What is a Knowledge Graph?

Mikhail Galkin

Postdoc @ Mila & McGill

Outline

- On the definition & representation
- Part I: Symbolic
  - Logical Foundations
  - Databases & Querying
  - KG Construction
- Part II: Vector
  - NLP
  - KG Embeddings
  - Graph ML

On the definition of a Knowledge Graph

Given entities E, relations R, KG is a directed multi-relational graph G that comprises triples (s, p, o)

$$\mathcal{G} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$$
$$(s, p, o) \in \mathcal{G}$$

* describes entities and relations
* defines a schema
* interrelating arbitrary entities
* various topical domains

"Abstract schema and instances"

"Every RDF / LPG / RDF* graph is a knowledge graph"

On the definition of a Knowledge Graph

Given entities E, relations R, KG is a directed multi-relational graph G that comprises triples (s, p, o)

$$\mathcal{G} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$$

$$(s, p, o) \in \mathcal{G}$$

* describes entities and relations
* defines a schema
* interrelating arbitrary entities
* various topical domains

“Every RDF / LPG / RDF* graph is a knowledge graph”

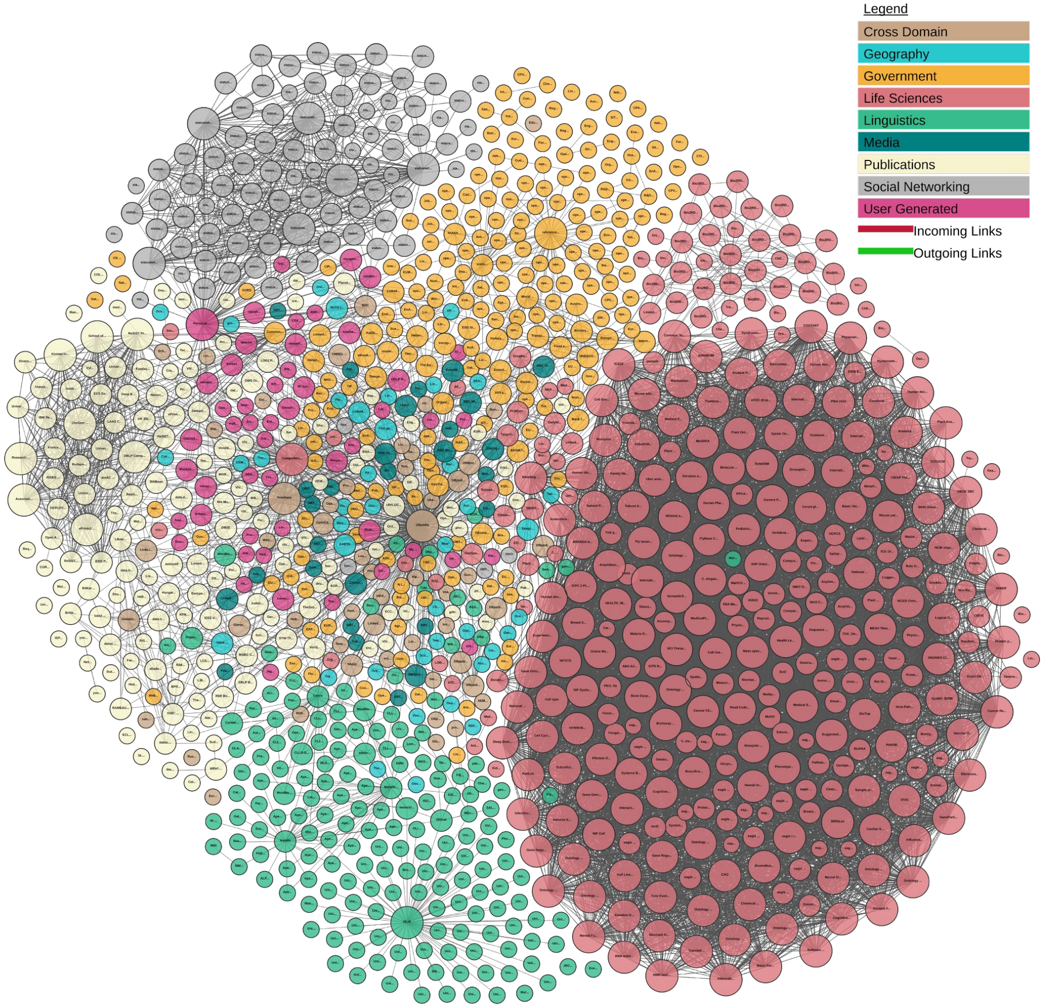
“Abstract schema and instances”

Graph-structured world model

World models?

Entities and relations define our **domain of discourse**

How to encode it?



Source: https://lod-cloud.net/

On representation of Knowledge Graphs

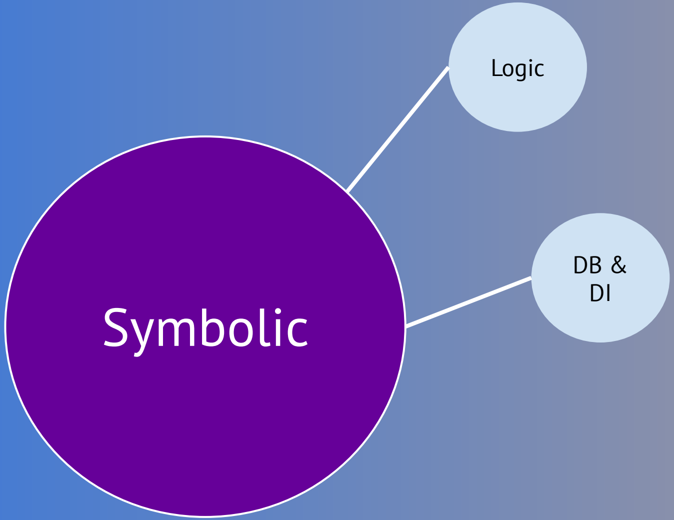
Symbolic

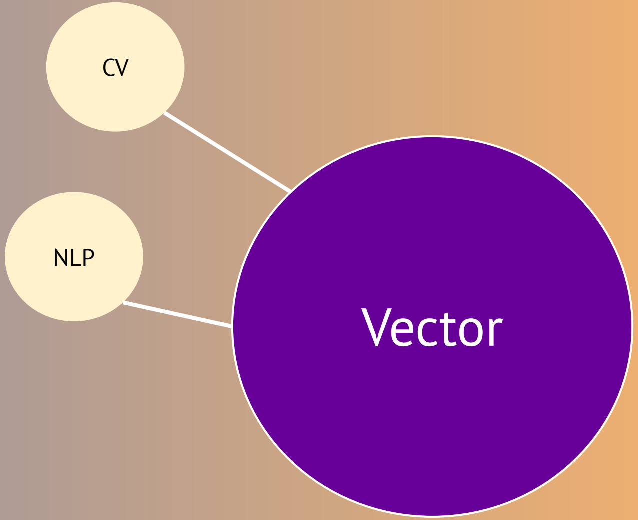
Vector

```
s,p,o
p(s,o)
(s,p,o)
```

$$s, p, o \in \mathbb{R}^d$$

On representation of Knowledge Graphs





Open-world assumption

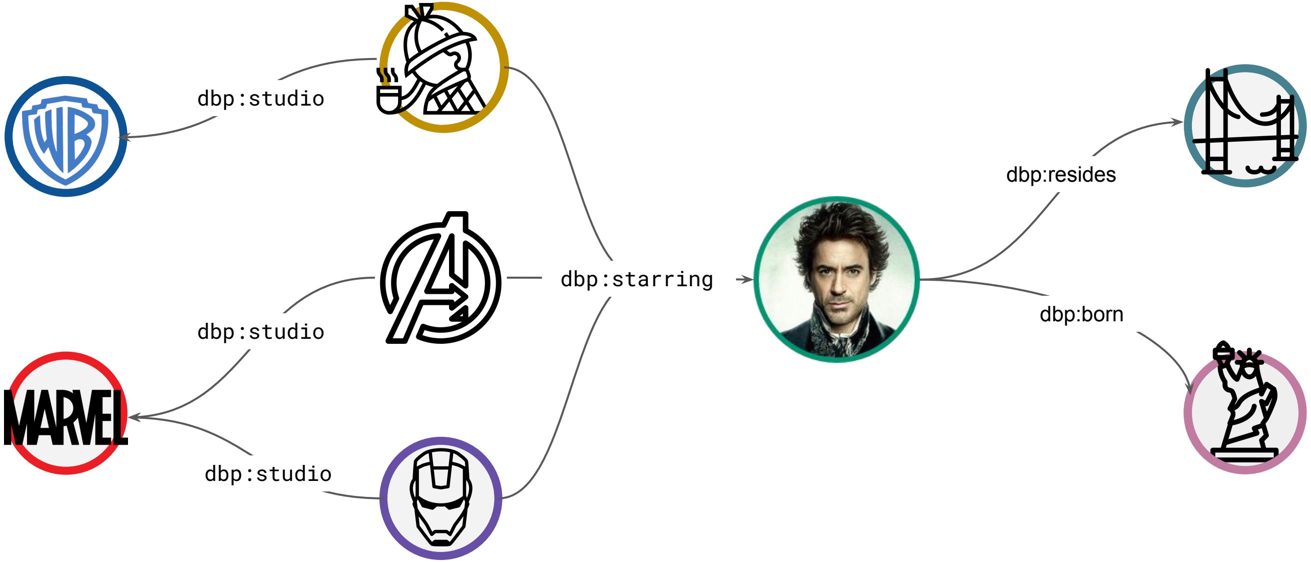
Closed-world assumption

Temporal / evolving

s,p,o
p(s,o)
(s,p,o)

$$s, p, o \in \mathbb{R}^d$$

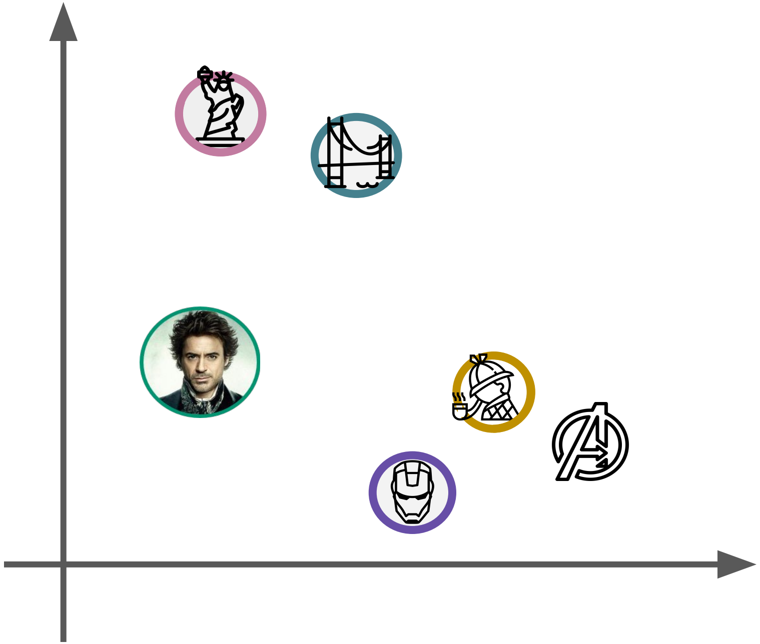
Symbolic: Triples

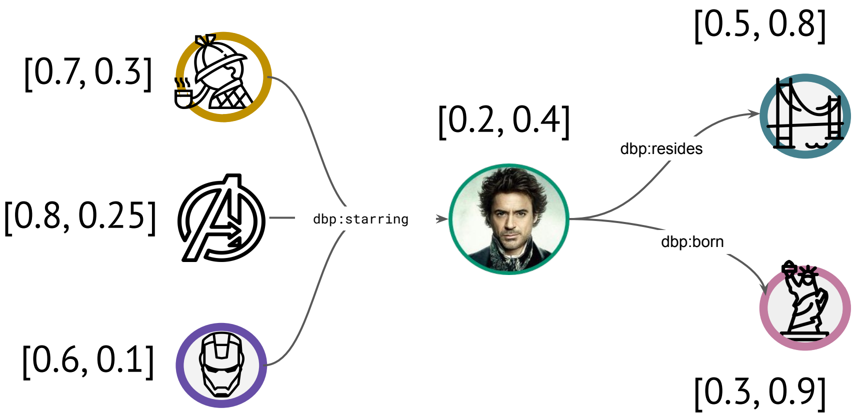


RDJ	dbp:resides	SF .	Avengers	dbp:studio	Marvel .
RDJ	dbp:born	NY .	Avengers	dbp:starring	RDJ .
Sherlock_Holmes	dbp:studio	WB .	Iron_Man	dbp:studio	Marvel .
Sherlock_Holmes	dbp:starring	RDJ.	Iron_Man	dbp:starring	RDJ .

Vector: Embeddings

$$E \in \mathbb{R}^{N_e \times d}$$
$$R \in \mathbb{R}^{N_r \times d}$$





Part I: Symbolic

History

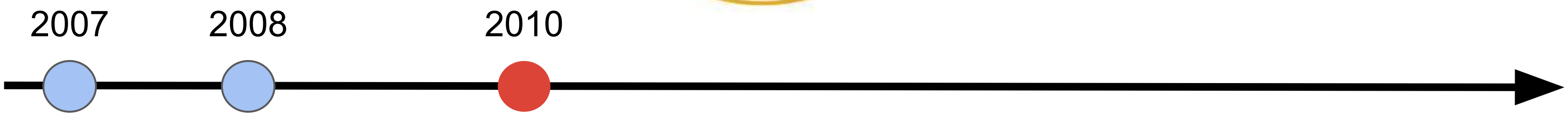
2007

- Derived from parsing Wikipedia infoboxes
- 6B+ facts
- The first de-facto standard for creating and publishing KGs

History

- Wikipedia + categories from WordNet
- 120M+ facts

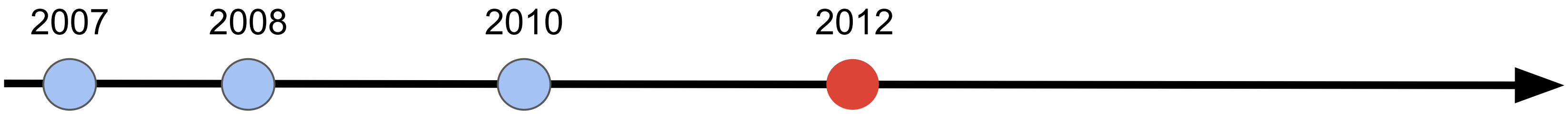
History



- NELL - Never Ending Language Learner
- Automatic facts from parsing web pages
- 15B+ facts

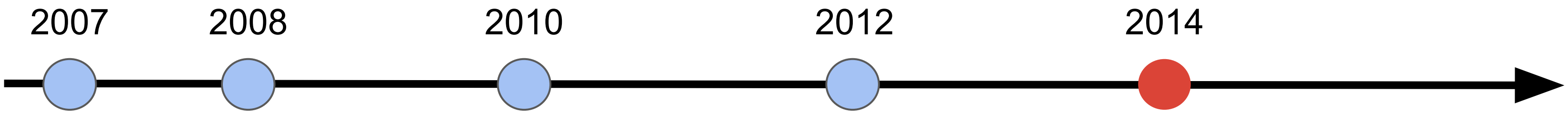
http://nell-ld.telecom-st-etienne.fr/

History



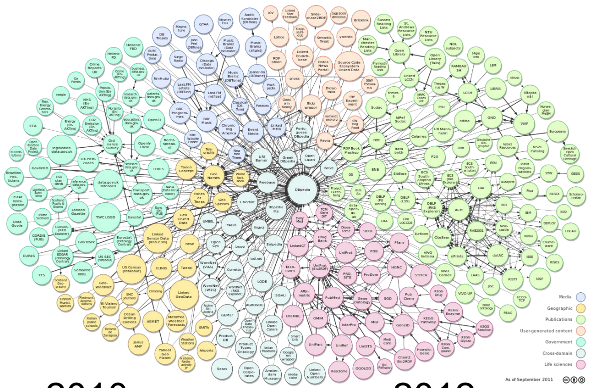
- The source of facts for Wiki infoboxes
- Flexible schema + qualifiers
- 100M+ entities, 7B+ statements
- Big Tech contributes to Wikidata

