RUHR-UNIVERSITÄT BOCHUM

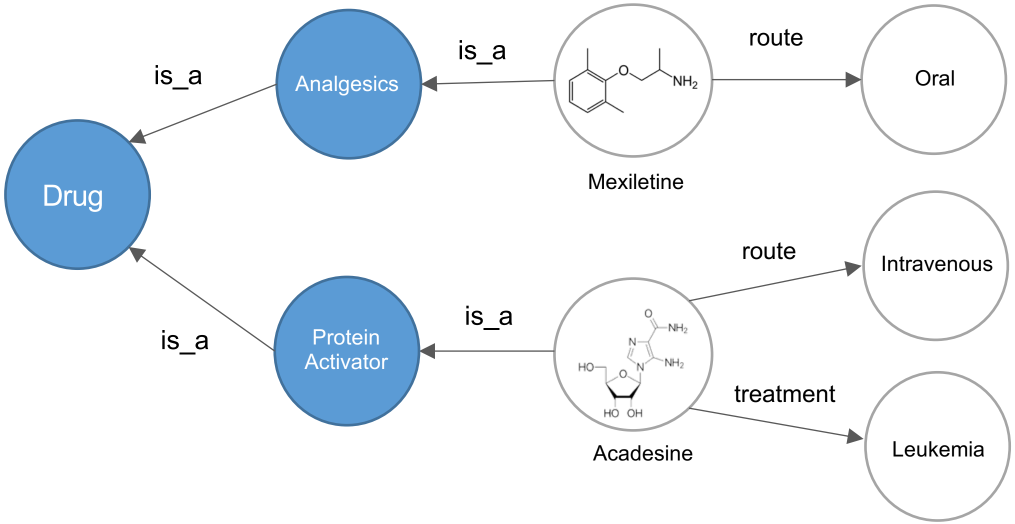
SYMBOLIC AND SUB-SYMBOLIC REPRESENTATIONS OF KNOWLEDGE GRAPHS – AN **INTRODUCTION**

Jun.-Prof. Dr. Maribel Acosta

Outline

1. **Introduction to Knowledge Graphs**
2. Symbolic Representations of Knowledge Graphs
3. Sub-Symbolic Representations of Knowledge Graphs
4. The Problem of Knowledge Graph Completion
5. Conclusion and Future Work

Knowledge Graphs



Knowledge representation where statements correspond to nodes and edges, where:

- **Nodes** are labelled and represent concepts, entities, or data values
- **Edges** are labelled and represent binary connections between nodes
- Concepts and properties are defined in a **vocabulary** or **ontology** (semantics)

RUHR UNIVERSITÄT BOCHUM

Evolution of Knowledge Graphs

review articles

DOI:10.1145/3418294

Tracking the historical events that lead to the interweaving of data and knowledge.

BY CLAUDIO GUTIERREZ AND JUAN F. SEQUEDA

Knowledge Graphs

"Those who cannot remember the past

The essential elements involved in the notion of Knowledge Graphs can be traced to ancient history in the core idea of representing knowledge in a diagrammatic form. Examples include: Aristotle and visual forms of reasoning, around 350 BC; Lull and his tree of knowledge; Linnaeus and taxonomies of the natural world; and in the 19th. century, the works on formal and diagrammatic reasoning of scientists like J.J. Sylvester, Charles Peirce and Gottlob Frege. These ideas also involve several disciplines like mathematics, philosophy, linguistics, library sciences, and psychology, among others.

This article aims to provide historical context for the roots of Knowledge Graphs grounded in the advancements of the computer science disciplines of knowledge, data, and the combination thereof, and thus, focus on the developments after the advent of computing in its modern sense (1950s). To the best of

a pedagogical emphasis directed particularly to young researchers. It presents a map and guidelines to navigate through the most relevant ideas, theories, and events that, from our perspective, have triggered current developments. The goal is to help understand what worked, what did not work, and reflect on how diverse events and re-

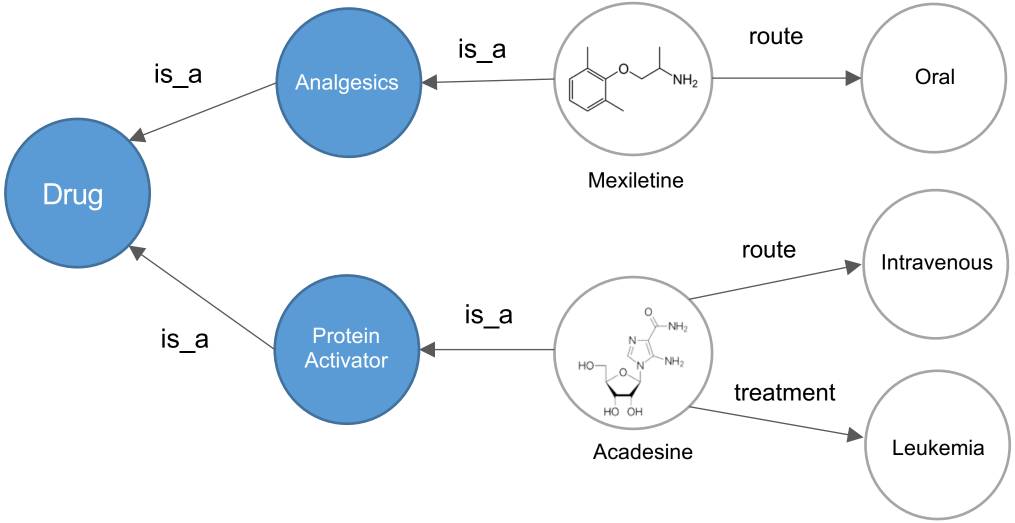
achievements of something desired or anticipated), and "limitations" (or, impediments) of the period. The idea is to motivate a reflection on a balance of the period. At the end of each section we include a paragraph indicating references to historical and/or technical overviews on the topics covered.

they developed the "General Solving Program" in 1958, which illustrates well the paradigm researchers were after: "*this program is part of a research effort by the authors to understand the information processes that underlie human intellectual, adaptive, and creative abilities.*" And the goal was stated as follows: "*to construct computer*

Source: https://cacm.acm.org/magazines/2021/3/250711-knowledge-graphs/

RUHR UNIVERSITÄT BOCHUM RUB

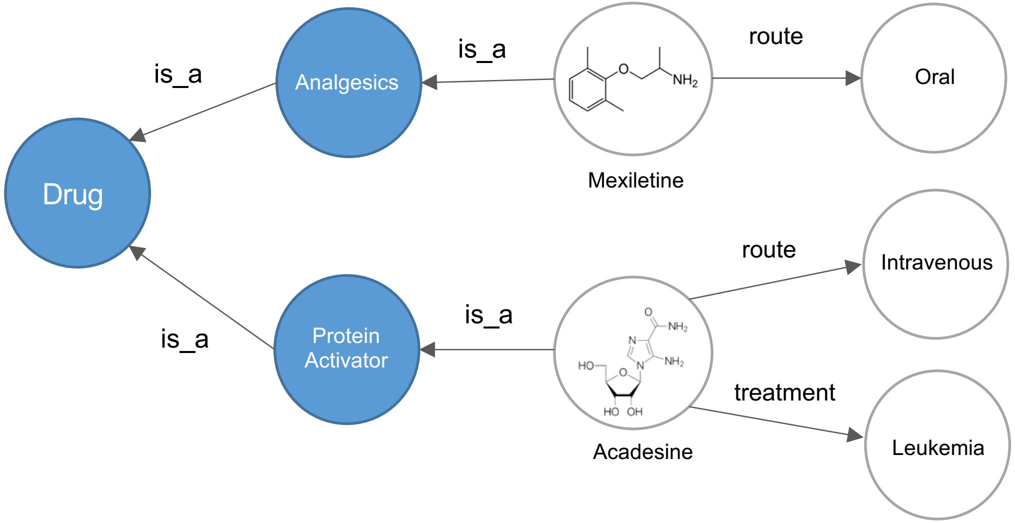
Knowledge Graphs & Open World Assumption



<u>Closed</u> World Assumption

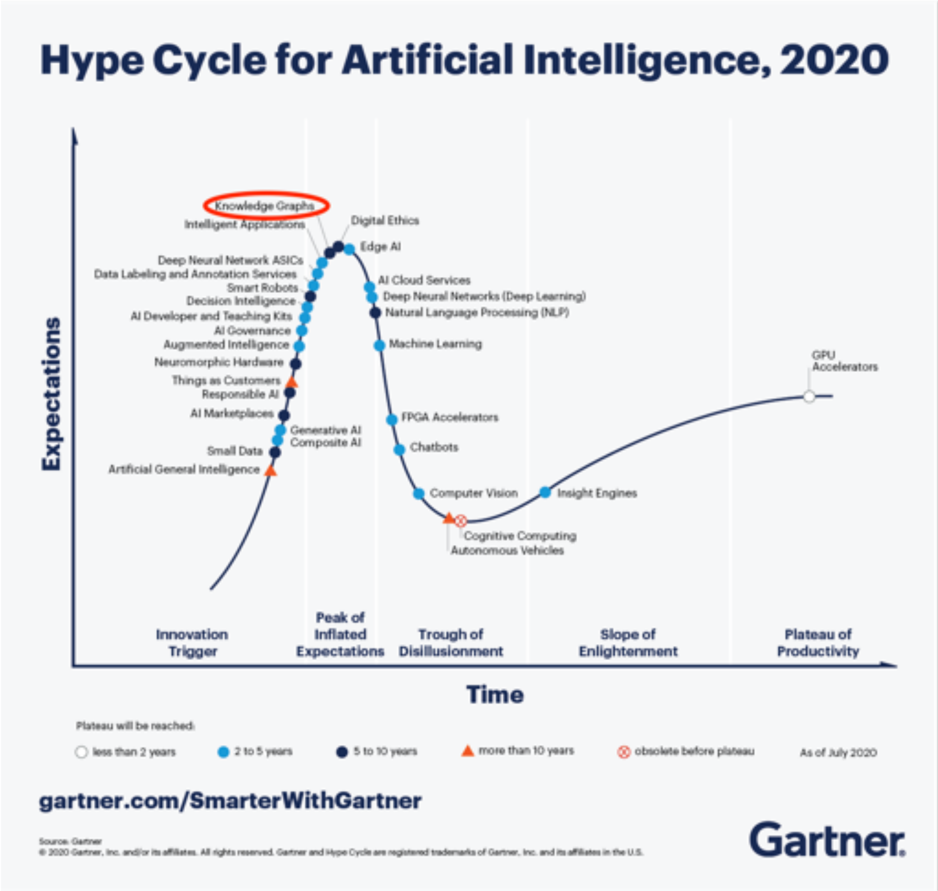
The treatment for Mexiletine is not in the graph → Mexiletine is not used in treatments.

