raph **E**ntity **R**esolution

- combine embedding techniques and conventional resolution methods
- embedding vectors and attribute comparisons as input for classification

Embedding Techniques

3 different techniques used:

- BootEA
- MultiKE
- RDGCN

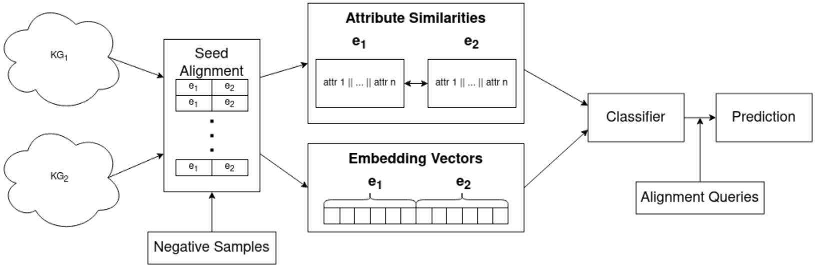
Very different approaches to embeddings but similar results in benchmarks[1]

[1] Sun et al. 2020: "A benchmarking study of embedding-based entity alignmentfor knowledge graphs"

Attribute Matching

- concatenate all attributes of each entity
- apply different similarity measures to each combination
  - Levenshtein
  - Jaccard
  - Trigram-Dice
- simple, yet already quite effective

Putting it all together



Datasets

ScaDS.AI DRESDEN LEIPZIG

Shallow Datasets

Constructed from tabular data

- movie datasets IMDB, TheMovieDB and TheTVDB
  - ≈5000 – 8000 entities
  - ≈2500 matches
- overall 3 datasets
- known and consistent schemata across KGs
- but sparse relations and disconnected graphs

Rich Datasets

Rich, existing KGs taken from benchmark study (Sun et al. 2020)

- Wikidata, DBPedia, Yago
- different combinations/versions:
  - sparse and densely sampled relations
  - multilingual (EN-DE, EN-FR)
  - small (15K entities/matches) and large (100K entities/matches)
  - 16 datasets overall

Training Split

Following the benchmark guidelines:

- Training-Validation-Test split: 20-10-70
- 5 fold cross-validation each
- rich datasets came pre-split, shallow datasets treated accordingly

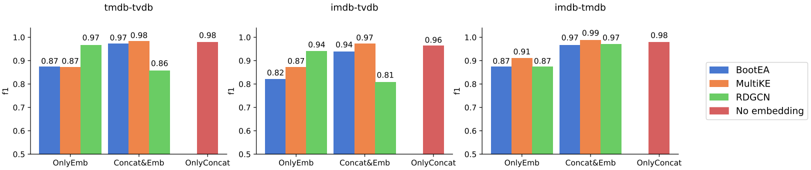
Results

Experiment Setup

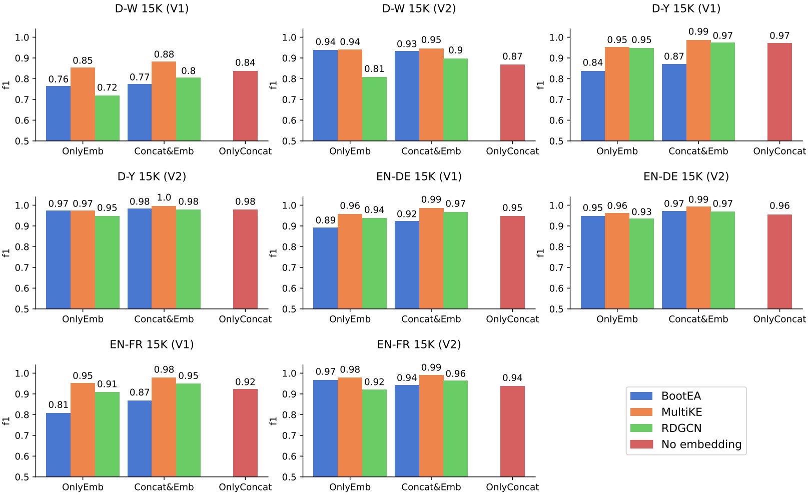
Investigate performance of combination trough ablation study
→Three different inputs for EAGER:

- **OnlyEmb**: Only use embeddings
- **OnlyConcat**: Only use concatenated attribute similarities
- **Concat&Emb**: Use both