History



- Google Knowledge Graph - based on the acquired Freebase (2007)
- Common knowledge + user-specific info
- Everyone else started to want “my own KG” after that

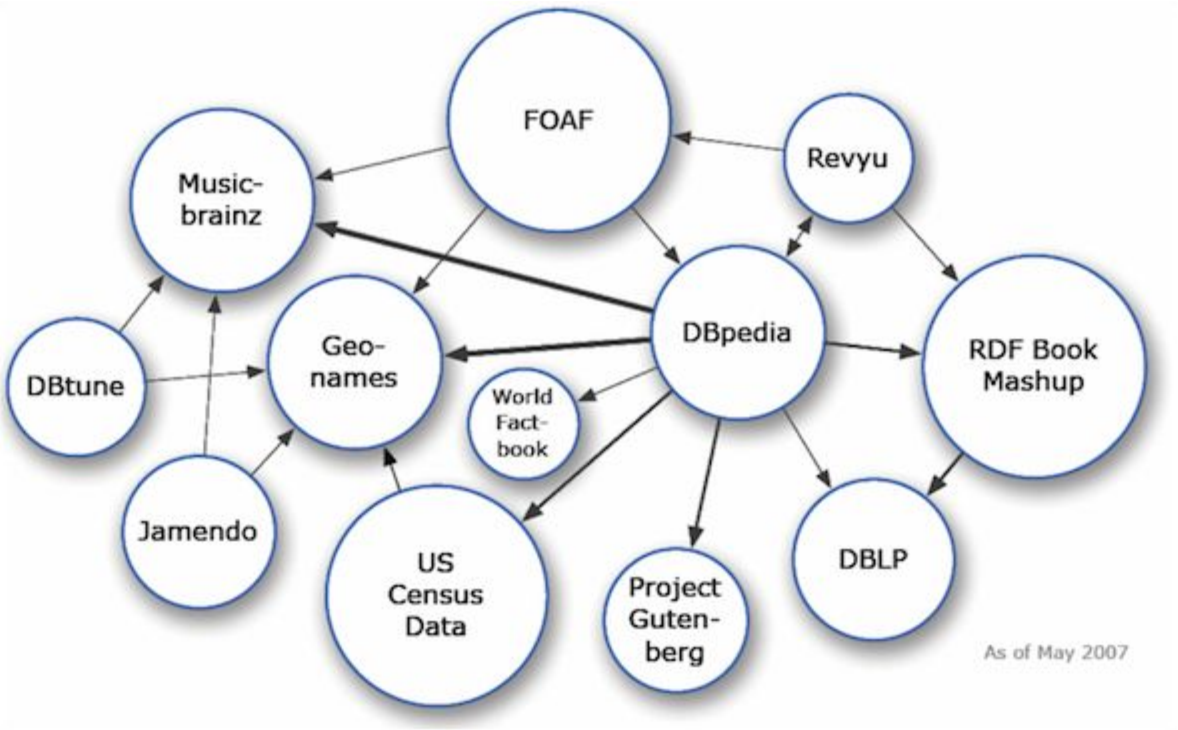
History



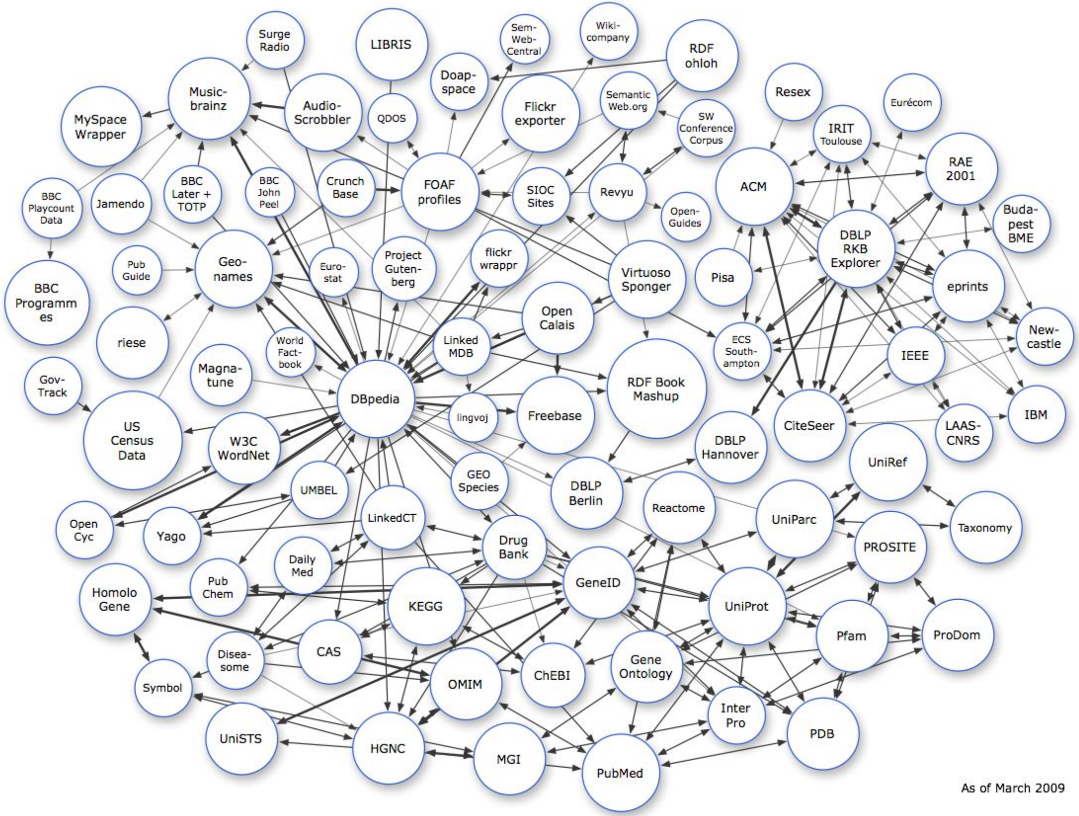
2007 2008 2010 2012 2014 2021

- Open and Linked KGs
- Domain-specific KGs in various domains
- Personal KGs
- … many more