Knowledge Graphs & Open World Assumption



<u>Open</u> World Assumption

The treatment for Mexiletine is not in the graph →
It is unknown whether Mexiletine is used in a treatment or not.


Hype Cycle for Artificial Intelligence, 2020
Expectations
Knowledge Graphs
Intelligent Applications
Digital Ethics
Edge AI
Deep Neural Network ASICs
Data Labeling and Annotation Services
Smart Robots
Decision Intelligence
AI Developer and Teaching Kits
AI Governance
Augmented Intelligence
Neuromorphic Hardware
Things as Customers
Responsible AI
AI Marketplaces
Generative AI
Composite AI
Small Data
Artificial General Intelligence
AI Cloud Services
Deep Neural Networks (Deep Learning)
Natural Language Processing (NLP)
Machine Learning
FPGA Accelerators
Chatbots
Computer Vision
Insight Engines
Cognitive Computing
Autonomous Vehicles
GPU Accelerators
Innovation Trigger
Peak of Inflated Expectations
Trough of Disillusionment
Slope of Enlightenment
Plateau of Productivity
Time
Plateau will be reached:
less than 2 years
2 to 5 years
5 to 10 years
more than 10 years
obsolete before plateau
As of July 2020
gartner.com/SmarterWithGartner
Source: Gartner
© 2020 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner and Hype Cycle are registered trademarks of Gartner, Inc. and its affiliates in the U.S.
Gartner

Applications of Knowledge Graphs

Information Retrieval

- Web search
- Question answering
- Personal assistant

E-commerce

- Product understanding
- Recommender systems
- Chatbots

Cognitive Systems

- Knowledge discovery
- Integrating interdisciplinary knowledge

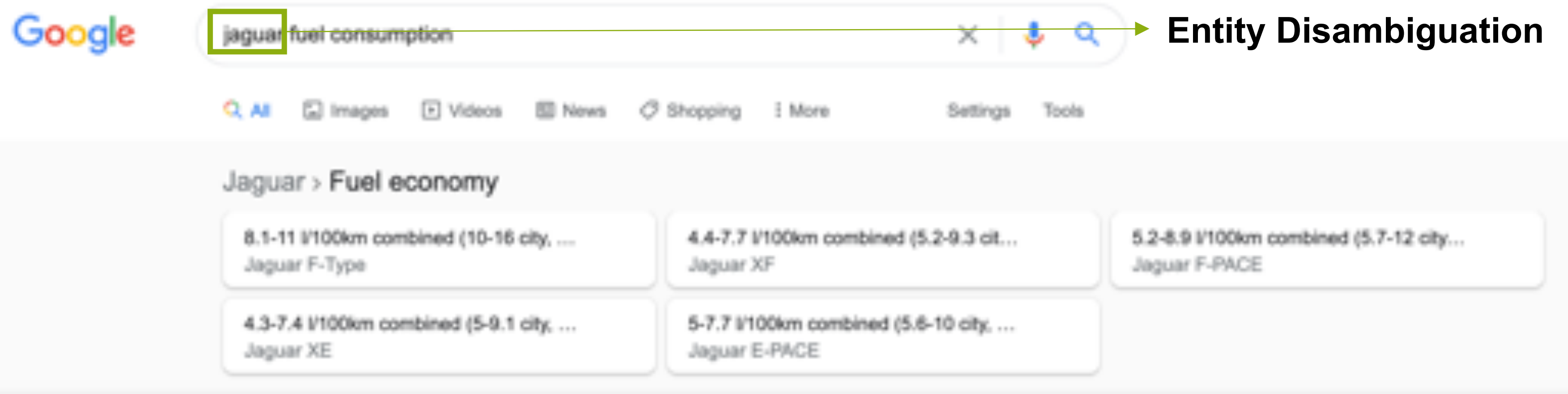
Natural Sciences

- Drug discovery and repurposing
- Medical treatment recommendation
- Reducing field experiments
- Integrating interdisciplinary knowledge

Google Knowledge Graph (1/2)

Question Answering

Google Knowledge Graph (2/2)

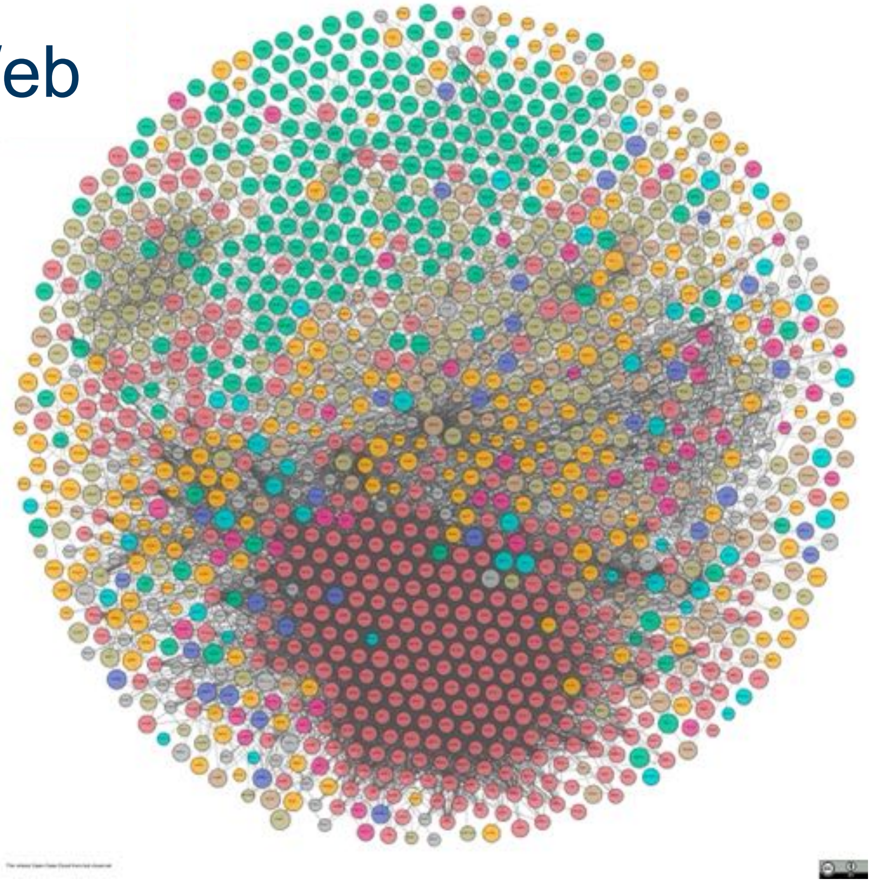


Entity Disambiguation

Knowledge Graphs on the Web

The Linked Open Data Cloud

- Depicts interlinked knowledge graphs.
- Each node is a knowledge graph.
- Edges represent links between the statements in the datasets.
- > 1,000 knowledge graphs, billions of statements.



https://lod-cloud.net/

RUHR UNIVERSITÄT BOCHUM RUB

DBpedia (1/2)

WIKIPEDIA
The Free Encyclopedia

DBpedia

Semi-structured data
from Wikipedia

https://www.dbpedia.org/

DBpedia (2/2)

DBpedia Browse using Formats Faceted Browser Sparql Endpoint

About: Germany

An Entity of Type : populated place, from Named Graph : http://dbpedia.org, within Data Space : dbpedia.org

Germany (/ˈdʒɜːrməni/; German: Deutschland, pronounced [ˈdɔʏtʃlant]), officially the Federal Republic of Germany (German: Bundesrepublik Deutschland,), is a federal parliamentary republic in central-western Europe. It includes 16 constituent states, covers an area of 357,021 square kilometres (137,847 sq mi), and has a largely temperate seasonal climate. With about 82 million inhabitants, Germany is the most populous member state of the European Union. After the United States, it is the second most popular immigration destination in the world. Germany's capital and largest metropolis is Berlin. Major urban areas include Ruhr, Hamburg, Munich, Cologne, Frankfurt and Stuttgart.

Property	Value
dbo:PopulatedPlace/areaTotal	• 357022.0910454866 • 357168.0
dbo:PopulatedPlace/populationDensity	• 225.09755843024595 • 227.0
dbo:abstract	• Germany (/ˈdʒɜːrməni/; German: Deutschland, pronounced [ˈdɔʏtʃlant]), officially the Federal Republic of Germany (German: Bundesrepublik Deutschland,), is a federal parliamentary republic in central-western Europe. It includes 16 constituent states, covers an area of 357,021 square kilometres (137,847 sq mi), and has a largely temperate seasonal climate. With about 82 million inhabitants, Germany is the most populous member state of the European Union. After the United States, it is the second most popular immigration destination in the world. Germany's capital and largest metropolis is Berlin. Major urban areas include Ruhr, Hamburg, Munich, Cologne, Frankfurt and Stuttgart. Various Germanic tribes have occupied the northern parts of modern Germany since classical antiquity. A region named Germania was documented before 100 AD. During the Migration Period the Germanic

http://dbpedia.org/page/Germany

Wikidata (1/2)

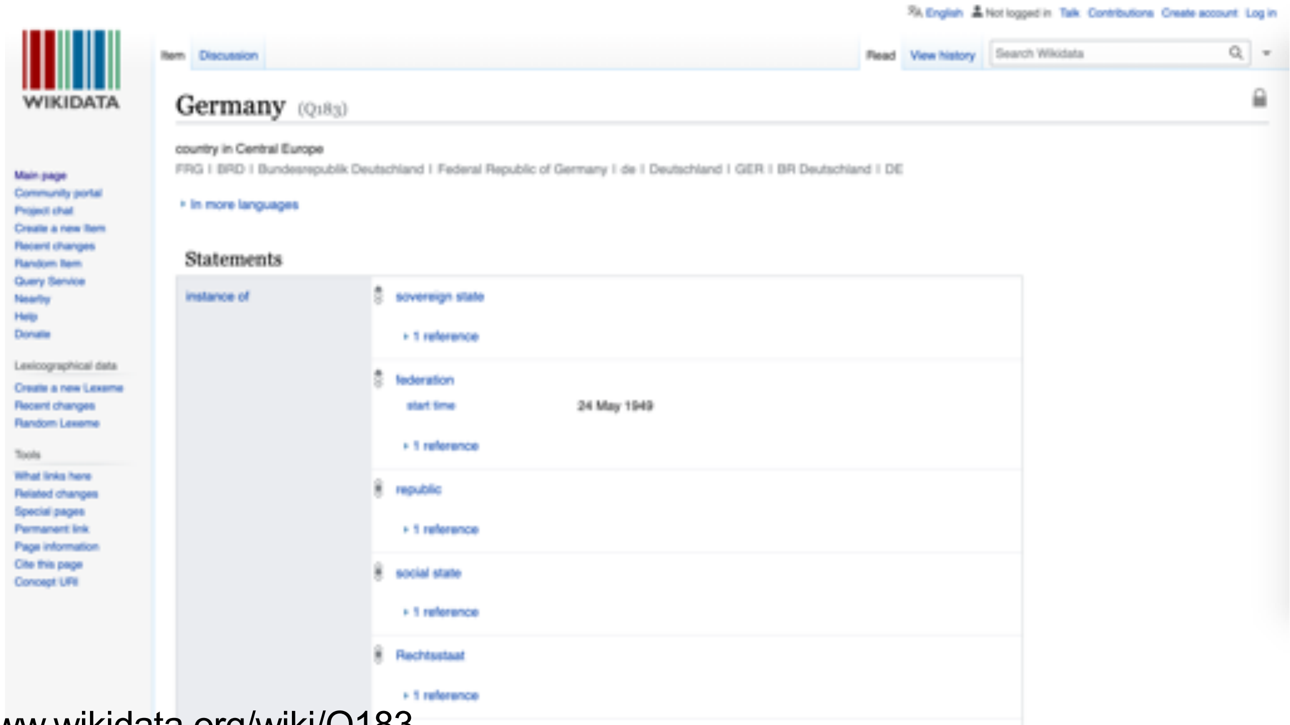
WIKIPEDIA
The Free Encyclopedia

WIKIDATA

Collaborative KG
for Wikipedia

http://wikidata.org/

Wikidata (2/2)



https://www.wikidata.org/wiki/Q183

Outline

1. Introduction to Knowledge Graphs
2. **Symbolic Representations of Knowledge Graphs**
3. Sub-Symbolic Representations of Knowledge Graphs
4. The Problem of Knowledge Graph Completion
5. Conclusion and Future Work