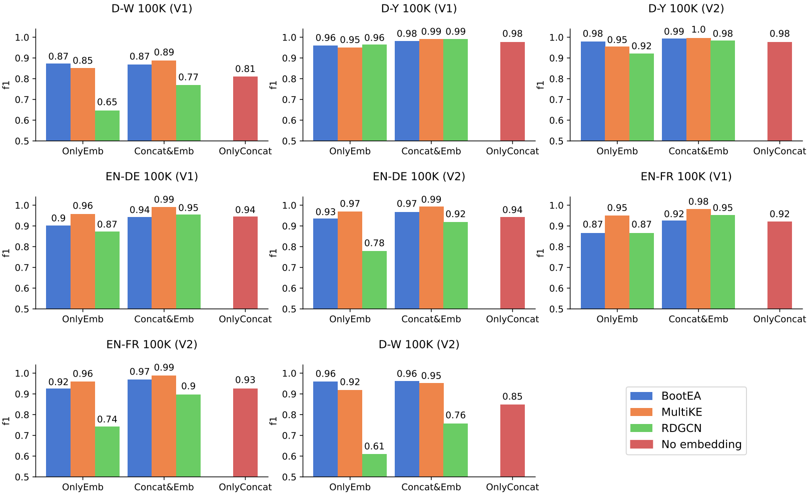
Shallow Datasets (for MLP)



Rich Datasets 15K (for MLP)



Rich Datasets 100K (for MLP)



Comparison

Comparison with other ER systems

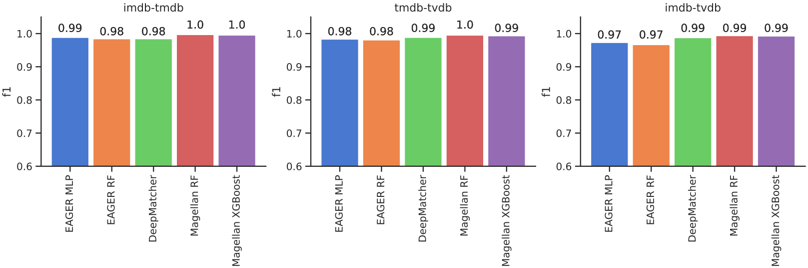
- Magellan: ER tool to match tables (ML classifiers available)
- DeepMatcher: Deep Learning Framework for ER
- Same train-test-val split as before

Comparison with other ER systems

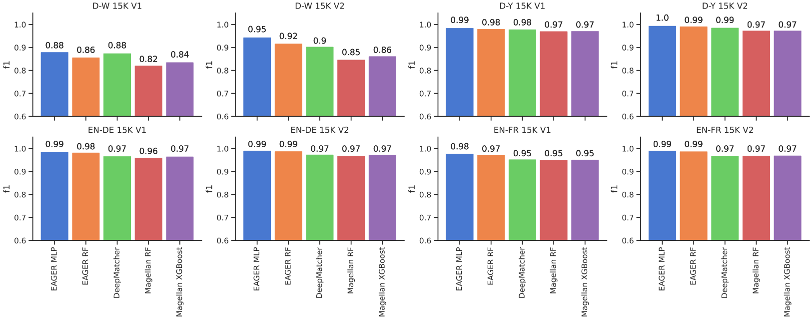
- Magellan: ER tool to match tables (ML classifiers available)
- DeepMatcher: Deep Learning Framework for ER
- Same train-test-val split as before

Magellan & DeepMatcher assume matched schemata
$\Rightarrow$ for shallow datasets match by hand, for rich datasets concatenate all attributes into single attribute

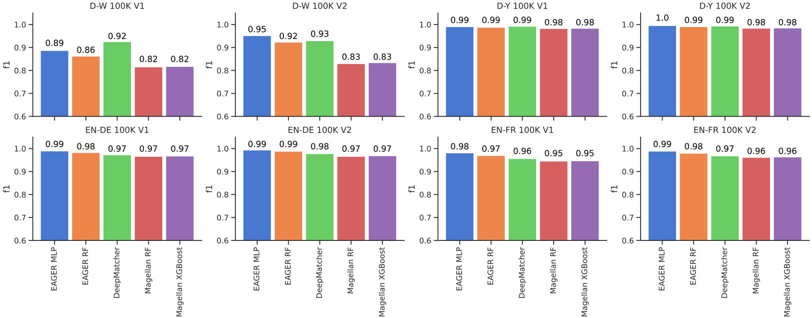
Shallow Datasets



Rich Datasets 15K



Rich Datasets 100K



Outlook & Conclusion

Outlook

- Investigate alternatives to blocking:
  - Clustering of KGE
  - Use sentence transformers for attributes $\rightarrow$ cluster those embeddings
- Use sentence embeddings instead of string similarities
- improve negative sampling strategy