Linked Open Data - 2007

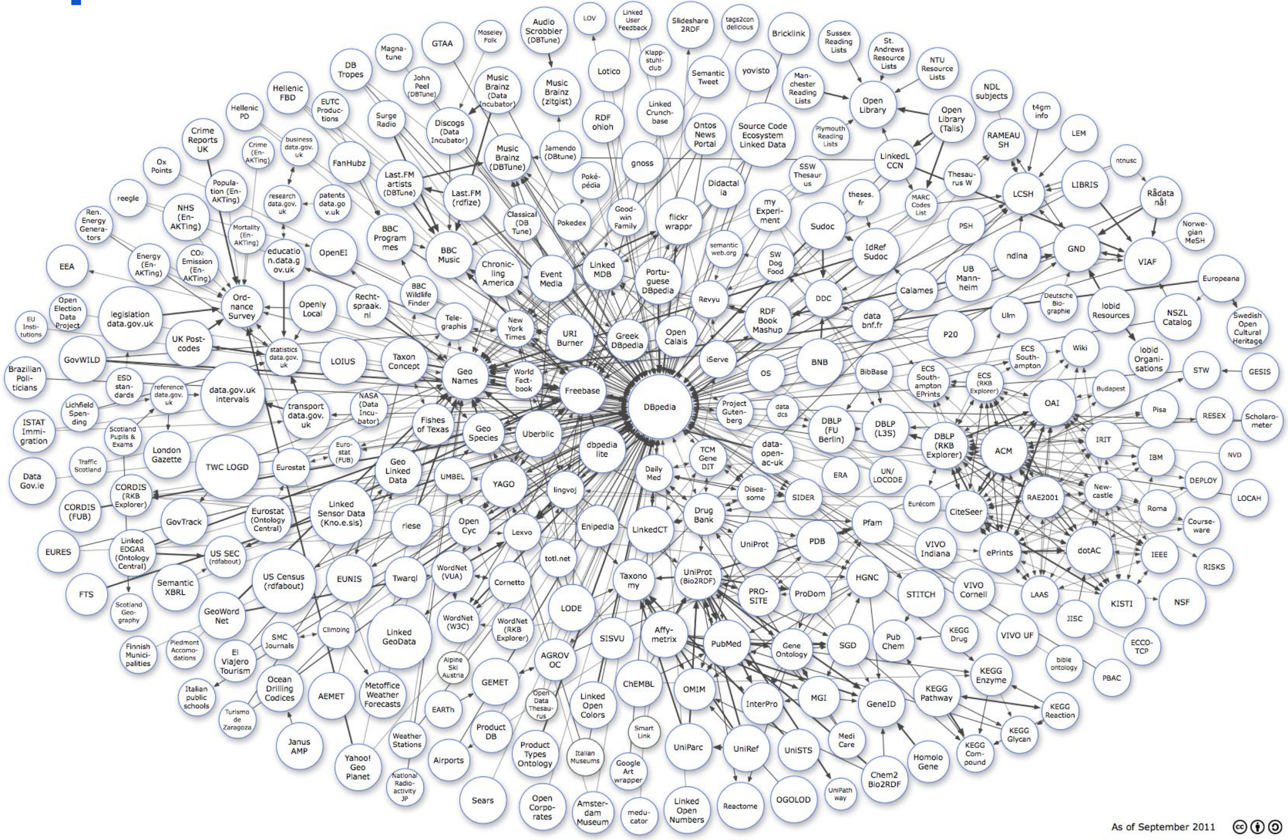


Linked Open Data - 2009

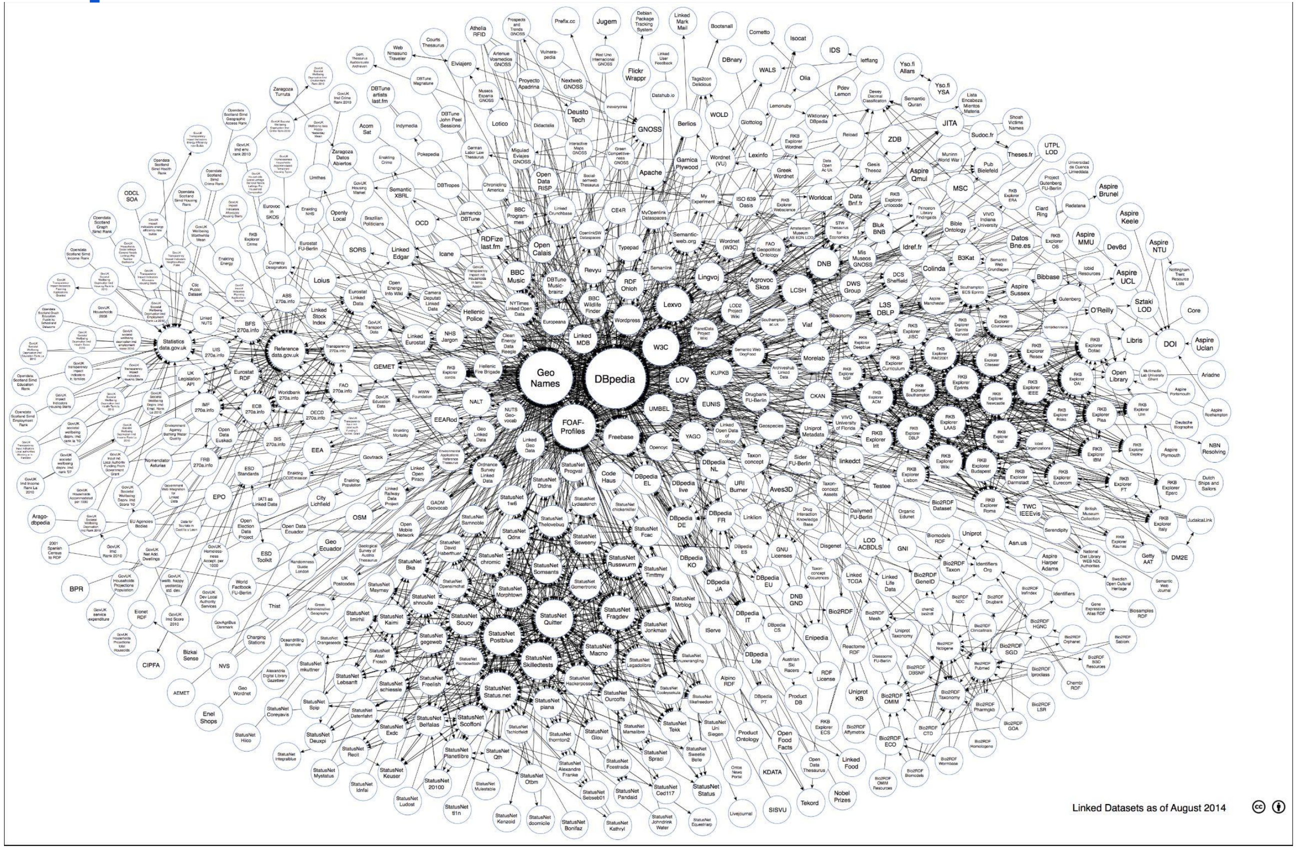


26.10.2021

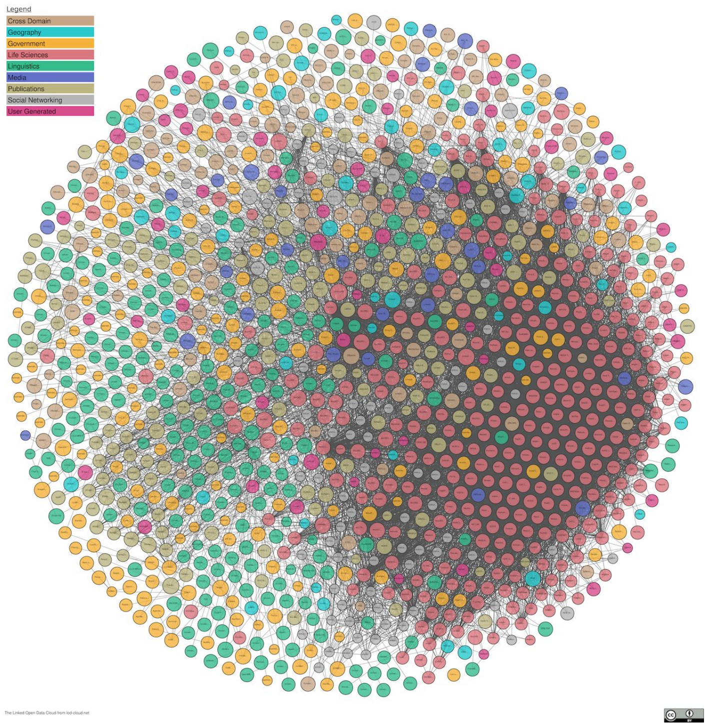
Linked Open Data - 2011



Linked Open Data - 2014



Linked Open Data - 2020



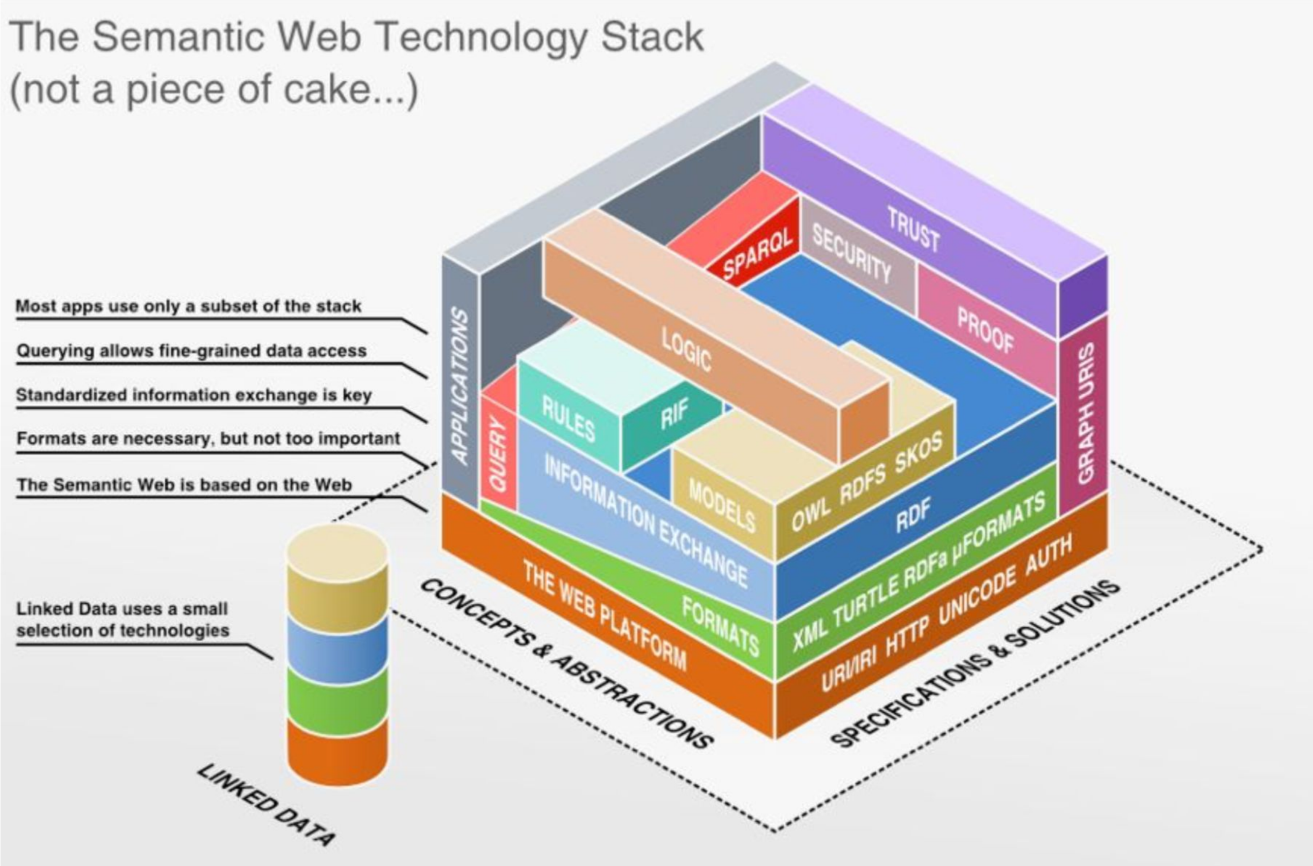
Part I: Symbolic

Logical Foundations

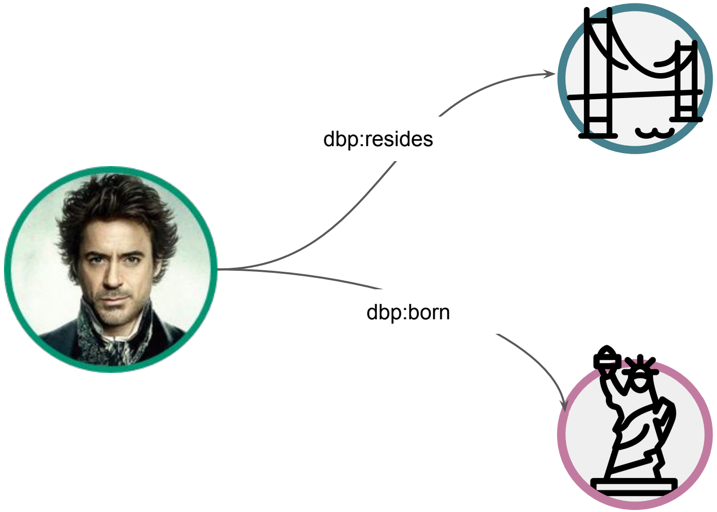
Semantic Web Layer Cake

A stack of standards for KGs

1. Web Platform
2. Serialization
3. Modeling
4. Complex Logic
5. Querying
6. Interfaces



Resource Description Framework (RDF)

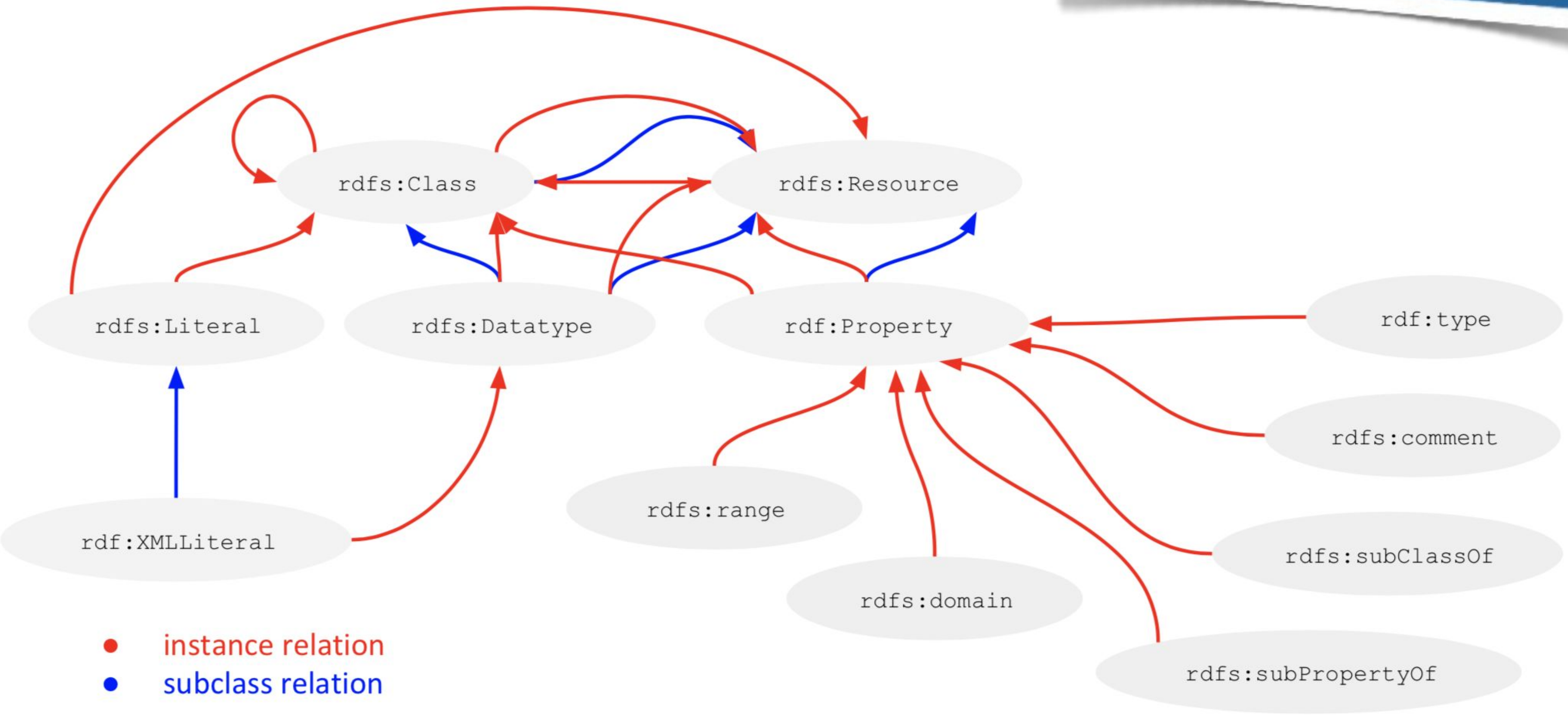


1. Facts as triples
2. All entities and relations are (ideally) URIs

http://example.com/RDJ http://dbpedia.org/resides http://dbpedia.org/resource/SanFrancisco

ex:RDJ dbr:resides dbp:SanFrancisco

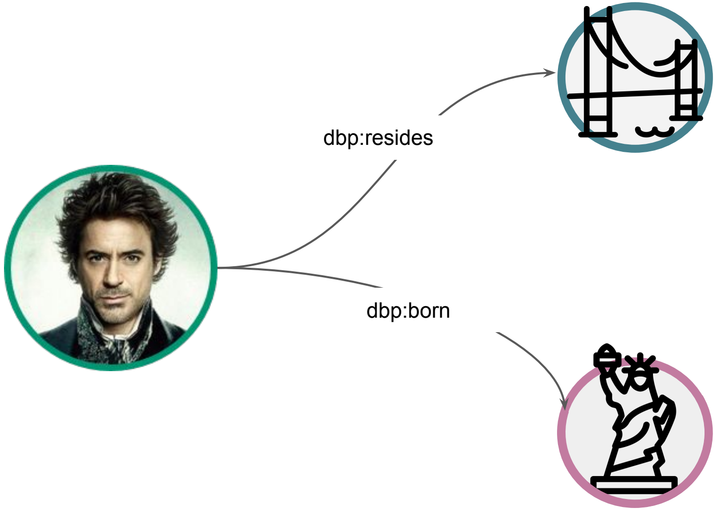
RDFS Language Model



- instance relation
- subclass relation

http://www.w3.org/TR/rdf-schema/#ch_classes

RDF + RDFS



ex:RDJ	**rdf:type**	ex:Person
ex:NewYork	**rdf:type**	ex:City

RDF + RDFS Class Hierarchies

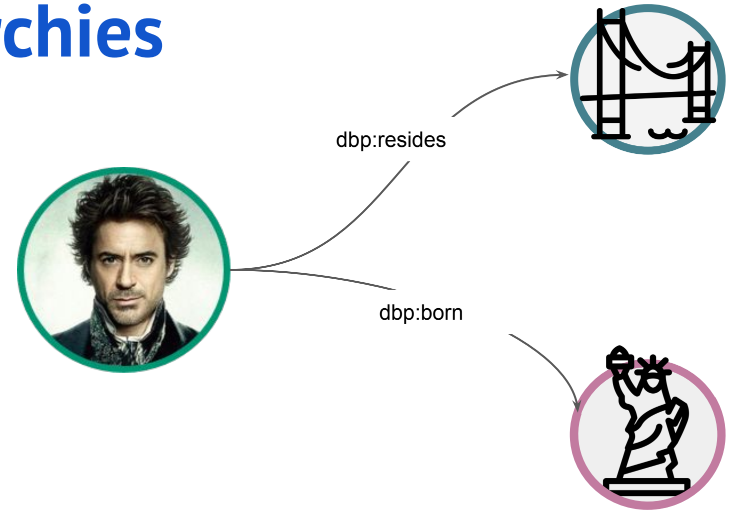
RDFS Vocabulary

rdfs:Class

rdf:Property
rdfs:range
rdfs:domain

rdfs:subClassOf
rdfs:subPropertyOf

rdfs:label
rdfs:comment
rdfs:seeAlso
rdfs:isDefinedBy



ex:RDJ	rdf:type	ex:Actor
ex:Actor	rdfs:subClassOf	ex:Person
ex:Person	rdfs:subClassOf	ex:Mammal

RDF + RDFS Property Ranges & Domains

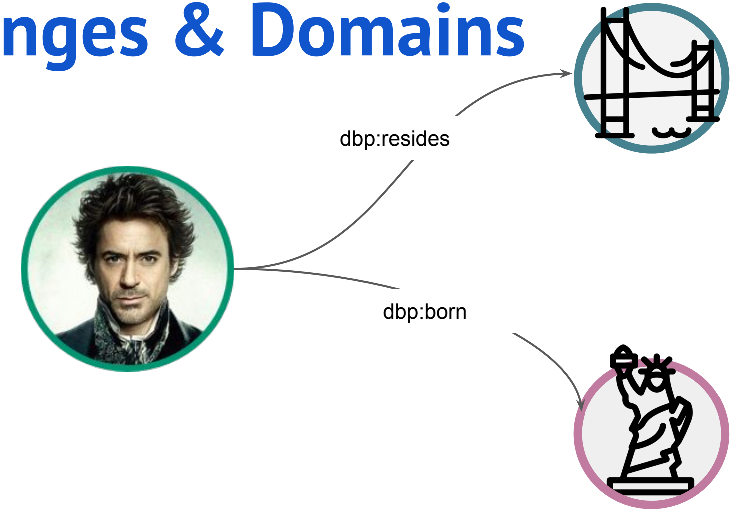
RDFS Vocabulary

rdfs:Class

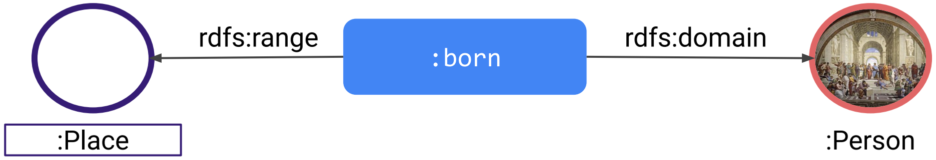
rdf:Property
rdfs:range
rdfs:domain

rdfs:subClassOf
rdfs:subPropertyOf

rdfs:label
rdfs:comment
rdfs:seeAlso
rdfs:isDefinedBy



dbp:born	rdfs:domain	ex:Person
dbp:born	rdfs:range	ex:Place



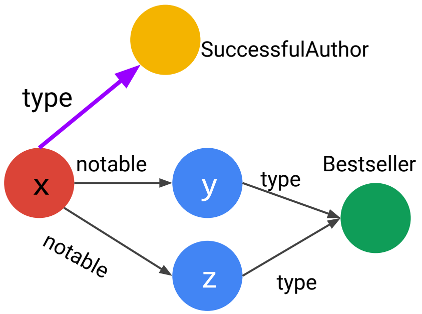
More Logics: OWL

Based on logical formalisms, e.g., Description Logics (DL)

schema, ontology, theory	TBox	$\texttt{SuccessfulAuthor} \sqsubseteq \geq 1\ \texttt{notableWork.Bestseller}$
instances, facts, assertions	ABox	$\texttt{SuccessfulAuthor(StanLee)}$
restrictions, constraints	RBox	$\texttt{notableWork} \sqsubseteq \texttt{created}$

Logically consistent collection of axioms

More Logics: OWL

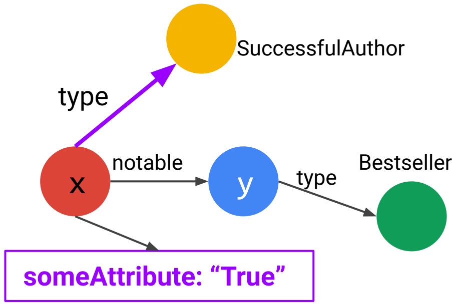


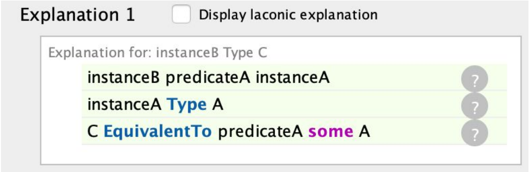
eg, Qualified Restrictions

$\text{SuccessfulAuthor} \sqsubseteq {\geq}1\ \text{notableWork.Bestseller}$

```
:SuccessfulAuthor a owl:Class ;
    rdfs:subClassOf [
        a owl:Restriction;
        owl:onProperty :notableWork;
        owl:minQualifiedCardinality 1;
        owl:onClass :Bestseller ] .
```

Reasoners





- Input: RDF/OWL graph
  Output: RDF/OWL graph with new facts (assertions)
  - New node attributes
  - New edges between nodes
- Every fact can be explained (most ML models can't do that)
- Inference time grows very fast with graph size
- 🔥 research: speed of ML reasoning + explainability of rule-based

Knowledge Graph

TBox (Terminology Box)
Data schema
Ontology

Онтология:

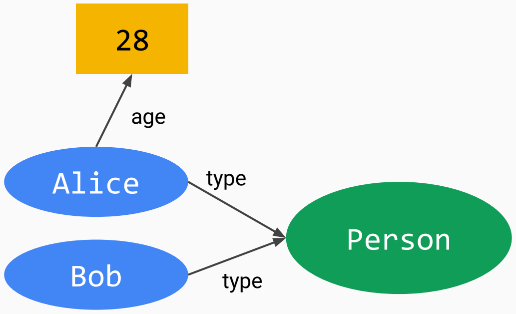
➔ Formal model curated by experts
➔ Often created & maintained manually

ABox (Assertion Box)
Actual “**content**”
Triples, edges between entities

➔ Often created from existing sources using **semantic data integration**

Open World Assumption

- Incomplete picture of the world
- Everything not explicitly stated - *possibly* true
- Extendable by design