Statement: `RUB was founded in 1962`

Image source: https://commons.wikimedia.org/wiki/File:AudiMax_Ruhr-Uni-Bochum_HDR_1.jpg

How to Represent Knowledge?

- We want to represent the statement "`RUB was founded in 1962`" in an intuitive way.

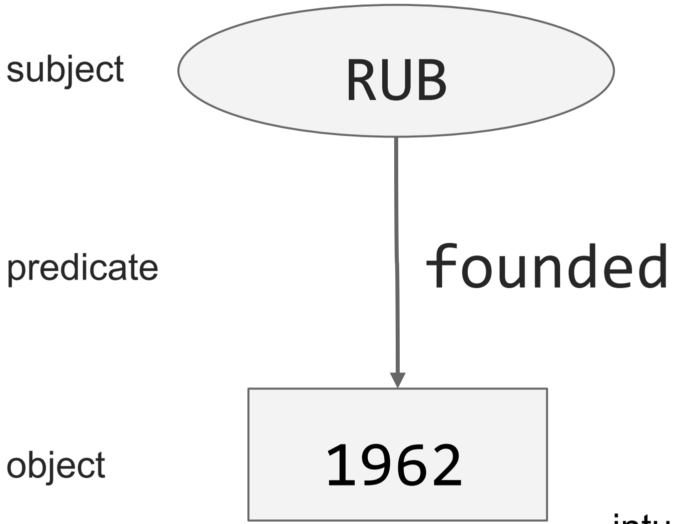


Image source: https://commons.wikimedia.org/wiki/File:Ruhr-Universit%C3%A4t_Bochum_(IA_und_weitere_I-Geb%C3%A4ude).jpg

intuitive knowledge representation with a **directed graph**

Triple-based Model for Knowledge Graphs

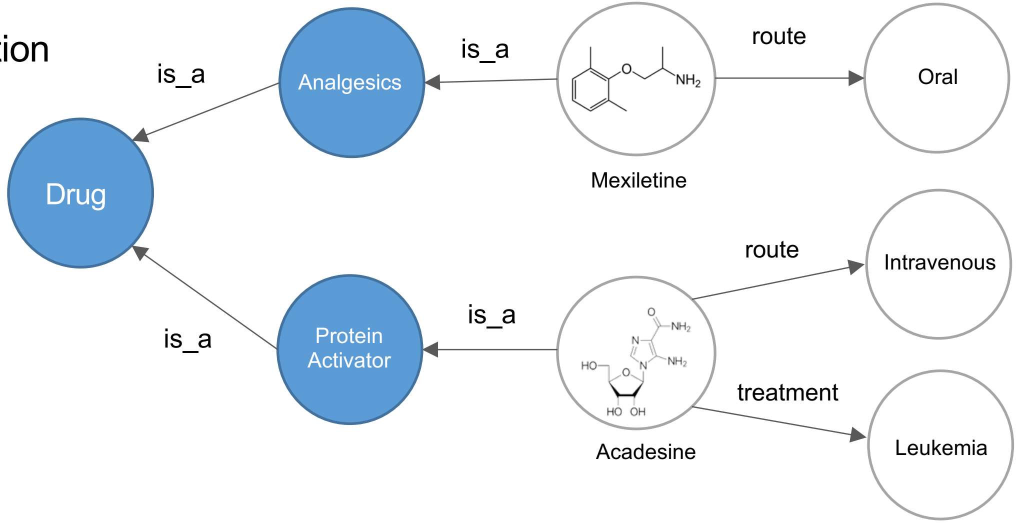
- A knowledge graph is a **labelled multidigraph** (V, E).

- Edges are represented as $s \overset{p}{\rightarrow} o$ or as triples (s, p, o), with $s, o \in V$ and $p \in E$, where
  - s is called the **subject** or **head**
  - p is called the **predicate** or **relation**
  - o is called the **object** or **tail**

A **knowledge graph** KG is a set of statements of the form (s, p, o), where s and o correspond to labelled nodes, and p corresponds to a labelled, directed edge.

Example

Visual representation

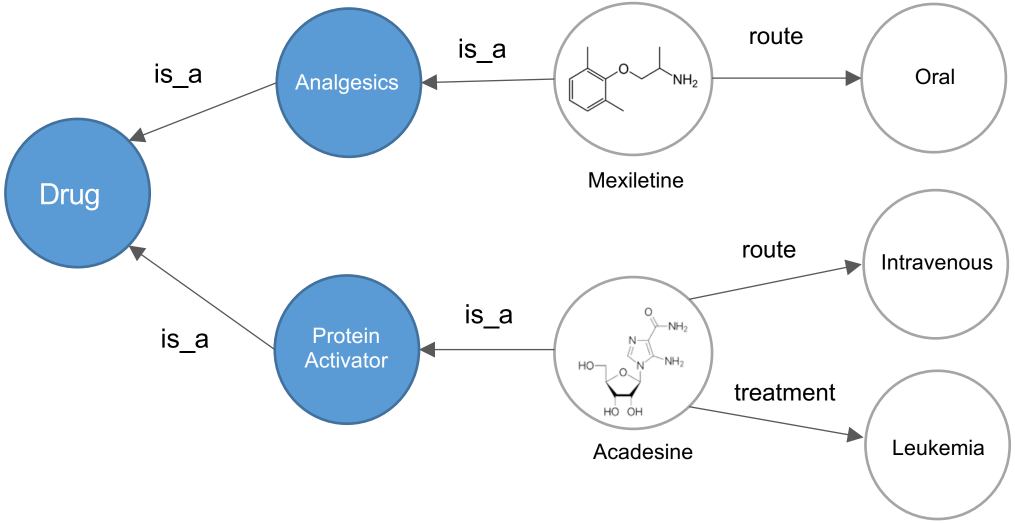


Triple-based representation:

```
(Acadesine, is_a, ProteinActivator)
(Acadesine, treatment, Leukemia)
(ProteinActivator, is_a, Drug)
```

What is "is_a"?
What is "Drug"?

Semantics in Knowledge Graphs

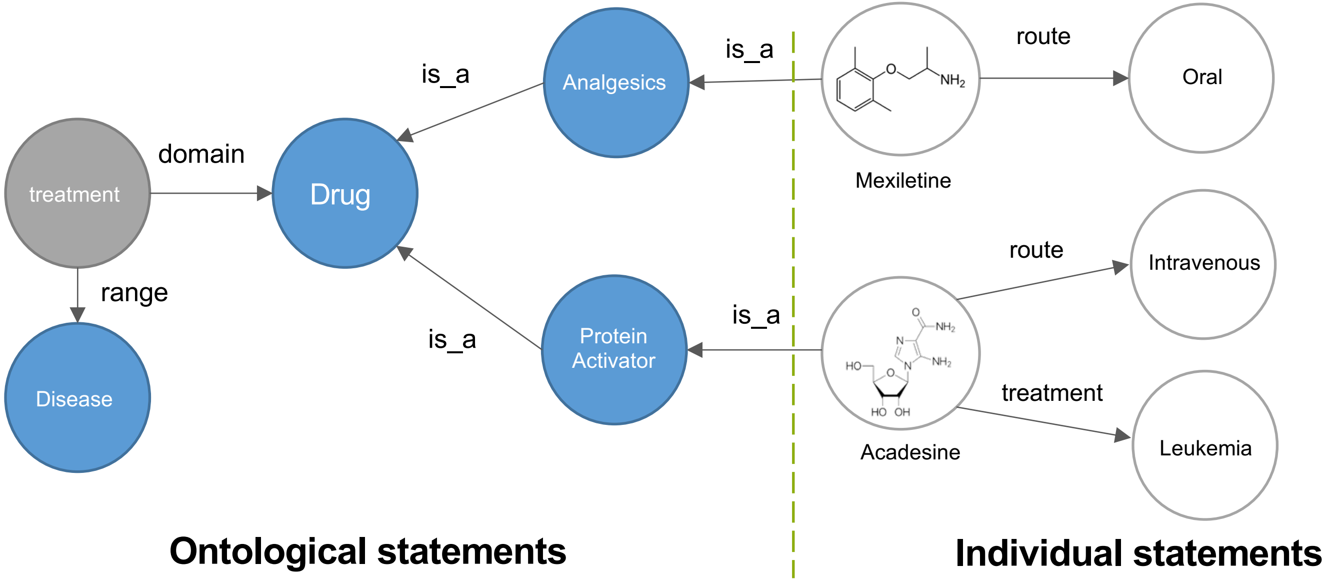


Data representation where statements correspond to nodes and edges, where:

- **Nodes** represent concepts, entities, or data values
- **Edges** are labelled and represent connections between nodes
- Concepts and properties are defined in a **vocabulary** or **ontology** (semantics)

RUHR UNIVERSITÄT BOCHUM RUB

Vocabularies and Ontologies in Knowledge Graphs



- Ontologies define the **formal meaning** of the symbols/labels used in the knowledge graph.

Vocabularies and Ontologies

- Set of schema-level terms or identifiers (classes and properties) and possibly instance-level identifiers (individuals), together with additional information.
- Represent **agreement** between people on the definition and meaning of the terms.
- In general, vocabularies and ontologies include the following definitions:
  - **(Named) Individuals:**
    Atomic unit in the vocabulary.
  - **Classes:**
    Set of individuals; a vocabulary includes the characteristics of classes.
  - **Properties:**
    Specification of properties and the characteristics of these properties.