Conclusion

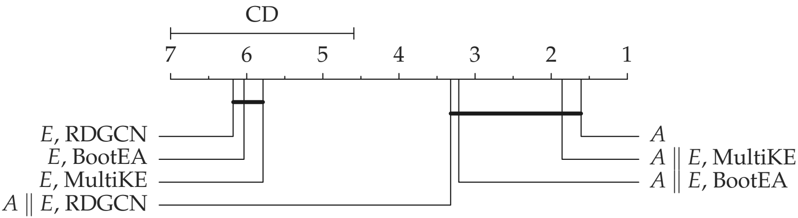
- Combining knowledge graph embeddings and attribute similarities leads to significantly better results, than either on their own
- Investigated different embedding techniques and classifiers
- Comparison with SOTA ER Frameworks: Competitive results on shallow data, outperforming on deeper KGs
- Code on Github: `https://github.com/jonathanschuchart/eager`

Thank You

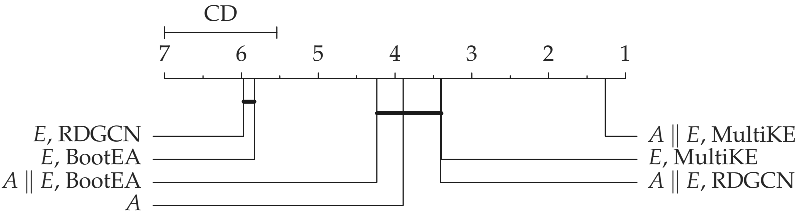
Details Results

	Dataset	$EAGER_{A\Vert E}$						$EAGER_A$		$EAGER_E$					
		BootEA		MultiKE		RDGCN				BootEA		MultiKE		RDGCN	
		MLP	RF	MLP	RF	MLP	RF	MLP	RF	MLP	RF	MLP	RF	MLP	RF
	imdb-tmdb	0.967	0.977	0.988	0.984	0.969	0.975	0.979	0.980	0.874	0.859	0.911	0.913	0.874	0.873
	imdb-tvdb	0.938	0.960	0.973	0.967	0.940	0.953	0.965	0.960	0.821	0.786	0.873	0.844	0.807	0.792
	tmdb-tvdb	0.973	0.977	0.983	0.981	0.966	0.977	0.980	0.978	0.874	0.844	0.871	0.877	0.857	0.831
15K	D-W(V1)	0.775	0.668	0.881	0.858	0.805	0.842	0.827	0.828	0.764	0.678	0.853	0.871	0.718	0.707
15K	D-W(V2)	0.934	0.841	0.945	0.918	0.897	0.890	0.868	0.870	0.938	0.847	0.939	0.942	0.808	0.796
15K	D-Y(V1)	0.870	0.775	0.986	0.982	0.974	0.986	0.972	0.971	0.837	0.746	0.952	0.941	0.947	0.953
15K	D-Y(V2)	0.983	0.908	0.995	0.993	0.977	0.991	0.978	0.978	0.975	0.888	0.973	0.971	0.947	0.960
15K	EN-DE(V1)	0.923	0.852	0.986	0.984	0.966	0.976	0.947	0.945	0.891	0.798	0.957	0.950	0.937	0.955
15K	EN-DE(V2)	0.970	0.918	0.992	0.990	0.968	0.978	0.956	0.955	0.946	0.875	0.961	0.958	0.934	0.956
15K	EN-FR(V1)	0.868	0.736	0.978	0.973	0.950	0.963	0.922	0.920	0.806	0.709	0.952	0.942	0.907	0.935
15K	EN-FR(V2)	0.965	0.876	0.991	0.989	0.963	0.977	0.937	0.936	0.942	0.875	0.977	0.978	0.921	0.948
100K	D-W(V1)	0.873	0.850	0.887	0.862	0.768	0.774	0.810	0.811	0.868	0.820	0.850	0.871	0.645	0.556
100K	D-W(V2)	0.962	0.927	0.951	0.923	0.756	0.792	0.845	0.844	0.959	0.916	0.917	0.957	0.610	0.609
100K	D-Y(V1)	0.980	0.958	0.990	0.987	0.991	0.993	0.975	0.975	0.959	0.942	0.949	0.954	0.963	0.968
100K	D-Y(V2)	0.993	0.965	0.995	0.990	0.983	0.989	0.976	0.975	0.979	0.958	0.953	0.978	0.921	0.968
100K	EN-DE(V1)	0.943	0.907	0.989	0.982	0.954	0.961	0.944	0.943	0.901	0.859	0.956	0.947	0.872	0.891
100K	EN-DE(V2)	0.965	0.933	0.993	0.988	0.926	0.932	0.943	0.941	0.934	0.890	0.970	0.969	0.779	0.847
100K	EN-FR(V1)	0.925	0.867	0.981	0.969	0.947	0.938	0.920	0.919	0.866	0.819	0.948	0.943	0.866	0.894
100K	EN-FR(V2)	0.968	0.899	0.989	0.979	0.897	0.901	0.925	0.923	0.925	0.877	0.959	0.968	0.742	0.806
	Avg Rank	6.211	10.105	1.316	2.842	6.895	5.474	6.947	7.632	8.947	12.789	6.737	6.211	12.000	10.895