Closed World Assumption

- Explicitly state everything about the world
- Everything not True == False
- Source of truth
  - Column in DB
  - Object field
  - Frame slot

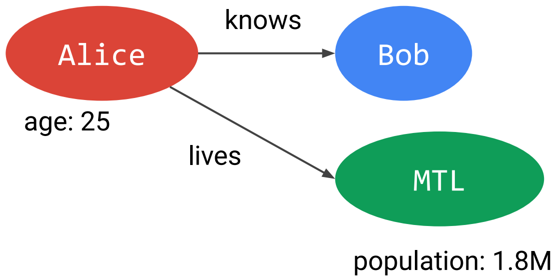
Person	Age
Alice	28
Bob	N/A

Part I: Symbolic

Graph Databases & Querying

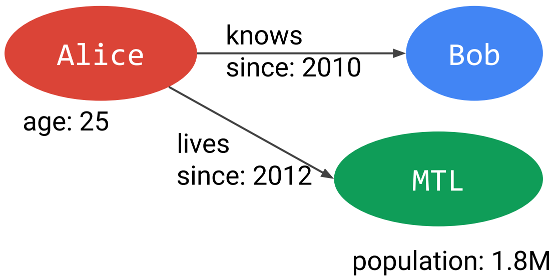
Graph DBs: RDF vs LPG

RDF



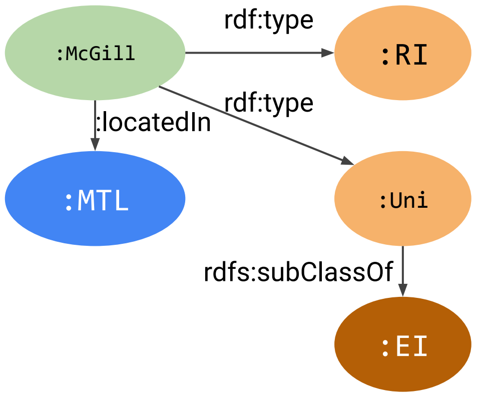
- Query Language: SPARQL
- RDFS/OWL properties of predicates
- Semantic scheme (w/ ontologies)
- Logical inference

LPG (Labeled Property Graph)



- QLs Cypher, Gremlin, GraphQL
- Any properties of predicates
- Non-semantic scheme
- No logical inference

SPARQL Query Structure




```
:McGill       rdf:type          :University .
:McGill       rdf:type          :Research_Institution .
:McGill       :locatedIn        :MTL .
:University   rdfs:subClassOf   :Educational_Institution .
```

SPARQL Query Structure

Prefixes

Query Type

Projected Variables

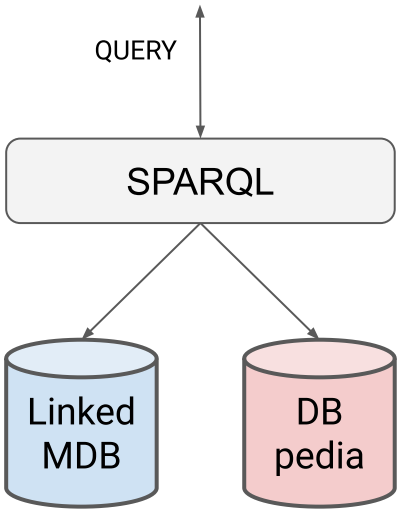
Graph Source

BGP

Modifiers

```
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

SELECT ?type ?city

FROM <named_graph>

WHERE {
   ?s     rdf:type            ?type .
   ?s     :locatedIn          ?city .
   ?type rdfs:subClassOf      ?num .
}

ORDER BY <> LIMIT <num> OFFSET <num>
```

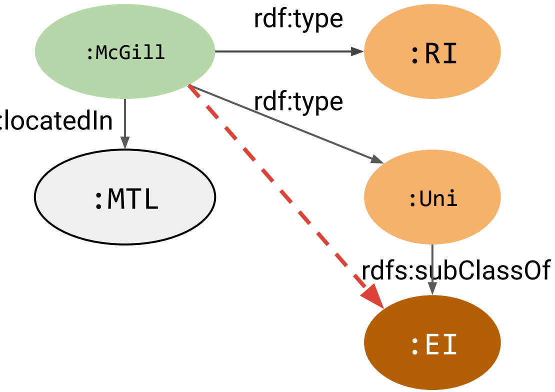
SPARQL 1.1 - Federated Querying



- Federated queries - query other endpoints within a query using **SERVICE**

```
SELECT ?film ?label ?subject WHERE {
    SERVICE <http://data.linkedmdb.org/sparql> {
        ?movie rdf:type movie:film .
        ?movie rdfs:label ?label .
        ?movie owl:sameAs ?dbpediaLink
        FILTER (regex(str(?dbpediaLink), “dbpedia”))
    }
    SERVICE <http://dbpedia.org/sparql> {
        ?dbpediaLink dct:subject ?subject .
    }
}
```

Advanced SPARQL - Reasoning




```
:McGill   rdf:type             :University .
:McGill   rdf:type             :Research_Institution .
:McGill   :locatedIn           :MTL .
:University   rdfs:subClassOf   :Educational_Institution .
(:McGill  rdf:type             :Educational_Institution .)
```

```
SELECT ?t WHERE {
  :McGill a ?t }
```

- Standard SPARQL - no reasoning, all triples have to be **materialized**
- Some Graph DBMS do allow for reasoning (inferring new triples in memory)
  - RDFS (subClassOf, range, domain)
  - OWL 2 RL / QL
  - SWRL
  - owl:sameAs

HOW TO STORE RDF DATA ?

https://commons.wikimedia.org/wiki/File:A_view_of_the_server_room_at_The_National_Archives.jpg

RDF & Graph Databases

[Faye et al. A survey of RDF storage approaches, 2012]

- Native (NoSQL) vs Non-native (SQL)
- RDF vs LPG

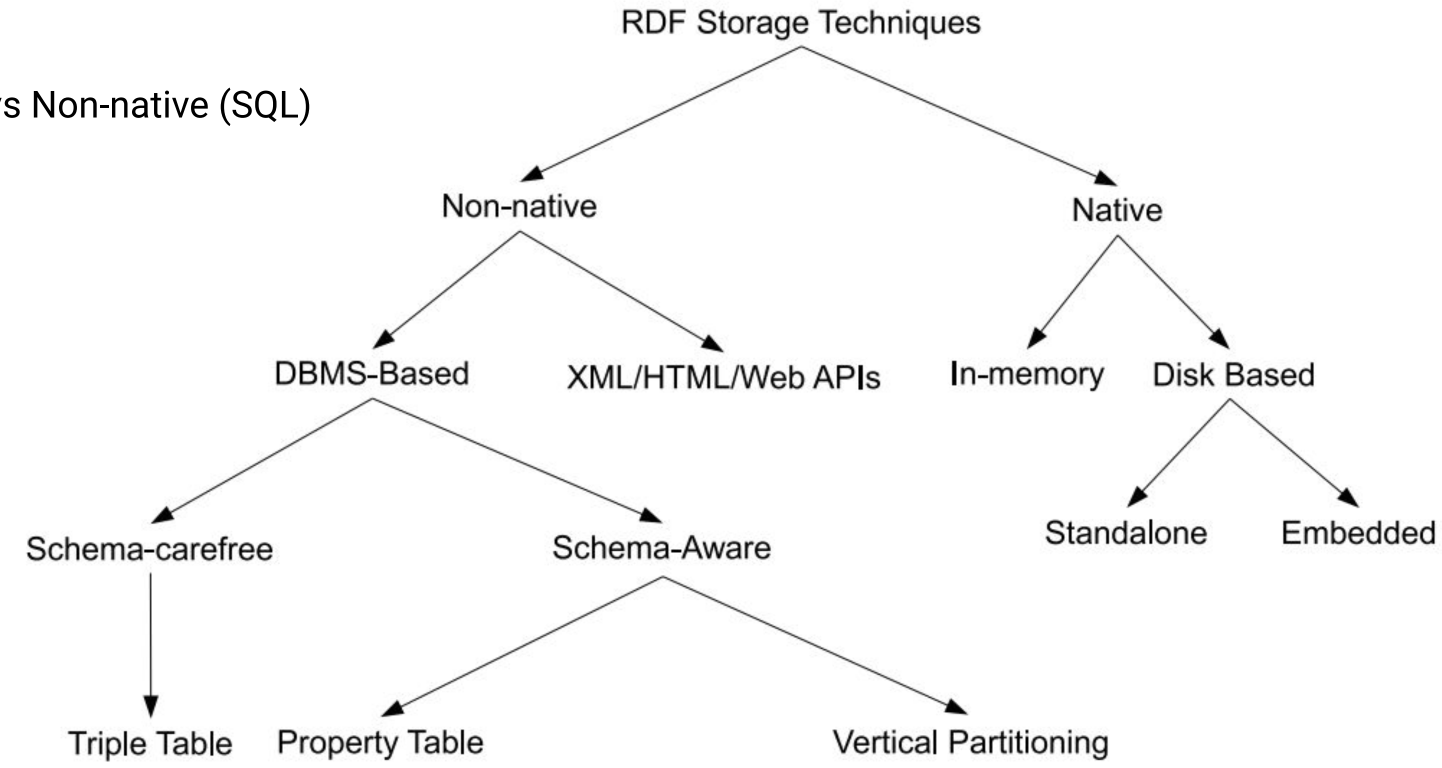
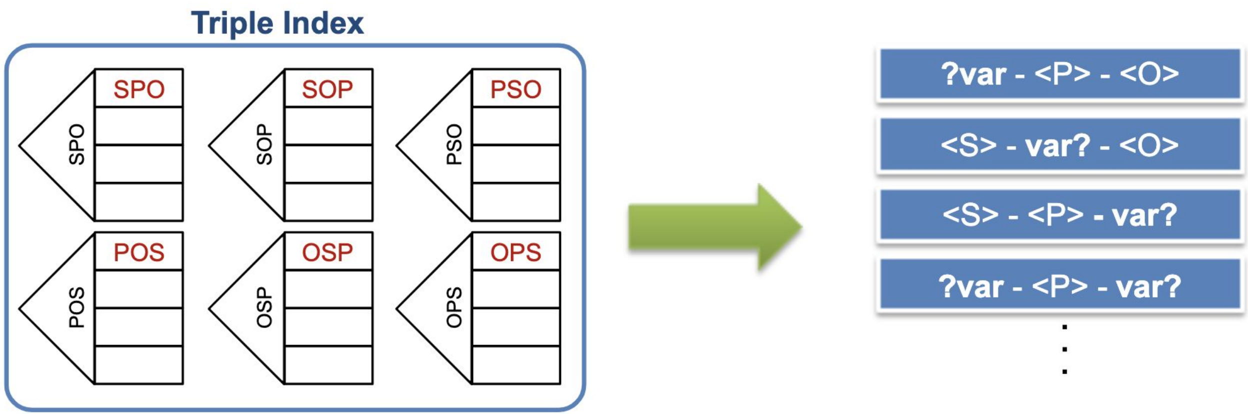


Figure 2. *A classification of RDF data storage approaches*

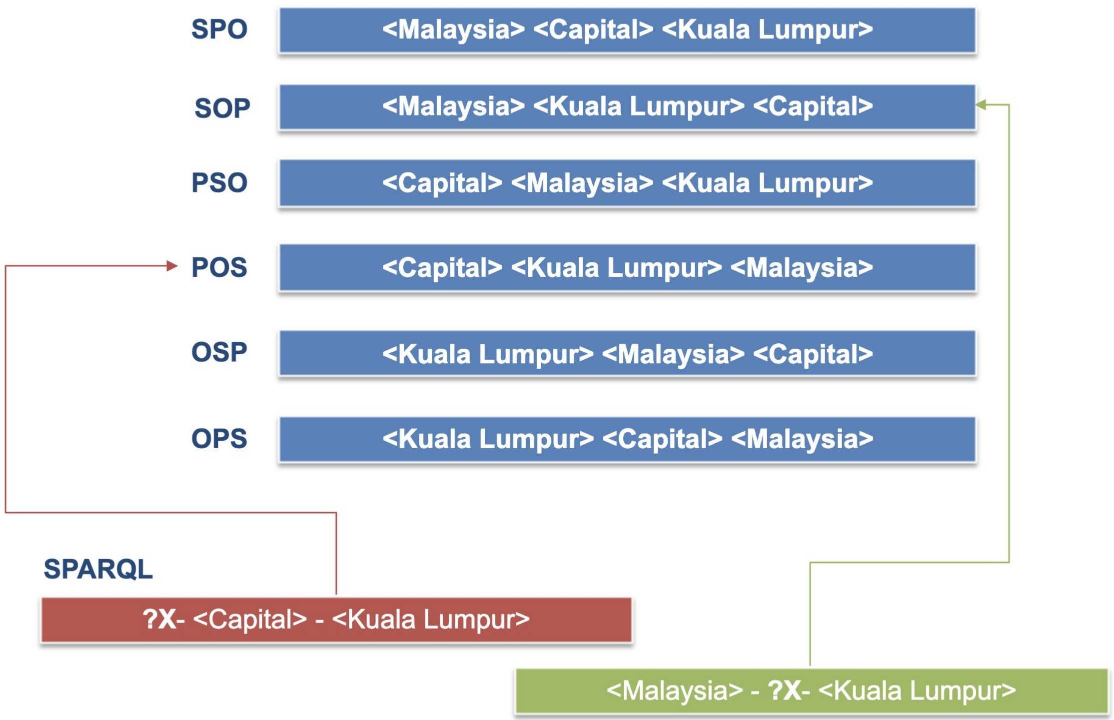
RDF Databases - Native - B+ Trees - RDF-3X

- Six separate indexes
  - (SPO, SOP, OSP, OPS, PSO, POS)
  - Stored in the leaf pages of the clustered B+ tree



[Neumann et al. RDF-3X: a RISC-style engine for RDF, 2008]

RDF Databases - Native - B+ Trees - RDF-3X



[Neumann et al. RDF-3X: a RISC-style engine for RDF, 2008]

RDF Databases - Native - B+ Trees - RDF-3X

- Store collation order
  - Neighboring indexes are very similar
  - Stores the change between triples

S	P	O
0	1	2
0	1	3
0	1	5

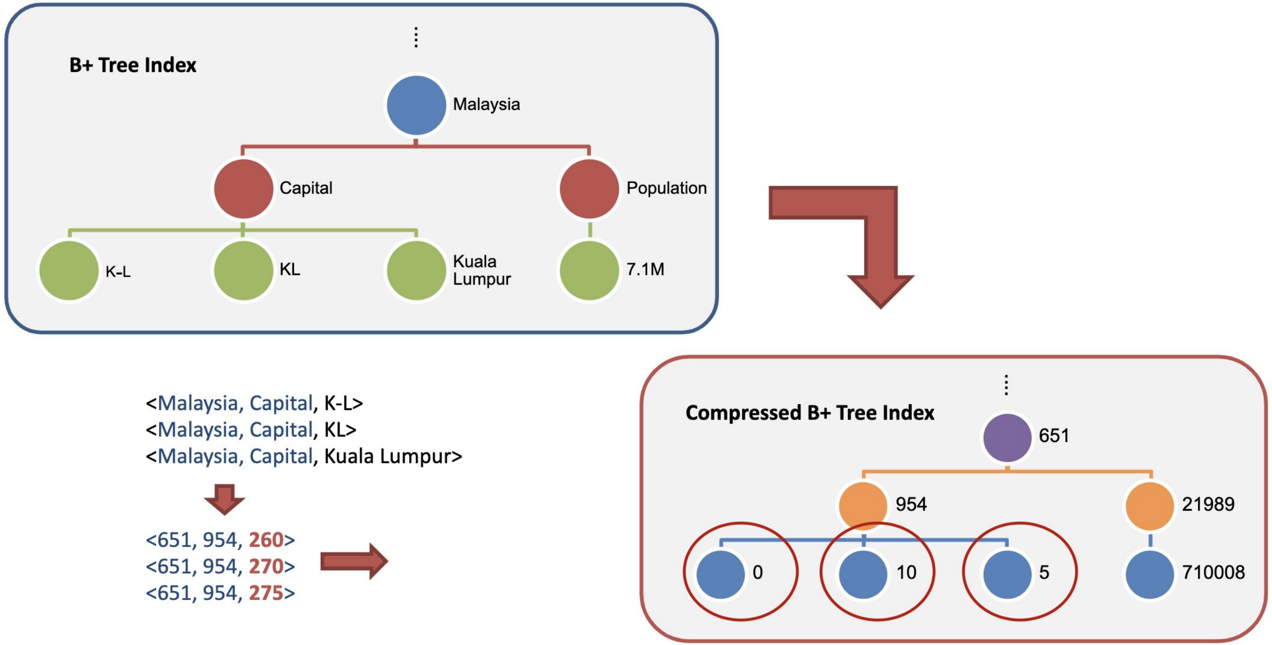
Compression using LZ77

S	P	O
0	1	(2-2) = 0
		(3-2) = 1
		(5-3) = 2

[Neumann et al. RDF-3X: a RISC-style engine for RDF, 2008]