Vocabularies and Ontolologies in the Semantic Web

Web Ontology Language (OWL) → Complex knowledge

RDF Schema (RDFS) → Hierarchies of classes and properties

Resource Description Framework (RDF) → Definition of properties

RDF Vocabulary

- The RDF[1] vocabulary contains identifiers (URIs) with defined meaning.

- The predicate `rdf:type` associates individuals with classes (this is the is_a relation).

```
:Berlin rdf:type :City .
```

- We can also define predicates using the class `rdf:Property`.

```
:population rdf:type rdf:Property .
```

[1] http://www.w3.org/1999/02/22-rdf-syntax-ns

RDFS Vocabulary

- The RDFS[1] (RDF Schema) vocabulary allows for defining classes and hierarchies.

- Classes can be declared using the pre-defined class `rdfs:Class`.

```
:Person rdf:type rdfs:Class .
```

- Hierarchies of classes can be created with the predicate `rdfs:subClassOf`.

```
:Student rdfs:subClassOf :Person .
```

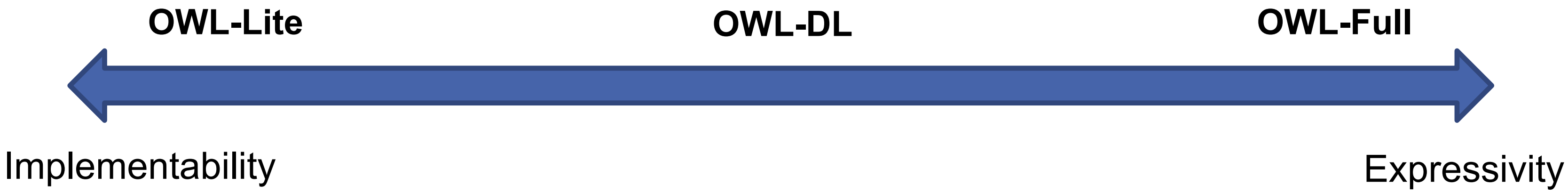
- Hierarchies of predicates can be created with the predicate `rdfs:subPropertyOf`.

```
:hasMother rdfs:subPropertyOf :hasParent .
```

[1] http://www.w3.org/2000/01/rdf-schema#

The Web Ontology Language (OWL)

- An ontology language that relies on the RDF model.
- Formal logics with a computational character are always a compromise between expressivity and implementability.
- OWL comes in different fragments which balance the user's expressivity needs with its implementability.



OWL DL

- We will focus on the OWL DL language

- Like RDFS, OWL has the concepts of class, property and instance.
  OWL is made up of terms which provide for:
  - Class axioms
  - Property axioms
  - Individual axioms
  - Class construction
  - Property construction

This tutorial

OWL DL Class Axioms

- Equivalent relationship (classes have the same individuals).
  - Example: *Every human is a person, and every person is a human*.

```
:Human owl:equivalentClass :Person .
:Alice rdf:type :Human .
:Alice rdf:type :Person .
```

- Disjointness (classes have no shared individuals).
  - Example: *Cats are not dogs.*

```
:Cat owl:disjointWith :Dog .
```

OWL DL Property Axioms

Apart from the sub-property relationship from RDFS, OWL also allows for expressing other types of property axioms.

OWL DL Property Axioms

- **Equivalent** properties (`owl:equivalentProperty`)
- **Inverse** properties (`owl:inverseOf`)
- **Transitive** property (`owl:TransitiveProperty`)
- **Symmetric** property (`owl:SymmetricProperty`)
- **Functional** property (`owl:FunctionalProperty`)
- **Inverse functional** property (`owl:InverseFunctionalProperty`)

Individual Axioms

- OWL Individuals represent instances of classes

- We can explicitly state that two individuals are the same.

```
dbr:Germany owl:sameAs wikidata:Q183 .
```

- We can explicitly state that two individuals are different.

```
dbr:Germany owl:differentFrom dbr:German_Empire .
```

What can we do with this type of semantics?

Entailment

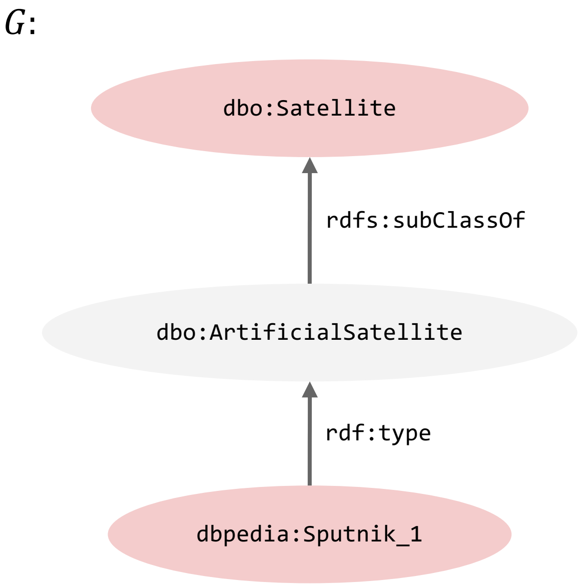
Reasoning

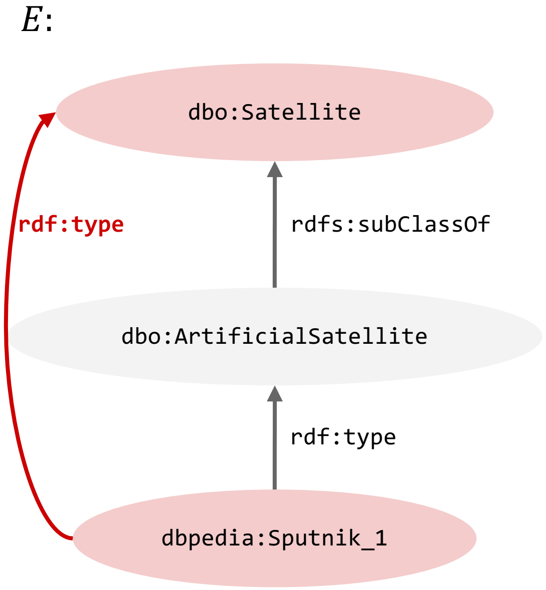
Entailment

Logical consequence (also **entailment**) is a fundamental concept in logic, which describes the relationship between statements that hold true when one statement logically follows from one or more statements.

https://en.wikipedia.org/wiki/Logical_consequence

Entailment over RDF Graphs: Example





Does G entail E?

Yes, under RDFS entailment, i.e., $G \vDash_{RDFS} E$.

Entailment over RDF Graphs

- A graph G entails another graph E (denoted $G \vDash E$), if there is a **logical consequence** from G to E.

- If two graphs G and E each entail the other ($G \vDash E$ and $E \vDash G$) then they are **logically equivalent**.

- Logical consequence is defined via entailment relations.

Entailment Relations

Entailment relations over RDF graphs are defined as a set of:

- **Axiomatic triples:**
  - A self-evident or universally recognized truth
  - Hold true for all RDF graphs

- **Entailment Rules:**
  - Define what statements can logically follow
  - Formally defined as $B \rightarrow H$, where
    - B is the rule body or antecedent
    - H is the rule head of consequent

Entailment Patterns

- Are used to specify entailment rules in RDF graphs.

- Example:

Body

Head

$$\frac{\texttt{?x rdfs:subClassOf ?y .?y rdfs:subClassOf ?z .}}{\texttt{?x rdfs:subClassOf ?z .}}\, rdfs11$$

Name of the rule

Applying Entailment Patterns

Statements:

`:Student rdfs:subClassOf foaf:Person .`

`foaf:Person rdfs:subClassOf foaf:Agent .`

Entailment pattern:

$$\frac{\boxed{\texttt{?x}}\ \texttt{rdfs:subClassOf}\ \boxed{\texttt{?y}}\ .\ \boxed{\texttt{?y}}\ \texttt{rdfs:subClassOf}\ \boxed{\texttt{?z}}\ .}{\boxed{\texttt{?x}}\ \texttt{rdfs:subClassOf}\ \boxed{\texttt{?z}}\ .}\ rdfs11$$

Entailed triple:

`:Student rdfs:subClassOf foaf:Agent .`

List of RDF/S Entailment Patterns (Selection)

	Body (If)	Head (Then)
rdfs5	`?x rdfs:subPropertyOf ?y . ?y rdfs:subPropertyOf ?z .`	`?x rdfs:subPropertyOf ?z .`
rdfs6	`?x rdf:type rdf:Property .`	`?x rdfs:subPropertyOf ?x .`
rdfs7	`?p2 rdfs:subPropertyOf ?p1 . ?x ?p2 ?y.`	`?x ?p1 ?y .`
rdfs9	`?x rdfs:subClassOf ?y . ?z rdf:type ?x .`	`?z rdf:type ?y .`
rdfs10	`?x rdf:type rdfs:Class .`	`?x rdfs:subClassOf ?x .`
rdfs11	`?x rdfs:subClassOf ?y . ?y rdfs:subClassOf ?z .`	`?x rdfs:subClassOf ?z .`
rdf1	`?s ?p ?o .`	`?p rdf:type rdf:Property .`

Full list: https://www.w3.org/TR/rdf11-mt/#patterns-of-rdfs-entailment-informative

List of OWL Entailment Patterns (Selection)

	Body (If)	Head (Then)
eq-sym	`?x owl:sameAs ?y .`	`?y owl:sameAs ?x .`
eq-rep-s	`?s owl:sameAs ?so . ?s ?p ?o .`	`?so ?p ?o .`
prp-eqp1	`?p1 owl:equivalentProperty ?p2 . ?x p1 ?y .`	`?x ?p2 ?y .`
prp-inv1	`?p1 owl:inverseOf ?p2 . ?x ?p1 ?y .`	`?y ?p2 ?x .`
prp-symp	`?p rdf:type owl:SymmetricProperty . ?x ?p ?y .`	`?y ?p ?x .`
prp-fp	`?p rdf:type owl:FunctionalProperty .` `?x ?p ?y1 . ?x ?p ?y2 .`	`?y1 owl:sameAs ?y2 .`
prp-if	`?p rdf:type owl:InverseFunctionalProperty .` `?x1 ?p ?y . ?x2 ?p ?y .`	`?x1 owl:sameAs ?x2 .`
scm-eqc1	`?c1 owl:equivalentClass ?c2 .`	`?c1 rdfs:subClassOf ?c2 .` `?c2 rdfs:subClassOf ?c1 .`
scm-eqc2	`?c1 rdfs:subClassOf ?c2 . ?c2 rdfs:subClassOf ?c1 .`	`?c1 owl:equivalentClass ?c2 .`
cax-eqc1	`?c1 owl:equivalentClass ?c2 . ?x rdf:type ?c1 .`	`?x rdf:type ?c2 .`

Full list: https://www.w3.org/TR/owl2-profiles/#Reasoning_in_OWL_2_RL_and_RDF_Graphs_using_Rules

Deductive Reasoning

- Formal manipulation of symbols representing a collection of propositions to produce representations of new propositions.