Significance Diagram Shallow



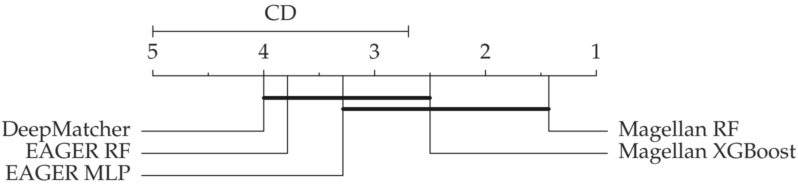
Significance Diagram Rich



Details Comparison

	Dataset	EAGER MLP			EAGER RF			DeepMatcher			Magellan XGBoost			Magellan RF		
		fm	prec	rec	fm	prec	rec	fm	prec	rec	fm	prec	rec	fm	prec	rec
	imdb-tmdb	0.990	0.987	0.993	0.986	0.979	0.994	0.983	0.970	0.996	0.996	0.999	0.994	0.997	0.998	0.997
	imdb-tvdb	0.979	0.965	0.994	0.974	0.951	0.999	0.989	0.981	0.998	0.991	0.990	0.992	0.992	0.989	0.995
	tmdb-tvdb	0.991	0.992	0.990	0.985	0.988	0.982	0.987	0.977	0.997	0.993	0.991	0.994	0.995	0.993	0.997
15K	D-W(V1)	0.898	0.989	0.823	0.872	0.991	0.779	0.876	0.844	0.910	0.837	0.886	0.793	0.823	0.865	0.784
	D-W(V2)	0.968	0.990	0.948	0.909	0.992	0.838	0.904	0.895	0.913	0.863	0.899	0.830	0.848	0.859	0.837
	D-Y(V1)	0.985	1.000	0.971	0.985	1.000	0.971	0.979	0.974	0.983	0.971	0.985	0.957	0.971	0.984	0.958
	D-Y(V2)	0.996	0.999	0.993	0.993	0.999	0.986	0.986	0.985	0.987	0.974	0.972	0.977	0.974	0.973	0.976
	EN-DE(V1)	0.985	0.996	0.973	0.984	0.995	0.973	0.971	0.976	0.966	0.969	0.992	0.948	0.962	0.977	0.948
	EN-DE(V2)	0.992	0.996	0.988	0.989	0.997	0.982	0.974	0.967	0.982	0.973	0.993	0.954	0.969	0.984	0.955
	EN-FR(V1)	0.980	0.995	0.965	0.973	0.994	0.952	0.956	0.959	0.953	0.953	0.983	0.924	0.952	0.979	0.926
	EN-FR(V2)	0.990	0.998	0.982	0.990	0.996	0.984	0.966	0.963	0.970	0.971	0.992	0.951	0.970	0.992	0.950
100K	D-W(V1)	0.873	0.996	0.777	0.864	0.990	0.767	0.926	0.907	0.945	0.815	0.907	0.741	0.812	0.896	0.742
	D-W(V2)	0.965	0.989	0.941	0.926	0.988	0.871	0.936	0.924	0.949	0.836	0.925	0.762	0.831	0.897	0.774
	D-Y(V1)	0.991	1.000	0.982	0.988	1.000	0.977	0.992	0.990	0.994	0.984	0.994	0.974	0.983	0.991	0.974
	D-Y(V2)	0.997	0.999	0.995	0.991	1.000	0.982	0.993	0.993	0.994	0.985	0.983	0.987	0.984	0.982	0.987
	EN-DE(V1)	0.990	0.997	0.982	0.982	0.997	0.968	0.972	0.971	0.972	0.968	0.990	0.946	0.967	0.988	0.946
	EN-DE(V2)	0.993	0.997	0.990	0.987	0.997	0.978	0.975	0.972	0.978	0.968	0.993	0.945	0.966	0.987	0.946
	EN-FR(V1)	0.980	0.997	0.964	0.969	0.994	0.944	0.956	0.959	0.953	0.947	0.988	0.910	0.946	0.985	0.911
	EN-FR(V2)	0.989	0.995	0.983	0.981	0.992	0.970	0.964	0.959	0.969	0.964	0.991	0.938	0.962	0.987	0.938
	Avg Rank	1.579			2.579			2.895			3.737			4.211		

Significance Diagram Comparison Shallow



Significance Diagram Comparison Rich