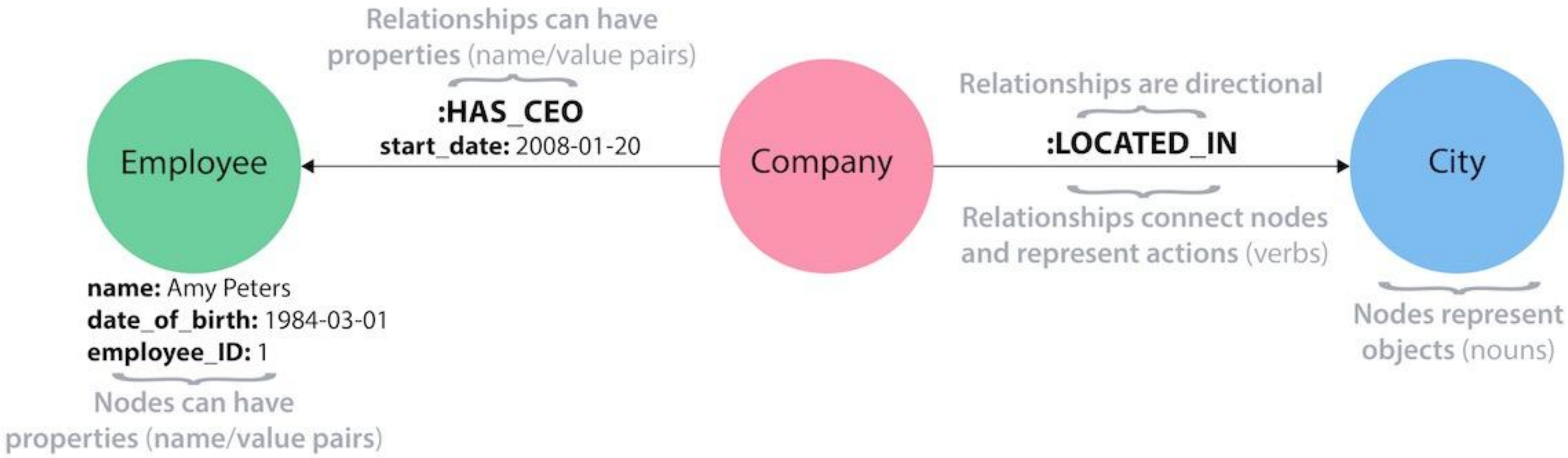
RDF Databases - Native - B+ Trees - RDF-3X

- Compression
  - Stores only the change (δ) between triples



[Neumann et al. RDF-3X: a RISC-style engine for RDF, 2008]

Labeled Property Graph (LPG) Databases



https://neo4j.com/blog/rdf-triple-store-vs-labeled-property-graph-difference/

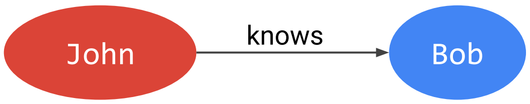
LPG - Cypher

- Standard for Neo4j
- Almost 1-1 mapping to SPARQL

Cypher	SPARQL

`MATCH (s:Person)`
`WHERE s.name = “John”`
`RETURN s;`
	`SELECT ?s WHERE {`
	`?s a :Person;`
	`   :name “John” }`

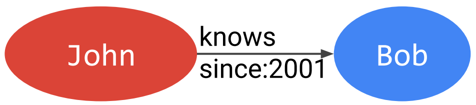
`MATCH (s:Person)-[:knows]-(friend)`
`WHERE s.name = “John”`
`RETURN s, friend ;`
	`SELECT ?s ?friend WHERE {`
	`?s a :Person;`
	`   :name “John” ;`
	`   :knows ?friend }`

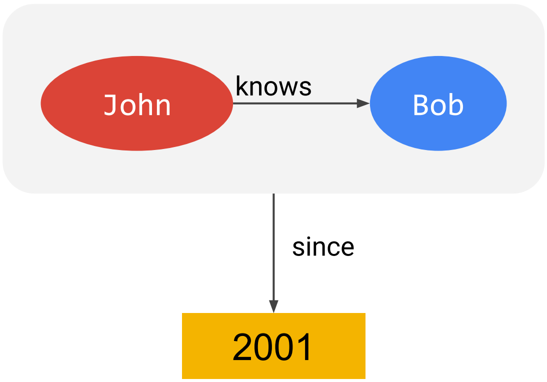


LPG - Cypher

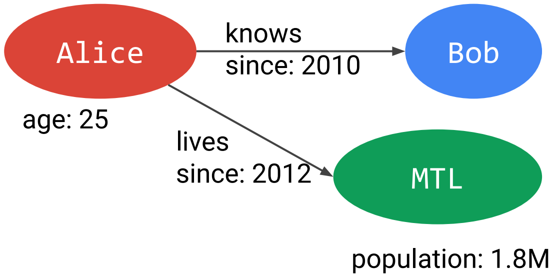
- Standard for Neo4j
- **Almost** 1-1 mapping to SPARQL

Cypher	SPARQL* (Reification)
`MATCH (s:Person)-[:knows {since:2001}] -> (js)` `RETURN s;`	`SELECT ?s WHERE {` `<<?s :knows :js>> :since 2001 }`



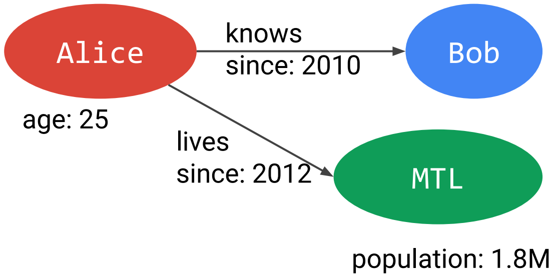


RDF* / SPARQL*




```
<< Alice knows Bob >> since 2010 .
<< Alice lives MTL >> since 2012 .

Alice     age          25 .
MTL       population   1.8M .
```

RDF* / SPARQL*




```
<< Alice knows Bob >> since 2010 .
<< Alice lives MTL >> since 2012 .

Alice      age          25 .
MTL        population   1.8M .
```

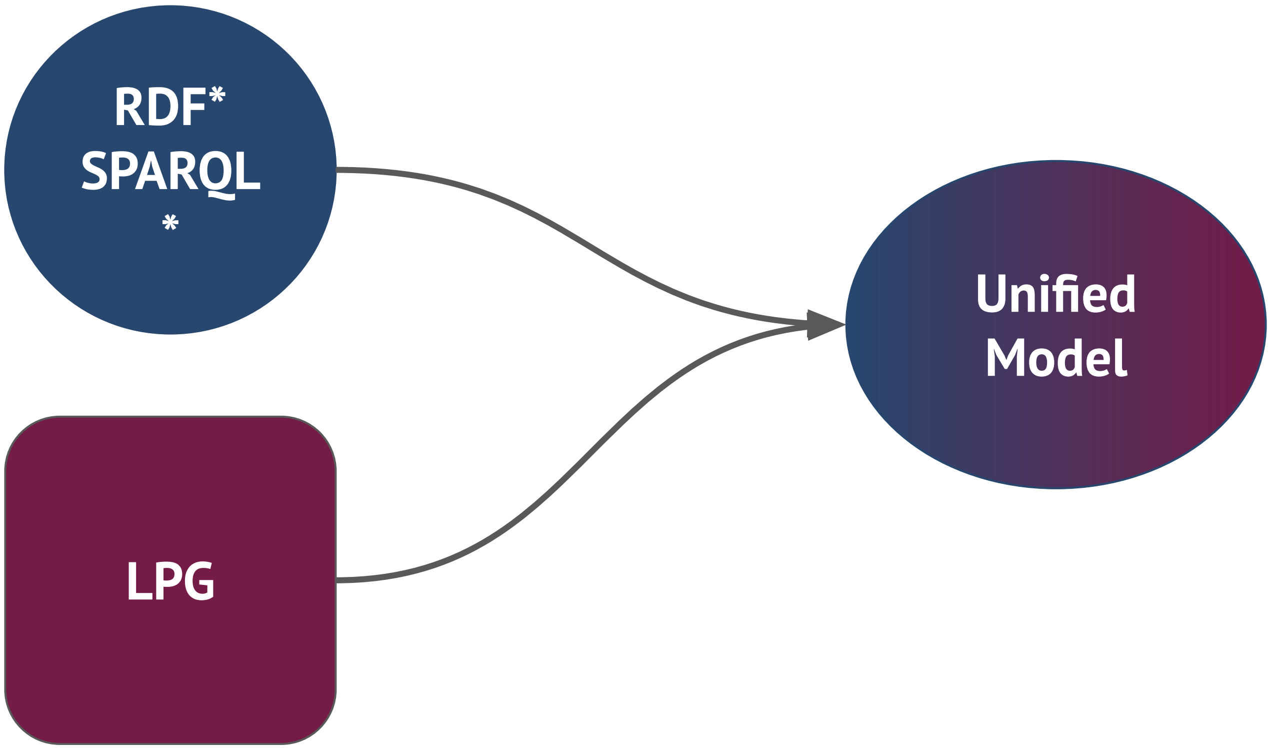
Since when Alice knows Bob?

```
SELECT ?date WHERE {
    << Alice knows Bob >> since ?date . }
```

What is a population of a city when Alice lives since 2012?

```
SELECT ?population WHERE {
    << Alice lives ?city >> since 2012 .
    ?city population ?population . }
```

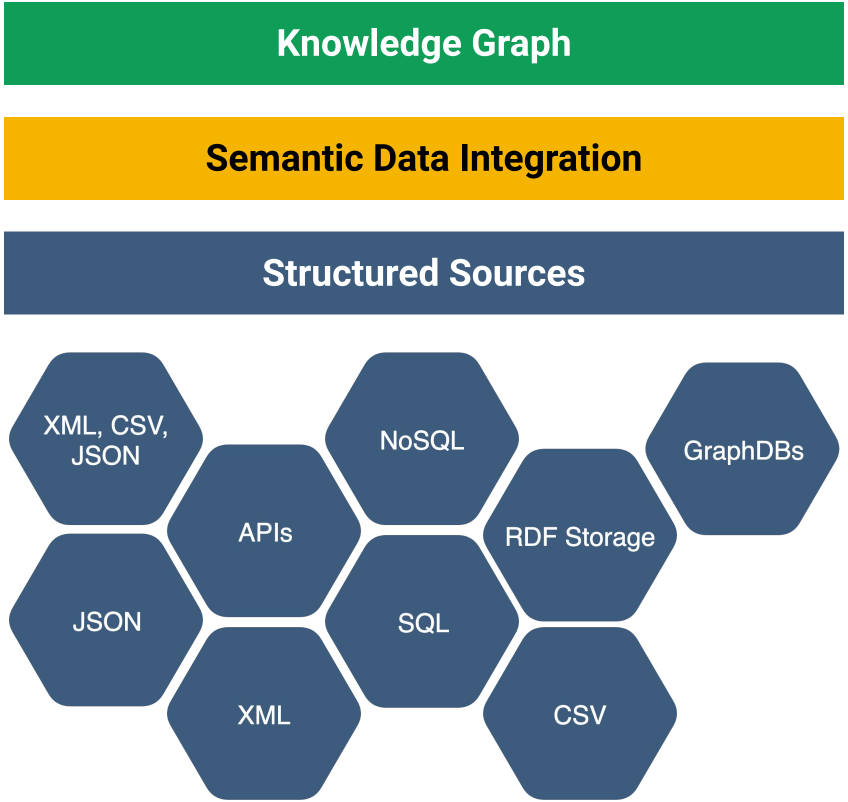
RDF* / SPARQL* + LPG Convergence



Part I: Symbolic

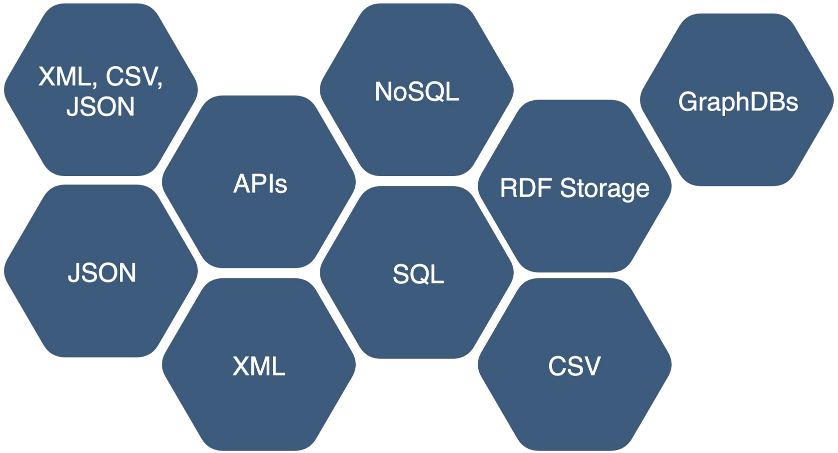
KG Construction

KG Construction

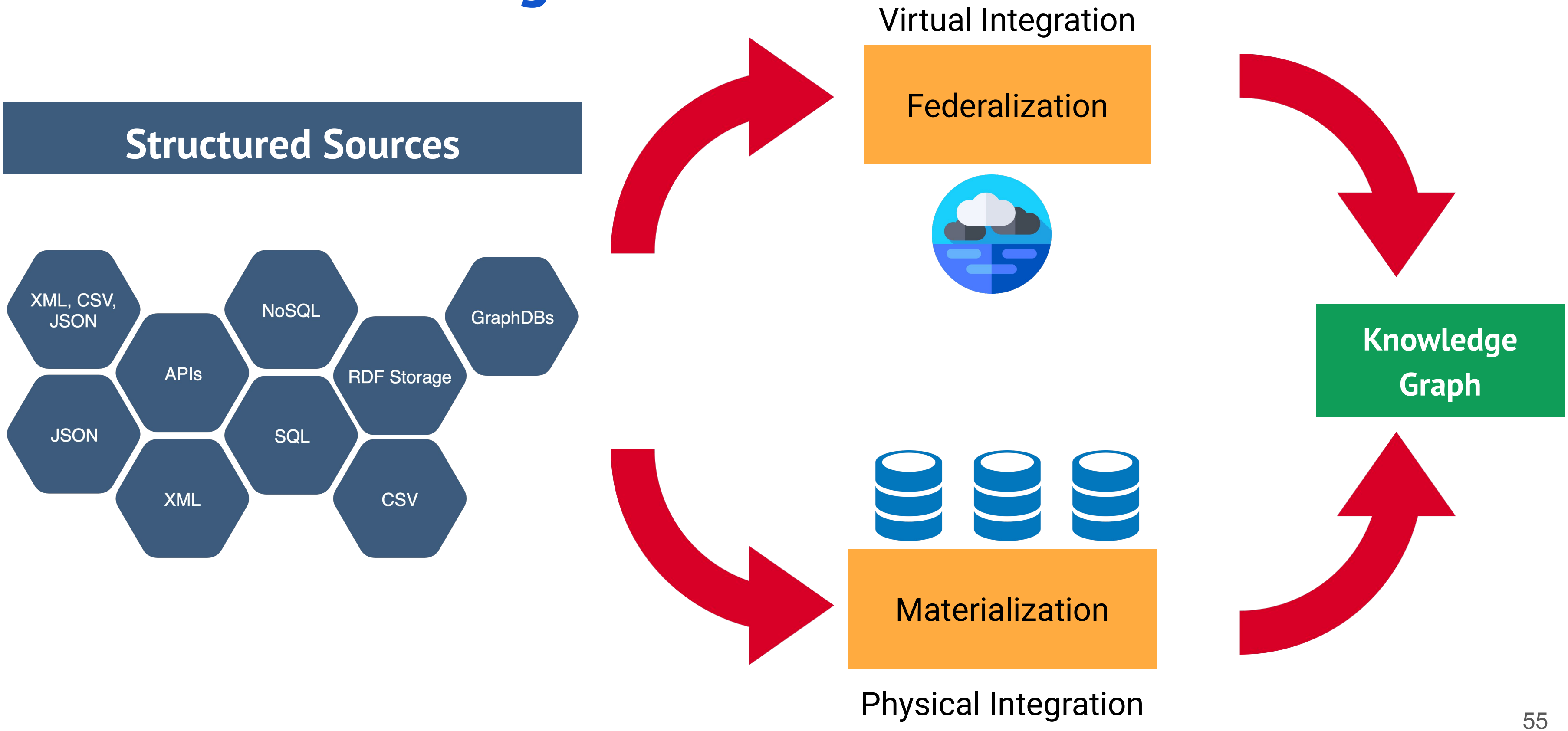


KG Construction

Knowledge Graph	Knowledge Graph
Semantic Data Integration	**Information Retrieval & NLP**
Structured Sources	**Unstructured Sources**