- Classical example:

Proposition	`:Man rdfs:subClassOf :Mortal .`
Proposition	`:Socrates rdf:type :Man .`
New proposition using **rdfs9**	`:Socrates rdf:type :Mortal .`

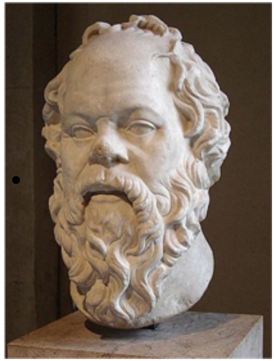


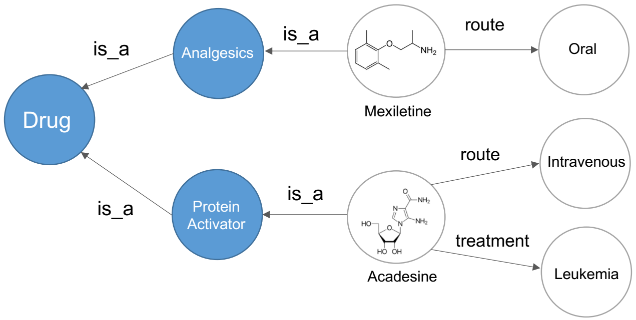
Photo from Wikipedia

- In RDF Graphs, we perform reasoning by computing new RDF triples from the consequent of rules, using the defined entailment relations.

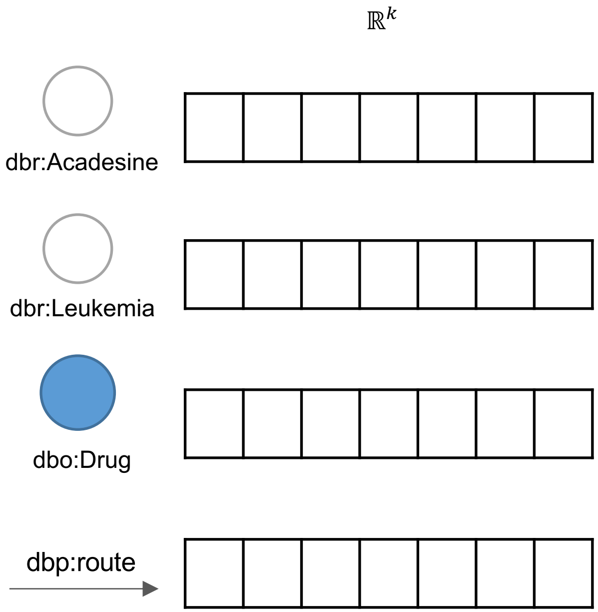
Outline

1. Introduction to Knowledge Graphs
2. Symbolic Representations of Knowledge Graphs
3. **Sub-Symbolic Representations of Knowledge Graphs**
4. The Problem of Knowledge Graph Completion
5. Conclusion and Future Work

Knowledge Graph Representations



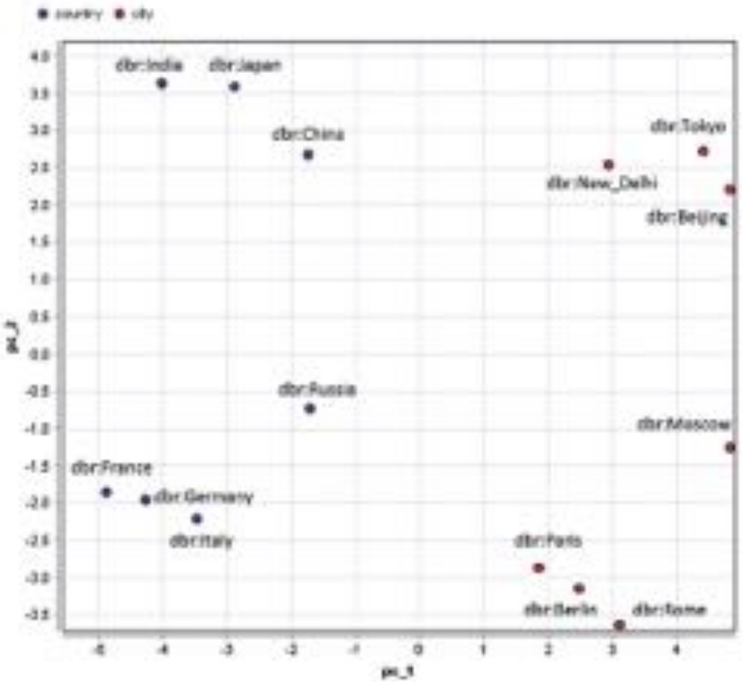
Symbolic Representation



Sub-Symbolic Representation

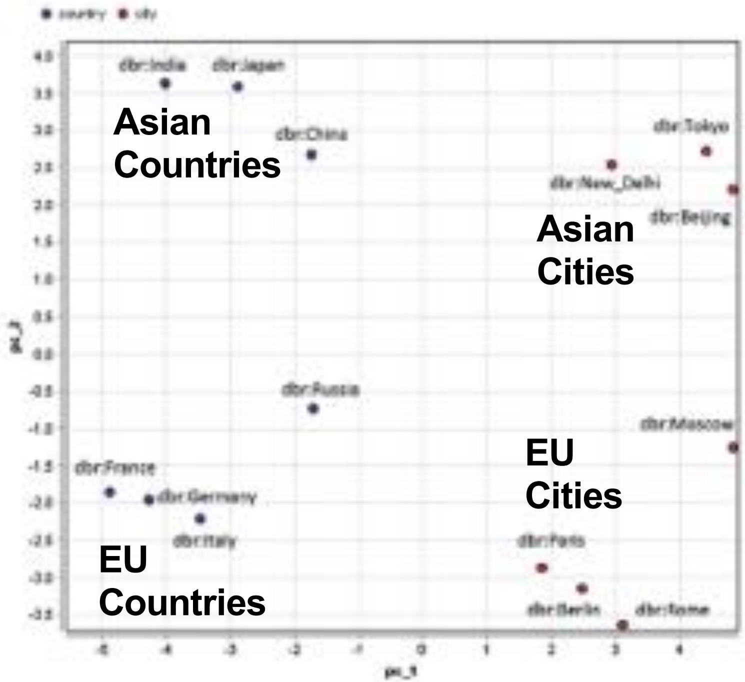
RUHR UNIVERSITÄT BOCHUM

Sub-Symbolic Representations for KGs



Allow for uncovering **hidden** patterns / associations

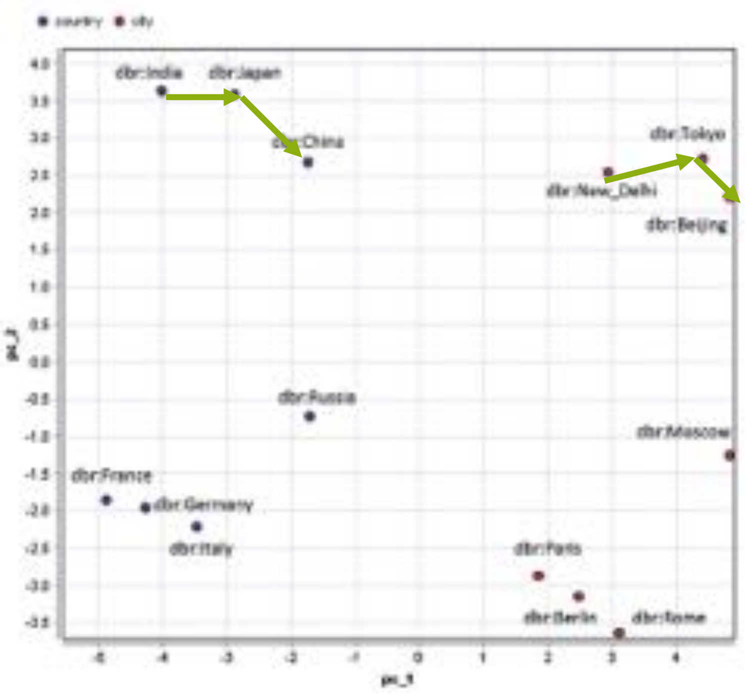
Sub-Symbolic Representations for KGs



Allow for uncovering **hidden** patterns / associations

RUHR UNIVERSITÄT BOCHUM RUB

Sub-Symbolic Representations for KGs



Allow for uncovering **hidden** patterns / associations

RUHR UNIVERSITÄT BOCHUM

Knowledge Graph Embeddings

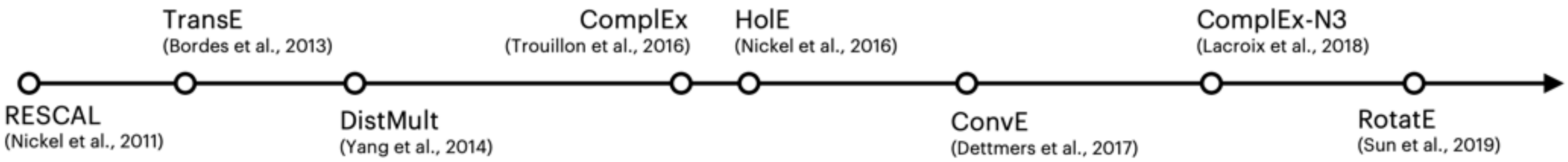
- KG embeddings represent entities (and relations) in a vector space.
- Embeddings can be computed with different **representation learning** methods:

Symbolic Representation   Representation Learning   Sub-symbolic Representation

Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

Knowledge Graph Embedding Models

Some KGE models in recent published literature:



Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

Scoring Function

- Assigns a score to a triple $t = (s, p, o)$.
  - High score = high chance that the triple t is true.

- Different types of scoring functions for KG embedding models:
  - Translation-based scoring functions
  - Factorization-based scoring functions
  - “Deeper” scoring functions

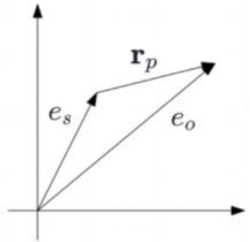
Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

Translation Based Scoring Function

- **TransE:** computes a similarity between the embedding of the subject $\boldsymbol{e}_s$ translated by the embedding of the predicate $\boldsymbol{r}_p$ and the embedding of the object e_o, using the L_1 or L_2 norm:

$$f_{TransE} = -\left\|\left(\boldsymbol{e}_s + \boldsymbol{r}_p\right) - e_o\right\|_n$$

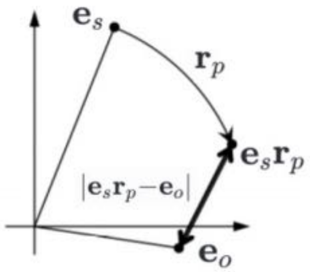
[Bordes et al. 2013]



- **RotatE:** relations modelled as rotations in a complex space.

$$f_{RotatE} = -\left\|\left(\boldsymbol{e}_s o\, \boldsymbol{r}_p\right) - e_o\right\|_n$$

[Sun et al. 2019]



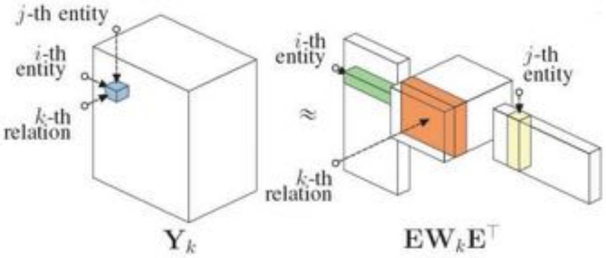
Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

Factorization-based Scoring Function

- **Rescal:** low-rank factorization with tensor product.

$$f_{RESCAL} = \boldsymbol{e}_s^T \boldsymbol{W}_r \boldsymbol{e}_0$$

[Nickel et al. 2013]



- **DistMult:** bilinear diagonal model. Dot product.

$$f_{DistMult} = \langle r_p, e_s, e_0 \rangle$$

[Yang et al. 2015]

- **ComplEx:** Complex embeddings. Extends DistMult with dot products in a complex space.

$$f_{ComplEx} = Re(\langle r_p, e_s, \overline{e_0} \rangle)$$

[Trouillon et al. 2015]

Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

“Deeper” Scoring Functions

- ConvE: reshaping + convolution.

$$f_{ConvE} = \langle \sigma\left(vec\left(g([\overline{e_s}; \overline{r_p}] * \Omega)\right)\boldsymbol{W}\right) e_o\rangle$$

[Dettmers et al. 2017]

2D reshaping

- ConvKB: convolutions and dot product.

$$f_{ConvKB} = concat\big(g([e_s, r_p, e_o]) * \Omega\big) \cdot W$$

[Nguyen et al. 2018]

- Computationally expensive!

Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

Loss Function

- **Pairwise Margin-based Hinge Loss:** Pays a penalty if the score of a positive triple is smaller than the score of a negative (synthetic) triple by margin γ.

[Bordes et al. 2013]

$$\mathcal{L}(\Theta) \sum_{t^+ \in \mathcal{G}} \sum_{t^- \in \mathcal{C}} \max(0, [\gamma + f(t^-;\Theta) - f(t^+;\Theta)])$$

Score assigned to a synthetic negative

Score assigned to a true triple

- **Negative Log-likelihood / Cross Entropy:**

[Trouillon et al. 2016]

$$\mathcal{L}(\Theta) \sum_{t \in \mathcal{G} \cup \mathcal{C}} \log(1 + \exp(-y\, f(t;\Theta)))$$

Label of the triple $y \in \{-1,1\}$

Θ denotes the parameters of the corresponding model

Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

Negative Generation

- Knowledge Graphs only contain positive statements (true statements).
- Where do negative examples (i.e., false statements) come from?
  - Synthetic Negative Generation
- Local Closed World Assumption: the KG is only locally complete.
- "Corrupted" versions of a triple as synthetic negatives:

$$\mathcal{C} = \{(\hat{s}, p, o) | \hat{s} \in \mathcal{E}\} \cup \{(s, p, \hat{o}) | \hat{o} \in \mathcal{E}\}$$

- Example:

$$\mathcal{E} = \{Mike, Liverpool, AcmeInc, George, LiverpoolFC\}$$
$$\mathcal{R} = \{bornIn, friendWith\}$$
$$t \in \mathcal{G} = (Mike\ bornIn\ Liverpool)$$

Negatives

$$(Mike, bornIn, AcmeInc)$$
$$(Mike, bornIn, LiverpoolFC)$$
$$(George, bornIn, Liverpool)$$

Knowledge Graph Embeddings: Considerations

Explainability

- It is not straightforward to understand the predictions done with sub-symbolic representations.
- This aspect is crucial in sensitive/critical use cases.

Unreliability

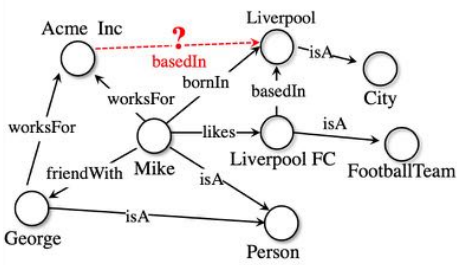
- Predictions using sub-symbolic representations are not based on logic (unlike reasoning).

Randomness

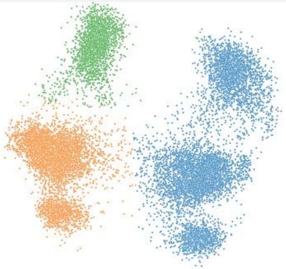
- Most representation learning techniques include random components.
- We can obtain (very) different embeddings for the same KG using the same representation learning approach.

Applications of Knowledge Graph Embeddings

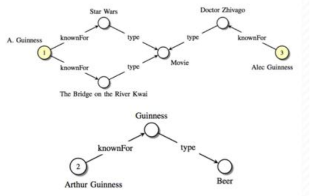
Knowledge Graph Completion



Semantic Similarity



Entity Matching



Pic from [Nickel et al. 2016a]

(and more)

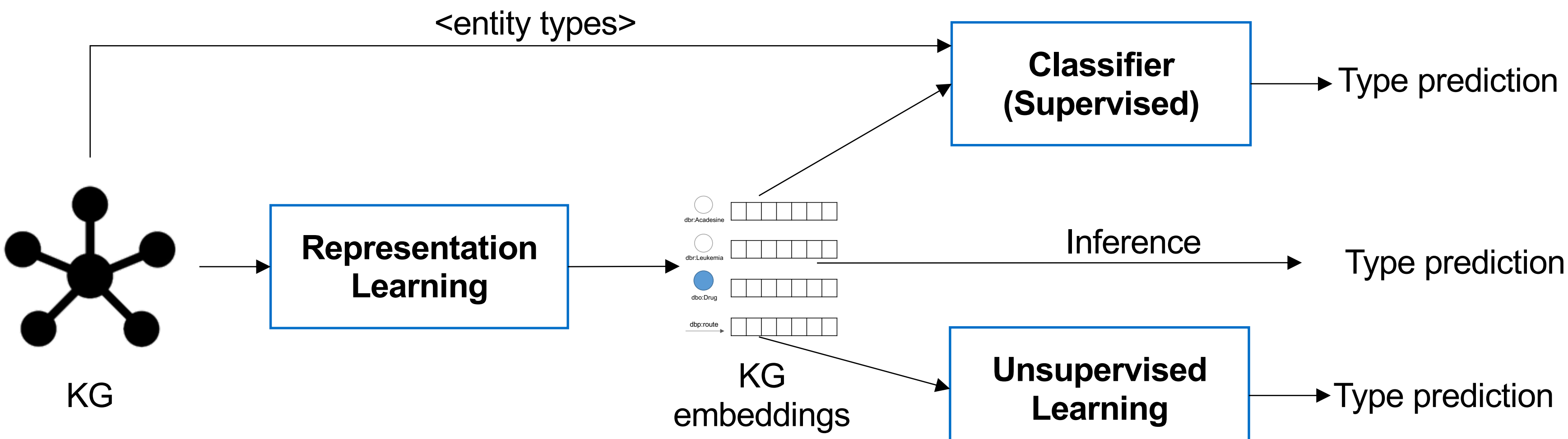
Outline

1. Introduction to Knowledge Graphs
2. Symbolic Representations of Knowledge Graphs
3. Sub-Symbolic Representations of Knowledge Graphs
4. **The Problem of Knowledge Graph Completion**
5. Conclusion and Future Work

Knowledge Graph Completion Tasks

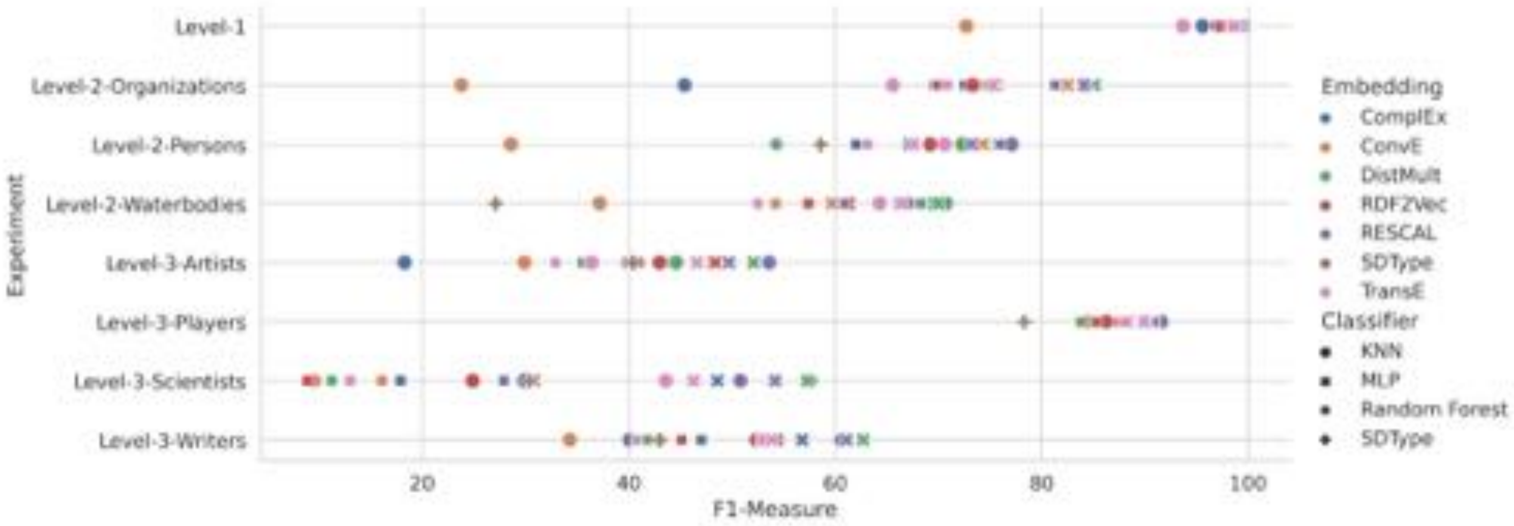
Task	Assumption	Example	Result
Triple Classification	**CWA**	(MarieCurie, occupation, Chemist)	(True, 0.95)
Tail Prediction	**CWA**	(MarieCurie, occupation, ?)	(1, Chemist, 0.95) (2, Physicist, 0.92)
Head Prediction	**CWA**	(?, occupation, Chemist)	(1, MarieCurie, 0.91) (2, PierreCurie, 0.89)
Relation Prediction	**CWA**	(MarieCurie, ?, PierreCurie)	(1, spouse, 0.90)
Entity Classification / Type Prediction	**CWA**	(MarieCurie, is_a, ?)	(1, Person, 0.92) (2, Scientist, 0.87)
Missing Relation Prediction	**OWA**	(MarieCurie, ?, *X*) *X* = existential variable	(birthPlace, 0.98) (awards, 0.80)