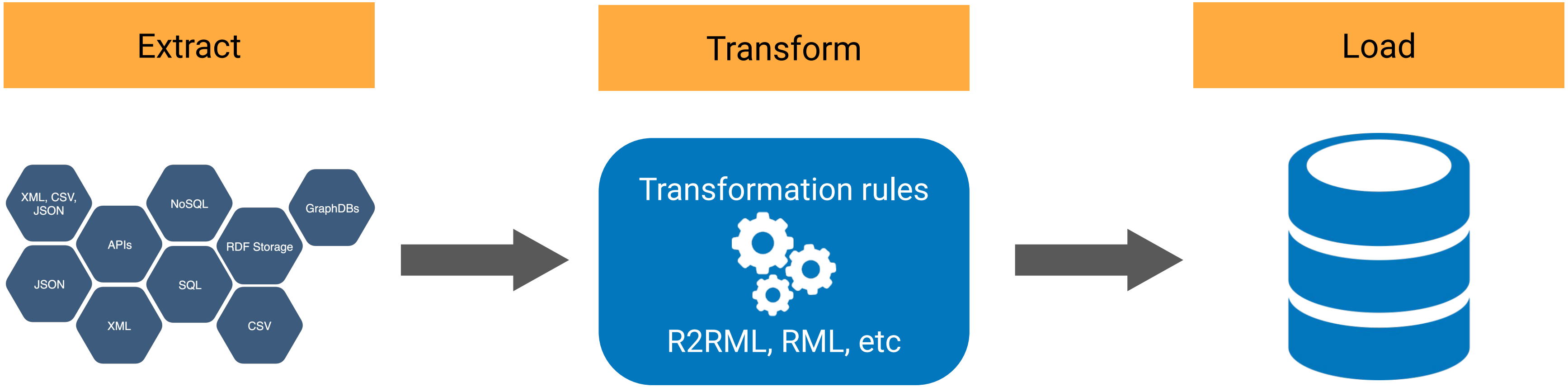


Semantic Data Integration

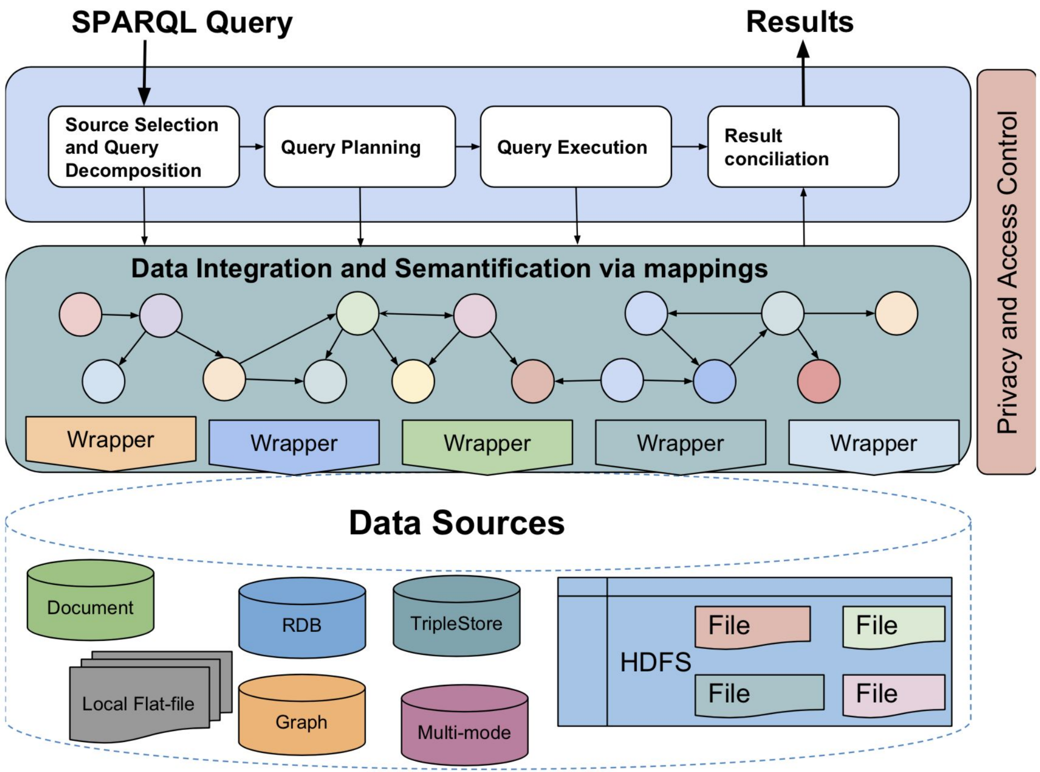


Physical Integration (Materialization)

Data Warehouses



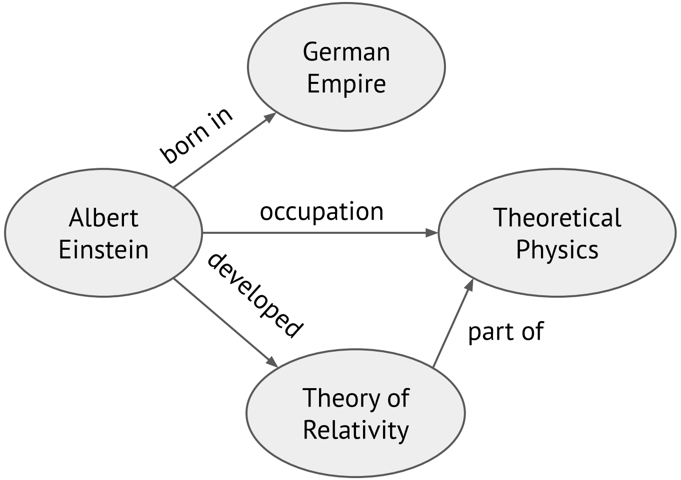
Virtual Integration (Federalization)



Data Lakes

Building KGs from texts

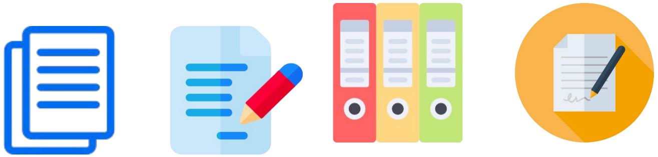
Albert Einstein was a German-born theoretical physicist who developed the theory of relativity.



Knowledge Graph

Information Retrieval

Unstructured Sources



Part II: Vector (some ML)

NLP

NLP - Named Entity Recognition

Entities in a Knowledge Graph

apple (Q89)

fruit of the apple tree
apples

Apple (Q1754545)

1990 album by Mother Love bone

Apple (Q213710)

UK international record label; imprint of Apple Corps Ltd.
LC 01074 | LC 1074 | Apple Records

Apple Inc. (Q312)

American producer of hardware, software, and services, based in Cupertino, California
Apple Computer, Inc. | Apple Computer | Apple Computer Inc | Apple | Apple Incorporated | Apple Computer Incorporated |

Who is the CEO of **Apple**?

Apple belongs to which genus?

movie character

Downey played **Iron Man** in which year?

Who is the alter ego of **Iron man**?

comic character

NLP - Relation Linking

Relations in a Knowledge Graph

- List of known relations
- Surface forms (synonyms), easily multi-lingual
- Relations constraints
- Relations hierarchy
- Most used types of subjects and objects

Name all the movies in which Robert Downey Jr $\underbrace{\text{acted}}^{\text{wdt:P161}}$?

Find me all the films **casting** Robert Downey Jr ?

List all the movies **starring** Robert Downey Junior?

RDJ **has acted** in which movies?

cast member (P161)

actor in the subject production |

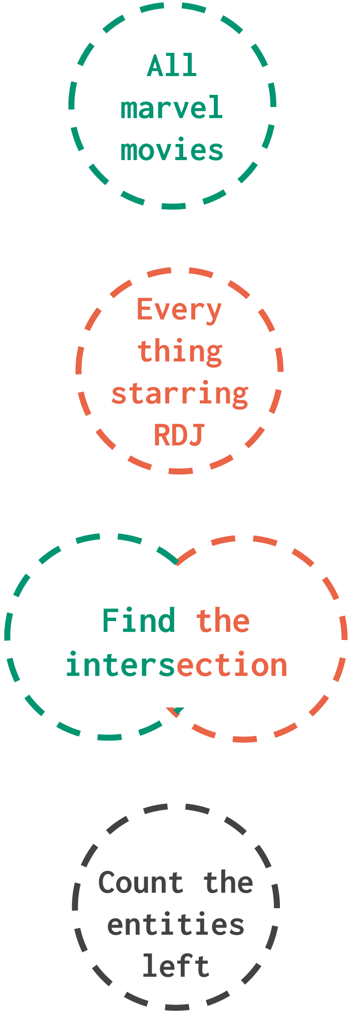
starring | film starring | actor | actress | contestant or a play

performer (P175)

actor, musician, band or other performer associated with this role or musical work

artist | musician | played by | portrayed by | recorded by | recording by | dancer | actor | musical artist

NLP - Question Answering



How many Marvel movies was Robert Downey Jr. casted in?

```
SELECT COUNT(?uri) WHERE {
    ?uri dbp:studio dbr:Marvel_Studios.
    ?uri dbo:starring dbr:Robert_Downey_Jr
}
```

NLP - Language Modeling

Robert Downey Jr. portrayed **[MASK]** in the Marvel movie in 2008.

Knowledge Graph

(Iron Man, cast member, Robert Downey Jr)
(Iron Man, production company, Marvel)
(Iron Man, released, 2008)
(Robert Downey Jr, character role, Tony Stark)
(Tony Stark, pseudonym, Iron Man)

Precise facts | Entities & relations | Explainability

Unstructured Sources

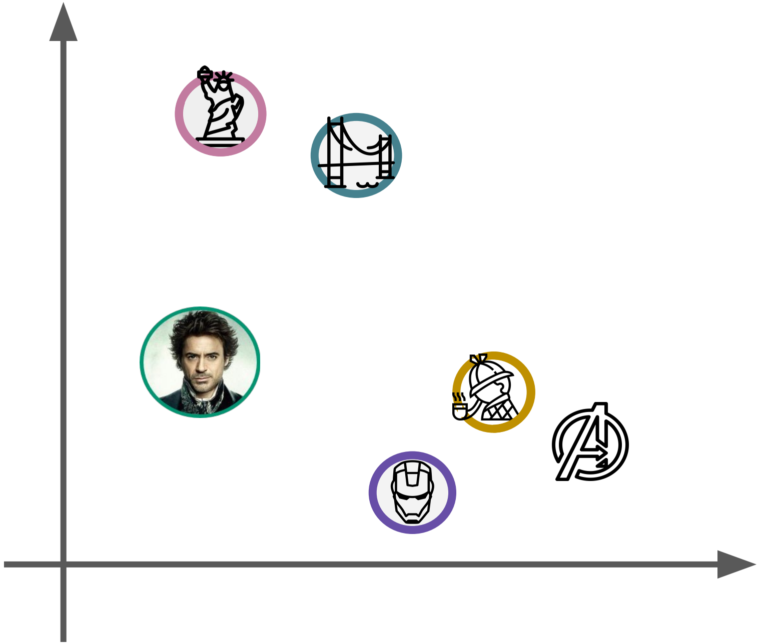
Large-scale text corpora
(Wikipedia, OpenBooks, Reddit, CommonCrawl, etc)

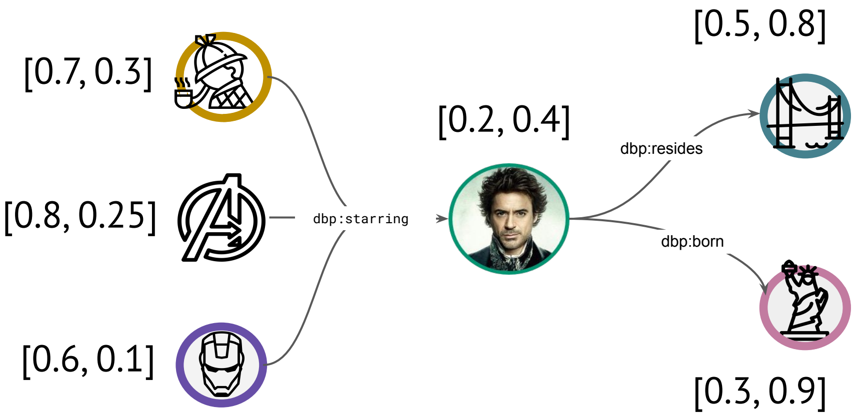
Part II: Vector (some ML)

Representation Learning
(KG Embeddings)

Embeddings

$$E \in \mathbb{R}^{N_e \times d}$$
$$R \in \mathbb{R}^{N_r \times d}$$





Embeddings

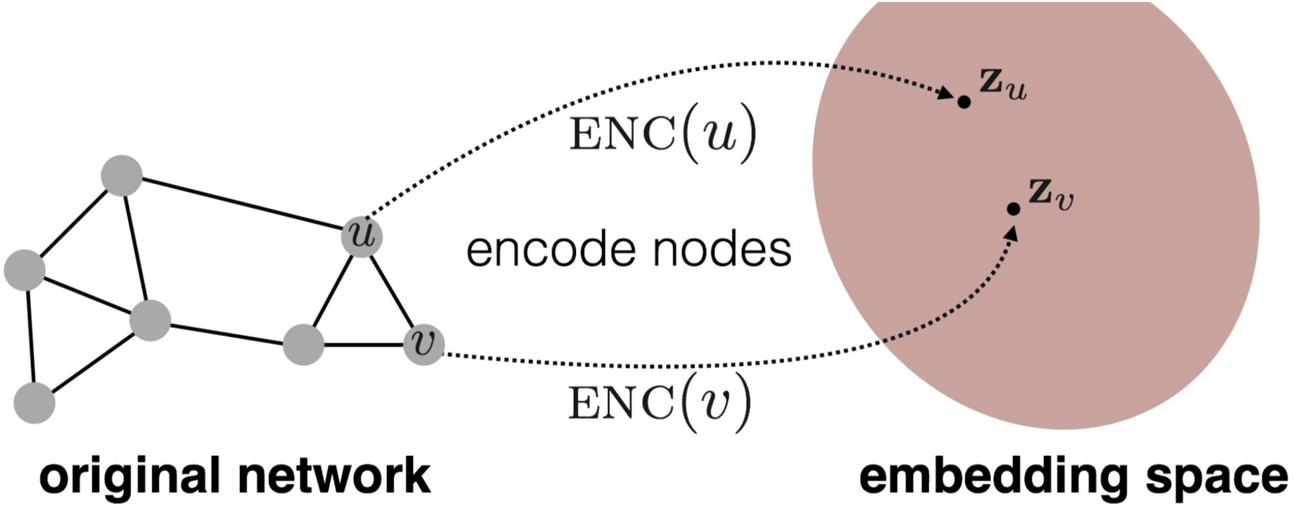
Tensor Factorization

Translation

Neural Networks

Graph Neural Nets

Goal: encode nodes so that **similarity in the embedding space (e.g., dot product)** approximates **similarity in the original network**