Type Prediction with Machine Learning



State-of-the-art: Type Prediction

Results: Type prediction is a difficult task for current KG embeddings. [Jain et al. 2021]



F1 measure for Yago3-10 classification experiments

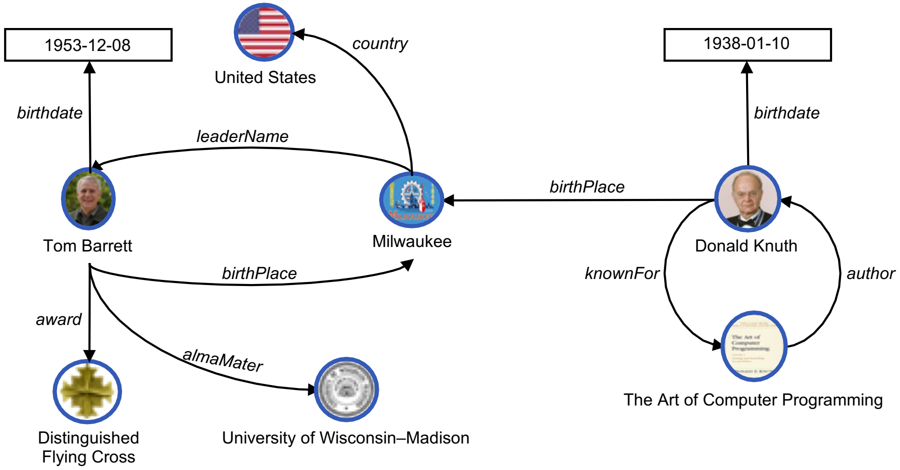
Our Approach: Ridle

Type Prediction

Our Approach: Ridle

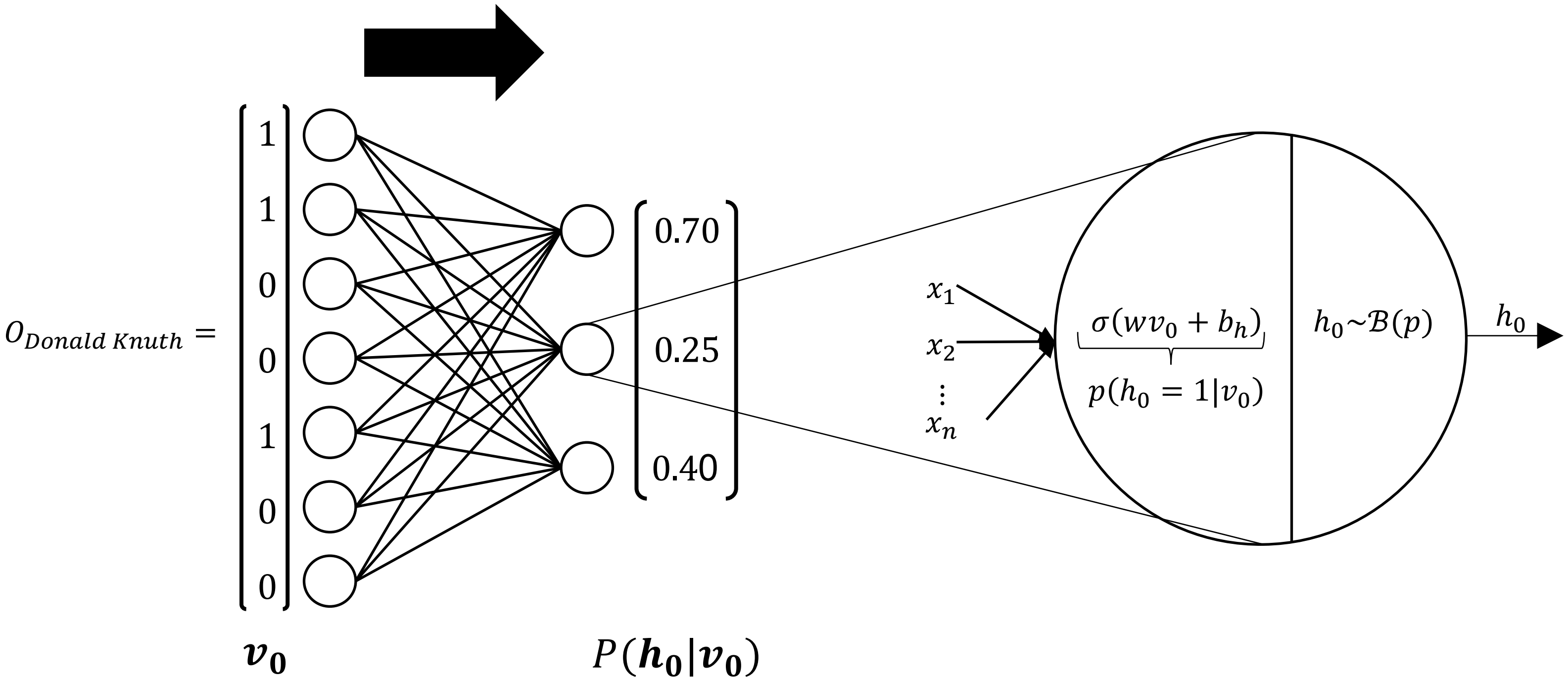
Idea: Similar entities have a similar distribution of used relations.

Estimate the unknown relation distribution by using a Restricted Boltzman Machine (RBM).

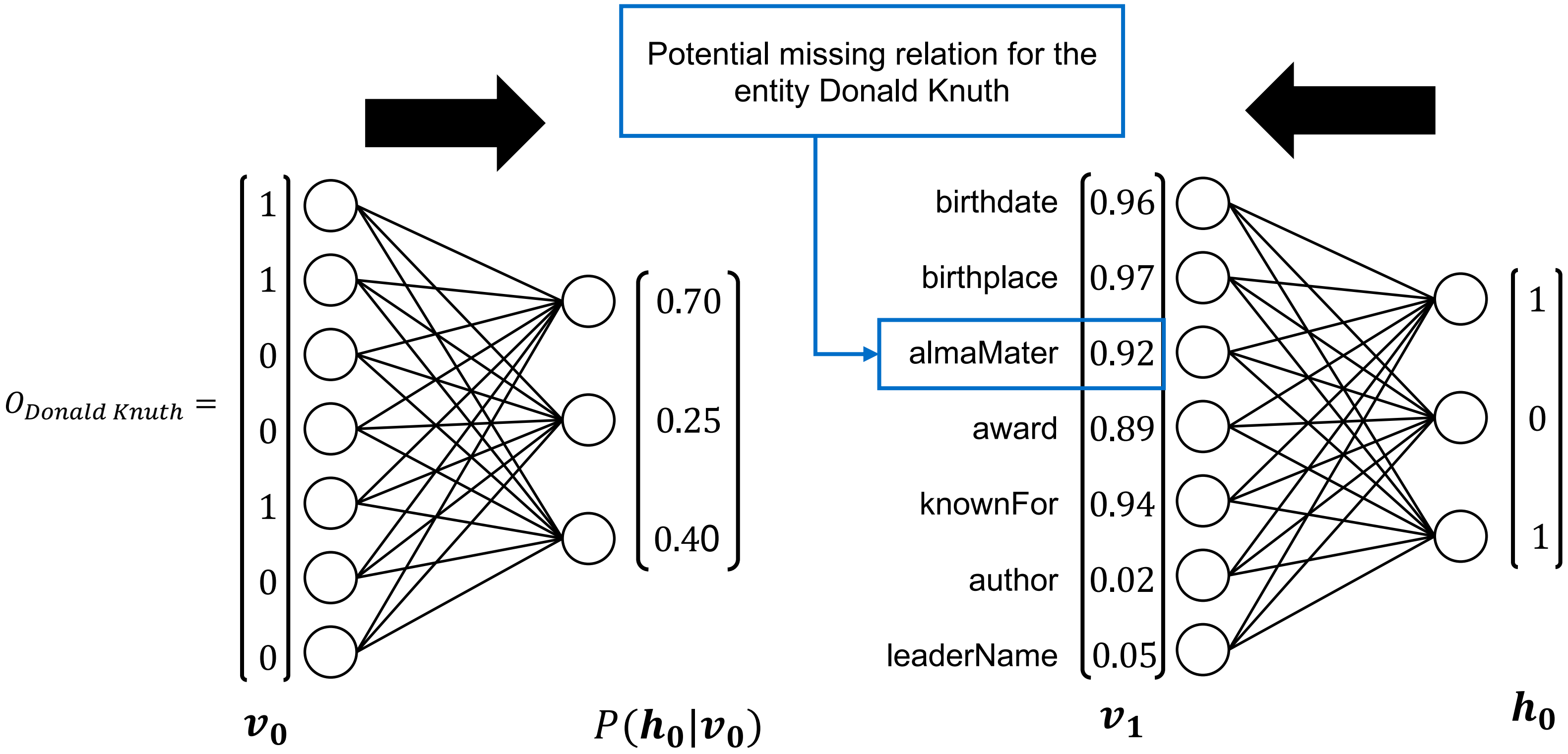


$$O_{Donald\ Knuth} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{array}{l} \text{birthdate} \\ \text{birthplace} \\ \text{almaMater} \\ \text{award} \\ \text{knownFor} \\ \text{author} \\ \text{leaderName} \end{array} \qquad O_{Tom\ Barrett} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Our Approach: Ridle



Our Approach: Ridle



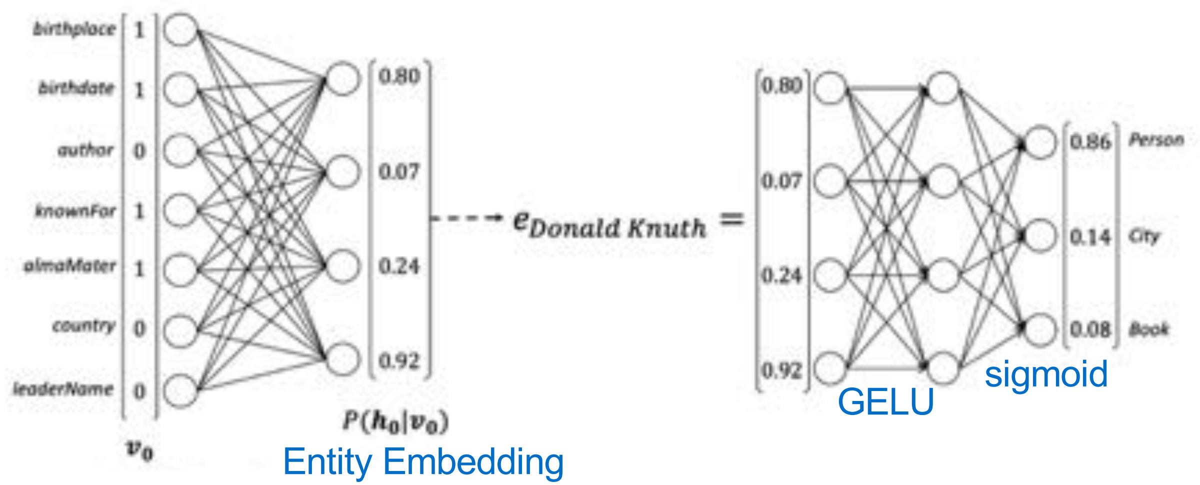
Ridle: Type Prediction

Hypothesis: Entities with similar relation distribution typically belong to the same classes.

Use the hidden layer of the RBM to train a neural network for predicting entitiy classes.

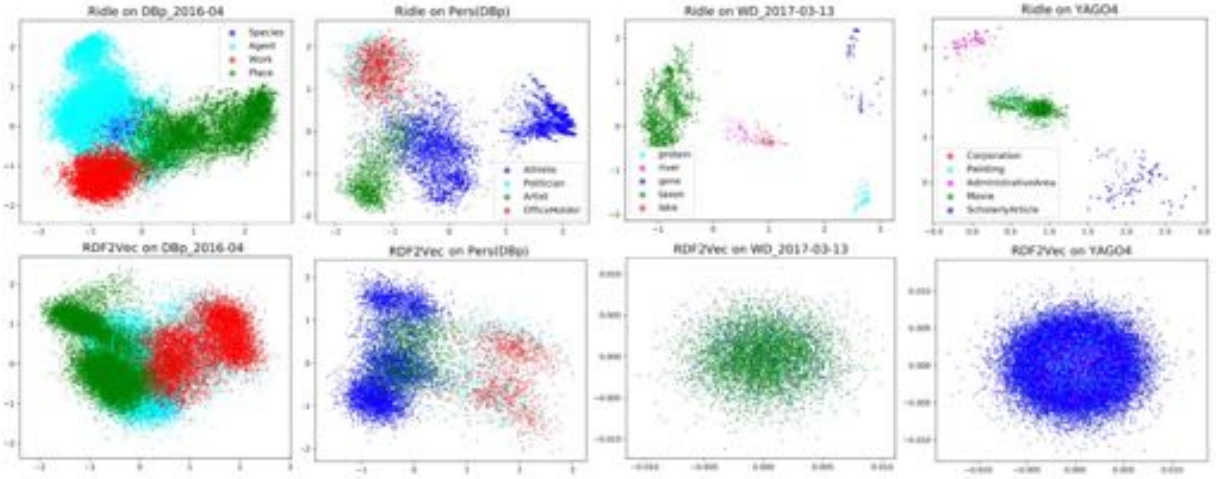
Representation Learning

Classifier (Supervised)



Experimental Results

KG	F1-Macro Ridle	SDType	RDF2Vec	RESCAL	IntE	TransE	F1-Micro Ridle	SDType	RDF2Vec	RESCAL	IntE	TransE
DBp_3.8	**.840±.01**	.224±.02	.331±.02	.370±.01	.098±.01	.376±.01	**.965±.00**	.662±.01	.000±.00	.688±.00	.002±.00	.716±.00
DBp_2016-04	**.846±.01**	.222±.01	.209±.02	.317±.02	.188±.02	.371±.02	**.968±.00**	.595±.01	.000±.00	.624±.00	.000±.00	.715±.00
WD_2017-03-13	**.805±.01**	.115±.01	.774±.01	.784±.01	.784±.01	.779±.01	.590±.01	.563±.01	.000±.00	.751±.00	.752±.00	**.801±.00**
YAGO4	**.727±.01**	.056±.00	.693±.02	.623±.01	.657±.01	.621±.01	**.965±.00**	.888±.00	.643±.00	.889±.00	.725±.00	.890±.00

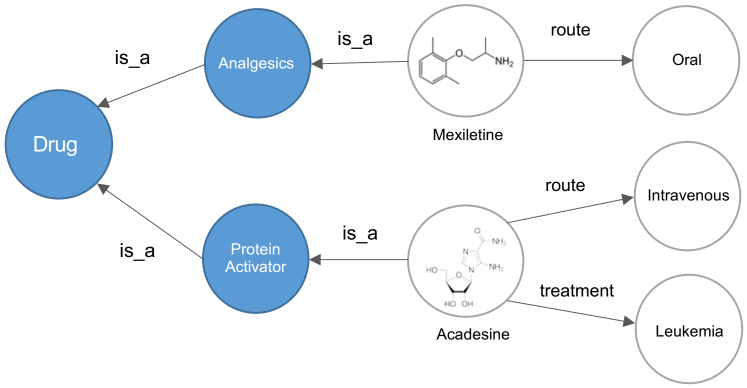


Outline

1. Introduction to Knowledge Graphs
2. Symbolic Representations of Knowledge Graphs
3. Sub-Symbolic Representations of Knowledge Graphs
4. The Problem of Knowledge Graph Completion
5. **Conclusion and Future Work**

Summary

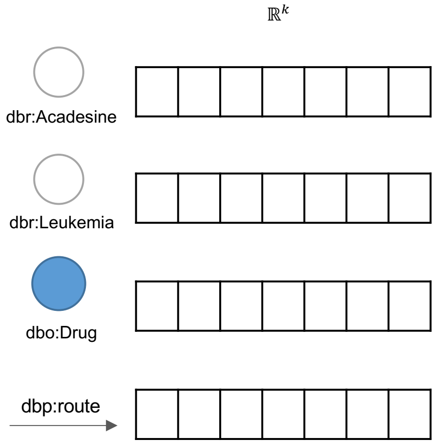
- Knowledge Graphs allow for representing interconnected statements.
  - Form a directed-labeled graph.
- Symbolic Representations of KGs:
  - Triple-based model (s, p, o).
  - Ontologies with different expressivity:
    - **RDF:** definition of properties.
    - **RDFS:** classes, hierarchies of classes and properties.
    - **OWL:** complex knowledge.
  - Suitable for entailment and reasoning.



RUHR UNIVERSITÄT BOCHUM RUB

Summary

- Sub-symbolic Representations of KGs:
  - Knowledge Graph embeddings.
  - Computed with Representation Learning approaches:
    - **Score function:** chances that a true triple belongs to the KG.
    - **Loss function:** takes into account the score function.
    - Synthetic **negative generation**.
    - **Optimization:** off-the-shelf SGD variants.
  - Suitable for knowledge graph completion.
  - Current limitations: expressivity and explainability.



RUHR UNIVERSITÄT BOCHUM RUB

Open Research Problems

More Expressive Models

Capture KG regularities and dependencies while keeping runtime/space complexity low.

Multimodal Support

Node and edge attributes, different forms of embeddings, time-awareness, uncertainty.

Beyond Link Prediction

Multi-path predictions, complex patterns.

Better Benchmarks

Fair evaluation protocols, novel datasets, including negative predictions.

Robustness & Interpretability

Techniques to dissect, investigate, explain, and protect from adversarial attacks.

Neuro-Symbolic Integration

Integrate KGE and entailment regimes to get the best of both worlds.

Slide source: Costabello, et al. Tutorial: Knowledge Graph Embeddings: From Theory to Practice. ECAI 2020.

Future Work


Sub-symbolic Representation
Neuro-symbolic KG Management
Symbolic Representation

References and Further Reading

Bordes, Antoine, et al. "Translating embeddings for modeling multi-relational data." *Neural Information Processing Systems (NIPS)*. 2013.

Bordes, Antoine, et al. "A semantic matching energy function for learning with multi-relational data." Machine Learning 94.2 (2014): 233-259.

L. Cai, B. Yan, G. Mai, K. Janowicz, R. Zhu. “TransGCN: Coupling Transformation Assumptions with Graph Convolutional Networks for Link Prediction.” KCAP 2019.

HongYun Cai, Vincent W. Zheng, Kevin Chen-Chuan Chang, A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. IEEE Trans. Knowl. Data Eng. 30(9): 1616-1637, 2018.

Dettmers, T., Minervini, P., Stenetorp, P. and Riedel, S., 2018, April. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 32, No. 1).

Jain, N., Kalo, J. C., Balke, W. T., & Krestel, R. (2021, June). Do Embeddings Actually Capture Knowledge Graph Semantics?. In European Semantic Web Conference (pp. 143-159). Springer, Cham.

Kipf, Thomas N, Welling, Max. 2016. Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016).

Lacroix, Timothée, Nicolas Usunier, and Guillaume Obozinski. "Canonical tensor decomposition for knowledge base completion." In *International Conference on Machine Learning*, pp. 2863-2872. PMLR, 2018.

References and Further Reading

D. Q. Nguyen, T. D. Nguyen, D. Q. Nguyen, and D. Phung. A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network. NAACL-HLT, 2018.

Nickel, Maximilian, Volker Tresp, and Hans-Peter Kriegel. "A Three-Way Model for Collective Learning on Multi-Relational Data." ICML. Vol. 11. 2011.

Nickel, Maximilian, Lorenzo Rosasco, and Tomaso Poggio. "Holographic embeddings of knowledge graphs." Thirtieth Aaaiconference on artificial intelligence. 2016.

Perozzi, Bryan, Rami Al-Rfou, and Steven Skiena. "Deepwalk: Online learning of social representations." 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In European Semantic Web Conference. Springer, 593–607.

Socher, Richard, et al. "Reasoning with neural tensor networks for knowledge base completion." Advances in neural information processing systems. 2013.

Sun, Z., Deng, Z.H., Nie, J.Y. and Tang, J., 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*.

Zequn Sun, JiaCheng Huang, Wei Hu, M. Chen, L. Guo, Yuzhong Qu. “TransEdge: Translating Relation-Contextualized Embeddings for Knowledge Graphs.” ISWC 2019.

References and Further Reading

Liu, Quan, et al. "Probabilistic reasoning via deep learning: Neural association models." arXiv preprint arXiv:1603.07704(2016).

Ristoski, Petar, and Heiko Paulheim. "Rdf2vec: Rdf graph embeddings for data mining." International Semantic Web Conference. Springer, Cham, 2016.

T. Trouillon, J. Welbl, S. Riedel, E. Gaussier and G. Bouchard, Complex Embeddings for Simple Link Prediction, ICML, 2016.

Weller, T. and Acosta, M., 2021, October. Predicting instance type assertions in knowledge graphs using stochastic neural networks. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management* (pp. 2111-2118).

Yang, Bishan, et al. "Embedding entities and relations for learning and inference in knowledge bases." arXiv preprint arXiv:1412.6575 (2014).

Thank you!

Contact: maribel.acosta@rub.de

Twitter: @maribelacosta