Source: Stanford CS224w, http://web.stanford.edu/class/cs224w/

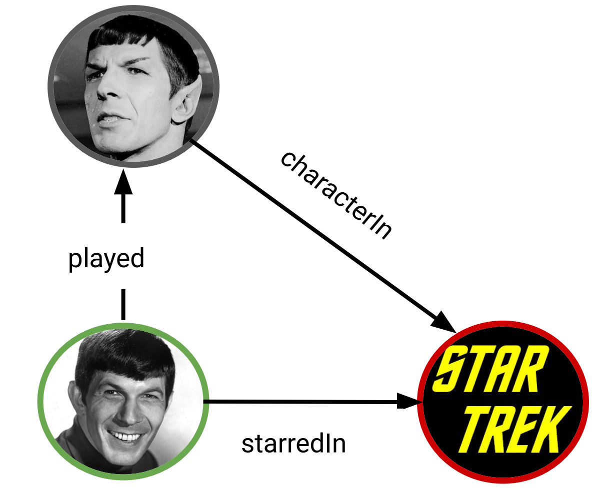
KGE - Graphs as Tensors

Leonard Nimoy Star Trek

		
	0	1
	0	0

starredIn

KGE - Graphs as Tensors



0	1	0
0	0	0
0	0	0

starredIn

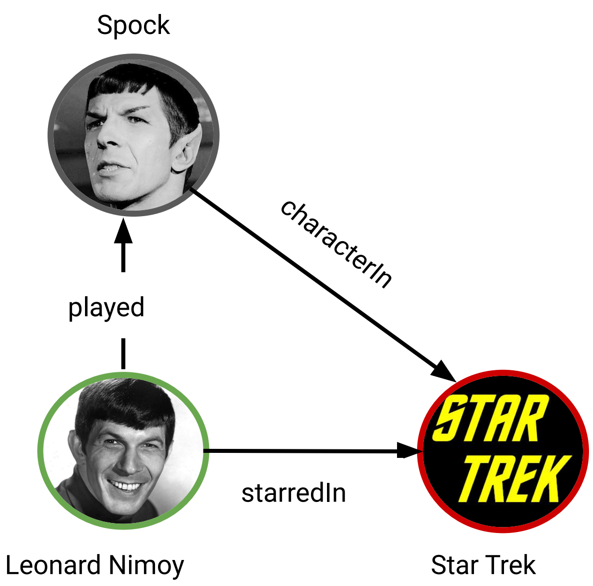
0	0	1
0	0	0
0	0	0

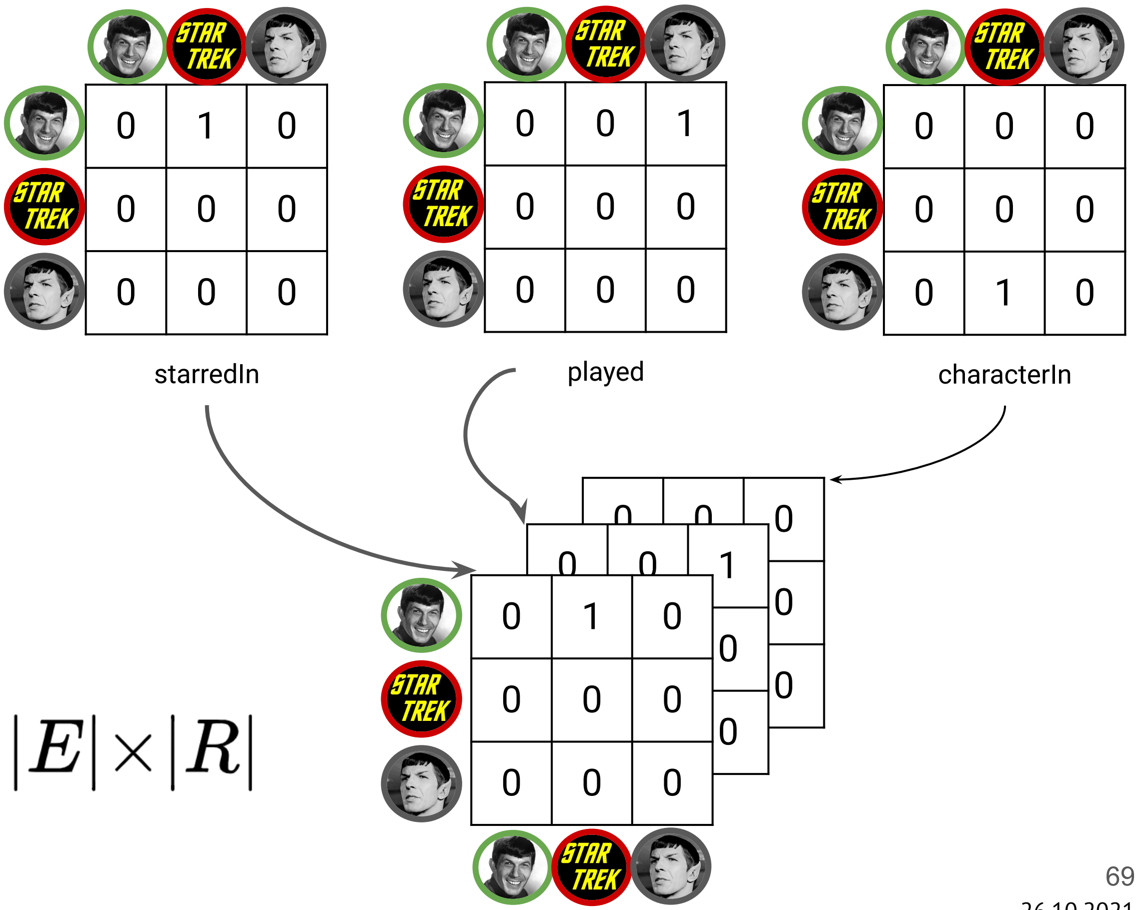
played

0	0	0
0	0	0
0	1	0

characterIn

KGE - Graphs as Tensors



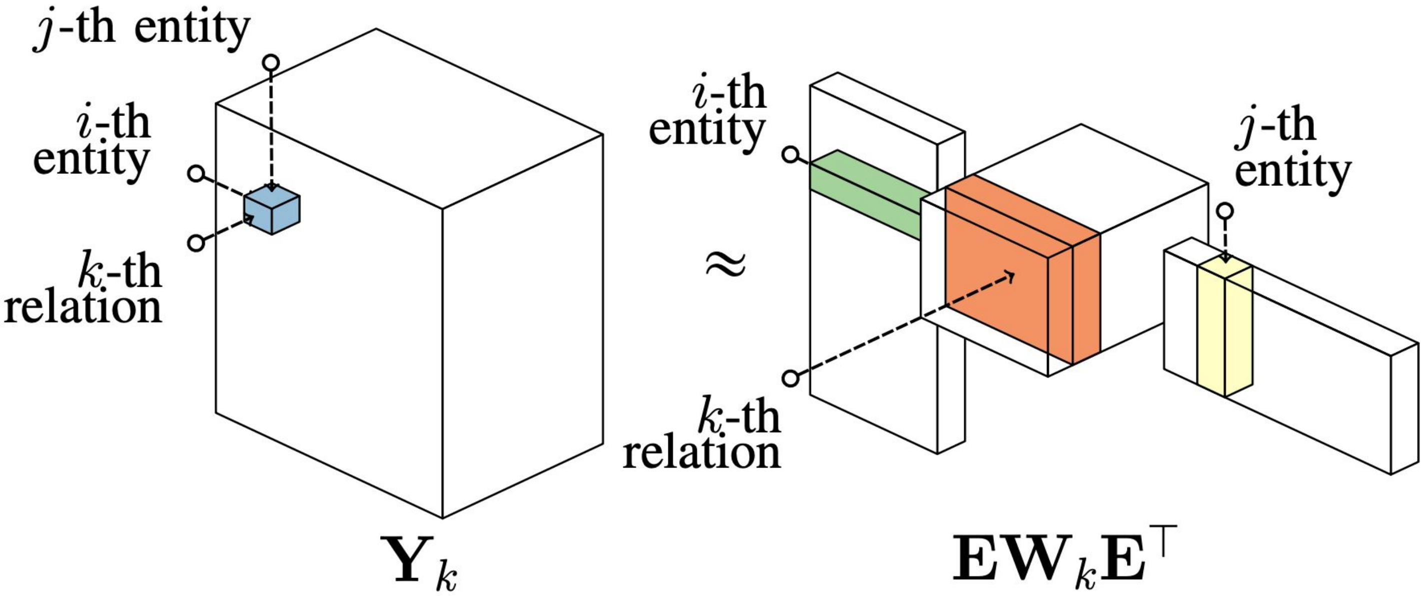


$$\mathcal{T} : \mathbb{R}^{|E|\times|E|\times|R|}$$

KGE - RESCAL

Tensor Factorization

Goal - factorize a sparse 3D tensor to dense E and R

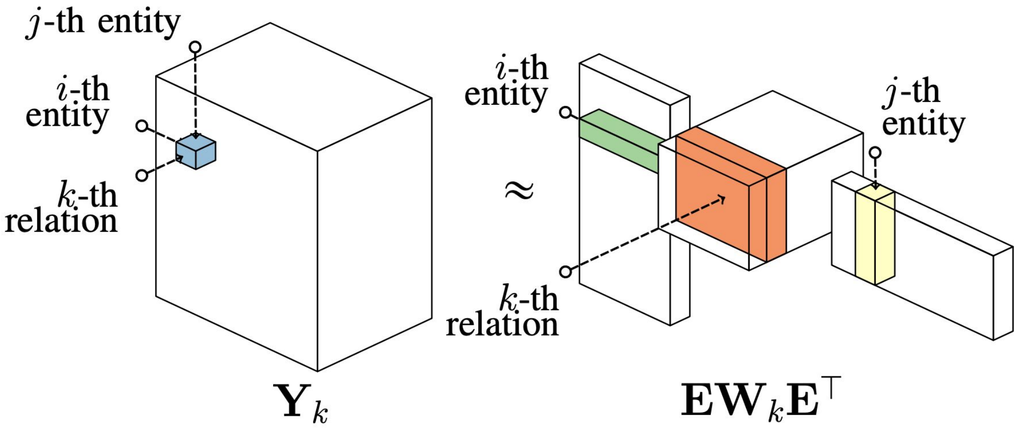


Nickel et al. A review of relational machine learning for knowledge graphs. IEEE. 2015

KGE - RESCAL

Tensor Factorization

Goal - factorize a sparse 3D tensor to dense E and R



$$\mathbf{E} : \mathbb{R}^{|E| \times n} \qquad \mathbf{W} : \mathbb{R}^{|k| \times n \times n}$$

Nickel et al. A review of relational machine learning for knowledge graphs. IEEE. 2015

KGE - TransE

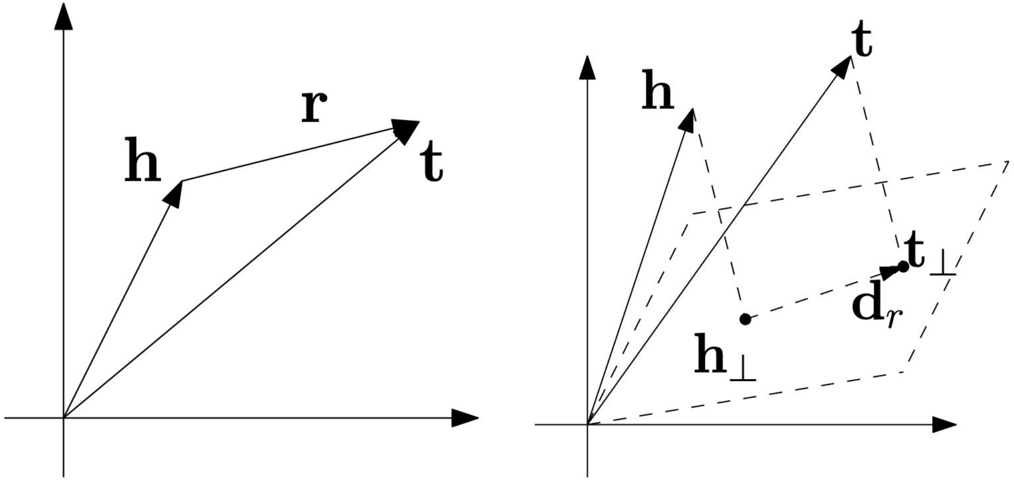
Tensor Factorization

Translation

Translate entities and relations into one embedding space

$$h + r \approx t \quad \text{Moscow} + \text{capitalOf} \approx \text{Russia}$$

$$||h||_2^2 = ||t||_2^2 = 1$$



(a) TransE (b) TransH

Bordes et al. Translating Embeddings for Modeling Multi-relational Data. NIPS 2013
Wang et al. Knowledge Graph Embedding by Translating on Hyperplanes. AAAI 2014

KGE - TransE

Tensor Factorization

Translation

LOTS of models

Cai et al. A Comprehensive Survey of Graph Embedding: Problems, Techniques and Applications. IEEE TKDE 2017

TABLE 9
Knowledge graph embedding using margin-based ranking loss.

GE Algorithm	Energy Function $\mathbf{f_r(h,t)}$
TransE [91]	$\|h+r-t\|_{l1}$
TKRL [53]	$\|M_{rh}h+r-M_{rt}t\|$
TransR [15]	$\|hM_r+r-tM_r\|_2^2$
CTransR [15]	$\|hM_r+r_c-tM_r\|_2^2+\alpha\|r_c-r\|_2^2$
TransH [14]	$\|(h-w_r^T h w_r)+d_r-(t-w_r^T t w_r)\|_2^2$
SePLi [39]	$\frac{1}{2}\|W_i e_{ih}+b_i-e_{it}\|^2$
TransD [125]	$\|M_{rh}h+r-M_{rt}t\|_2^2$
TranSparse [126]	$\|M_r^h(\theta_r^h)h+r-M_r^t(\theta_r^t)t\|_{l_1/2}^2$
m-TransH [127]	$\|\sum_{\rho\in\mathcal{M}(R_r)} a_r(\rho)\mathbb{P}_{n_r}(t(\rho))+b_r\|^2, t\in\mathcal{N}^{\mathcal{M}(R_r)}$
DKRL [128]	$\|h_d+r-t_d\|+\|h_d+r-t_s\|+\|h_s+r-t_d\|$
ManifoldE [129]	Sphere: $\|\varphi(h)+\varphi(r)-\varphi(t)\|^2$ Hyperplane: $(\varphi(h)+\varphi(r_{head}))^T(\varphi(t)+\varphi(r_{tail}))$ φ is the mapping function to Hilbert space
TransA [130]	$\|h+r-t\|$
puTransE [43]	$\|h+r-t\|$
KGE-LDA [60]	$\|h+r-t\|_{l1}$
SE [90]	$\|R_u h-R_u t\|_{l1}$
SME [92] linear SME [92] bilinear	$(W_{u1}r+W_{u2}h+b_u)^T(W_{v1}r+W_{v2}t+b_v)$ $(W_{u1}r+W_{u2}h+b_u)^T(W_{v1}r+W_{v2}t+b_v)$
SSP [59]	$-\lambda\|e-s^T es\|_2^2+\|e\|_2^2,\ S(s_h,s_t)=\frac{s_h+s_t}{\|s_h+s_t\|_2^2}$
NTN [131]	$u_r^T\tanh(h^T W_r t+W_{rh}h+W_{rt}t+b_r)$
HOLE [132]	$r^T(h\star t)$, where $\star$ is circular correlation
MTransE [133]	$\|h+r-t\|_{l1}$

KGE - Incorporating OWL Rules

Tensor Factorization

Translation

$$\min_{\theta} \sum_{(h,r,t)\in\mathcal{S}} \alpha^r_{h,t} \log(1+\exp(-y^r_{h,t} f^r_{h,t})) + \lambda \sum_{i=1}^{l} \frac{\mathcal{R}_i}{N_i}$$
$$\text{subject to} \quad \|h\| = 1 \text{ and } \|t\| = 1 .$$

Rule	Definition $\forall \mathtt{h}, \mathtt{t}, \mathtt{s} \in \mathcal{E} : \ldots$	Formulation based on score function	Formulation based on NN	Equivalent regularization form (Denoted as $\mathcal{R}_i$ in Equation (2))
Equivalence	$(\mathtt{h}, r_1, \mathtt{t}) \Leftrightarrow (\mathtt{h}, r_2, \mathtt{t})$	$f^{r_1}_{h,t} = f^{r_2}_{h,t} + \xi_{h,t}$	$\Phi^T_{h,t}(\boldsymbol{\beta}^{r_1} - \boldsymbol{\beta}^{r_2}) = \xi_{h,t}$	$\max(\lVert\boldsymbol{\beta}^{r_1} - \boldsymbol{\beta}^{r_2}\rVert_1 - \xi_{\text{Eq}}, 0)$
Symmetric	$(\mathtt{h}, r, \mathtt{t}) \Leftrightarrow (\mathtt{t}, r, \mathtt{h})$	$f^r_{h,t} = f^r_{t,h} + \xi_{h,t}$	$(\Phi_{h,t} - \Phi_{t,h})^T \boldsymbol{\beta}^r = \xi_{h,t}$	$\max(\lvert(\Phi_{h,t} - \Phi_{t,h})^T \boldsymbol{\beta}^r\rvert - \xi_{\text{Sy}}, 0)$
Asymmetric	$(\mathtt{h}, r, \mathtt{t}) \Rightarrow \neg(\mathtt{t}, r, \mathtt{h})$	$f^r_{h,t} = f^r_{t,h} + \mathcal{M}_{h,t}$	$(\Phi_{h,t} - \Phi_{t,h})^T \boldsymbol{\beta}^r = \mathcal{M}$	NC
Negation	$(\mathtt{h}, r_1, \mathtt{t}) \Leftrightarrow \neg(\mathtt{h}, r_2, \mathtt{t})$	$f^{r_1}_{h,t} = \mathcal{M} - f^{r_2}_{h,t} + \xi_{h,t}$	$\Phi^T_{h,t}(\boldsymbol{\beta}^{r_1} + \boldsymbol{\beta}^{r_2}) = \mathcal{M} + \xi_{h,t}$	NC
Implication	$(\mathtt{h}, r_1, \mathtt{t}) \Rightarrow (\mathtt{h}, r_2, \mathtt{t})$	$f^{r_1}_{h,t} \leq f^{r_2}_{h,t}$	$\Phi^T_{h,t}(\boldsymbol{\beta}^{r_1} - \boldsymbol{\beta}^{r_2}) \leq 0$	$\max(\sum_i(\boldsymbol{\beta}^{r_1}_i - \boldsymbol{\beta}^{r_2}_i) + \xi_{\text{Im}}, 0)$
Inverse	$(\mathtt{h}, r_1, \mathtt{t}) \Rightarrow (\mathtt{t}, r_2, \mathtt{h})$	$f^{r_1}_{h,t} \leq f^{r_2}_{t,h}$	$\Phi^T_{h,t}\boldsymbol{\beta}^{r_1} - \Phi^T_{t,h}\boldsymbol{\beta}^{r_2} \leq 0$	$\max(\Phi^T_{h,t}\boldsymbol{\beta}^{r_1} - \Phi^T_{t,h}\boldsymbol{\beta}^{r_2} + \xi_{\text{In}}, 0)$
Reflexivity	$(\mathtt{h}, r, \mathtt{h})$	$f^r_{h,h} = \mathcal{M} - \xi_{h,h}$	$\Phi^T_{h,h}\boldsymbol{\beta}^r = \mathcal{M} - \xi_{h,h}$	NC
Irreflexive	$\neg(\mathtt{h}, r, \mathtt{h})$	$f^r_{h,h} = \xi_{h,h}$	$\Phi^T_{h,h}\boldsymbol{\beta}^r = \xi_{h,h}$	NC
Transitivity	$(\mathtt{h}, r, \mathtt{t}) \wedge (\mathtt{t}, r, \mathtt{s}) \Rightarrow (\mathtt{h}, r, \mathtt{s})$	$\sigma(f^r_{h,s}) \geq \sigma(f^r_{h,t}) \times \sigma(f^r_{t,s})$	$\sigma(\Phi_{h,t}\boldsymbol{\beta}^r) \times \sigma(\Phi_{t,s}\boldsymbol{\beta}^r) - \sigma(\Phi^T_{h,s}\boldsymbol{\beta}^r) \leq 0$	$\max(\sigma(\Phi_{h,t}\boldsymbol{\beta}^r) \times \sigma(\Phi_{t,s}\boldsymbol{\beta}^r) - \sigma(\Phi^T_{h,s}\boldsymbol{\beta}^r) + \xi_{\text{Tr}}, 0)$
Composition	$(\mathtt{h}, r_1, \mathtt{t}) \wedge (\mathtt{t}, r_2, \mathtt{s}) \Rightarrow (\mathtt{h}, r_3, \mathtt{s})$	$\sigma(f^{r_1}_{h,s}) \geq \sigma(f^{r_2}_{h,t}) \times \sigma(f^{r_3}_{t,s})$	$\sigma(\Phi_{h,t}\boldsymbol{\beta}^{r_1}) \times \sigma(\Phi_{t,s}\boldsymbol{\beta}^{r_2}) - \sigma(\Phi^T_{h,s}\boldsymbol{\beta}^{r_3}) \leq 0$	$\max(\sigma(\Phi_{h,t}\boldsymbol{\beta}^{r_1}) \times \sigma(\Phi_{t,s}\boldsymbol{\beta}^{r_2}) - \sigma(\Phi^T_{h,s}\boldsymbol{\beta}^{r_3}) + \xi_{\text{Co}}, 0)$

Table 1: Formulation and representation of rules (NC: Not considered for implementation).

Nayyeri et al. LogicENN: A Neural Based Knowledge Graphs Embedding Model with Logical Rules. https://arxiv.org/abs/1908.07141

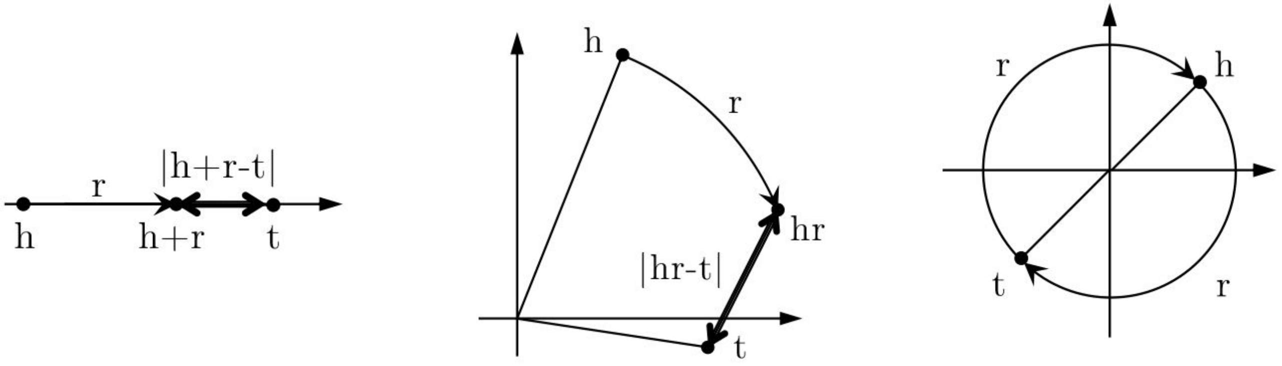
KGE - RotatE

Tensor Factorization

Translation

Idea:
Entities are vectors in **complex space**

Relations: rotations in **complex space**



(a) TransE models $\mathbf{r}$ as translation in real line.

(b) RotatE models $\mathbf{r}$ as rotation in complex plane.

(c) RotatE: an example of modeling symmetric relations $\mathbf{r}$ with $r_i = -1$

Figure 1: Illustrations of TransE and RotatE with only 1 dimension of embedding.

Score function: $d_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$ $\quad |r_i| = 1$

Loss & Optimization: $L = -\log \sigma(\gamma - d_r(\mathbf{h}, \mathbf{t})) - \sum_{i=1}^{n} \frac{1}{k} \log \sigma(d_r(\mathbf{h}'_i, \mathbf{t}'_i) - \gamma),$

Sun et al. RotatE: Knowledge Graph Embedding By Relational Rotation In Complex Space. ICLR 2019

KGE - RotatE & Patterns

Translation

Model	Score Function	Symmetry	Antisymmetry	Inversion	Composition
SE	$-\lVert \boldsymbol{W}_{r,1}\mathbf{h} - \boldsymbol{W}_{r,2}\mathbf{t} \rVert$	✗	✗	✗	✗
TransE	$-\lVert \mathbf{h} + \mathbf{r} - \mathbf{t} \rVert$	✗	✓	✓	✓
TransX	$-\lVert g_{r,1}(\mathbf{h}) + \mathbf{r} - g_{r,2}(\mathbf{t}) \rVert$	✓	✓	✗	✗
DistMult	$\langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle$	✓	✗	✗	✗
ComplEx	$\mathrm{Re}(\langle \mathbf{h}, \mathbf{r}, \overline{\mathbf{t}} \rangle)$	✓	✓	✓	✗
RotatE	$-\lVert \mathbf{h} \circ \mathbf{r} - \mathbf{t} \rVert$	✓	✓	✓	✓

Table 2: The pattern modeling and inference abilities of several models.

Sun et al. Rotate: Knowledge graph embedding by relational rotation in complex space. ICLR 2019

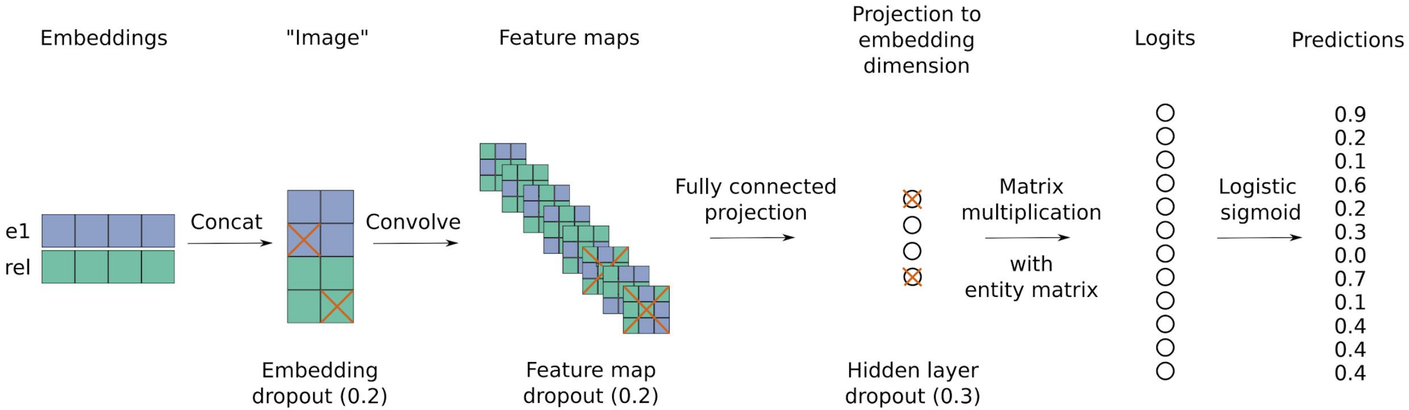
KGE - ConvE

Tensor Factorization

Translation

Convolution

Goal: CNNs for predicting a probability of the object



Score function: $$\psi_r(\mathbf{e}_s, \mathbf{e}_o) = f(\mathrm{vec}(f([\overline{\mathbf{e}_s}; \overline{\mathbf{r}_r}] * \omega))\mathbf{W})\mathbf{e}_o,$$

Loss & Optimization: $$\mathcal{L}(p,t) = -\frac{1}{N}\sum_i (t_i \cdot \log(p_i) + (1-t_i)\cdot\log(1-p_i)),$$

Minervini et al. Convolutional 2D Knowledge Graph Embeddings. AAAI 2018

KGE - CoKE

Tensor Factorization

Translation

Transformer

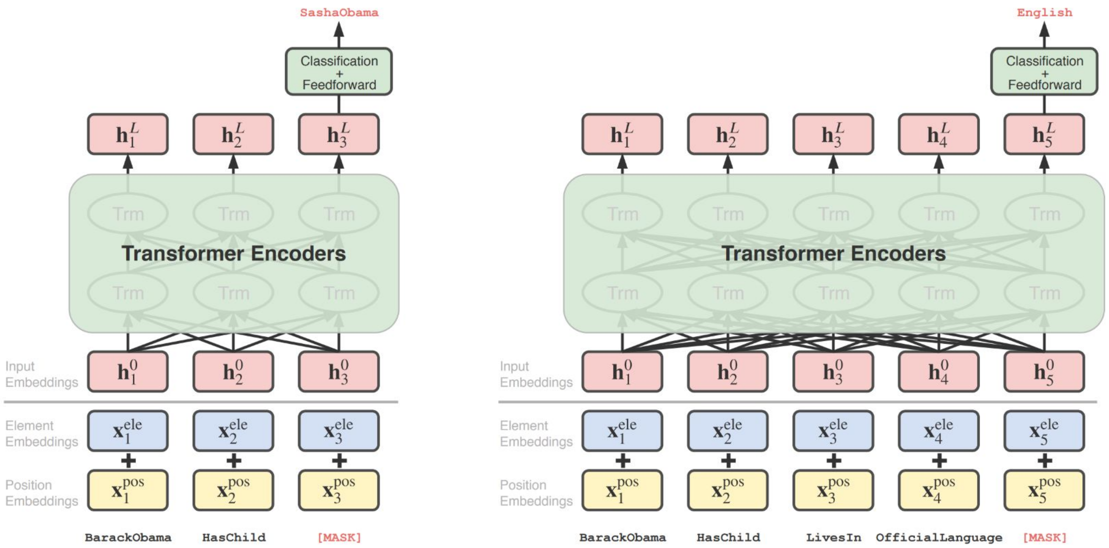


Figure 2: Overall framework of CoKE. An edge (left) or a path (right) is given as an input sequence, with an entity replaced by a special token [MASK]. The input is then fed into a stack of Transformer encoder blocks. The final hidden state corresponding to [MASK] is used to predict the target entity.

Wang et al. CoKE: Contextualized Knowledge Graph Embeddings. arxiv 2019

KGE - CompGCN Encoder

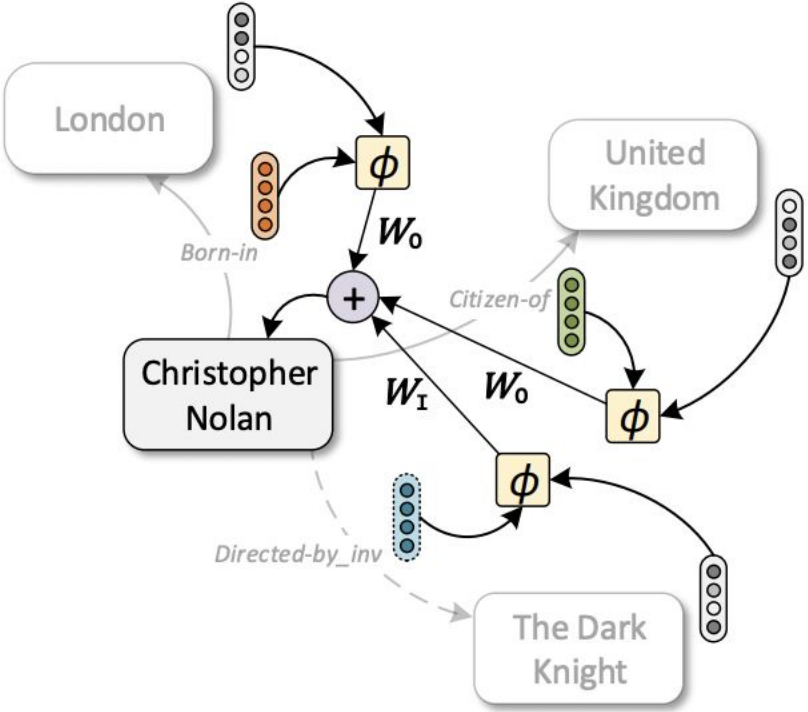
Tensor Factorization

Translation

Transformer

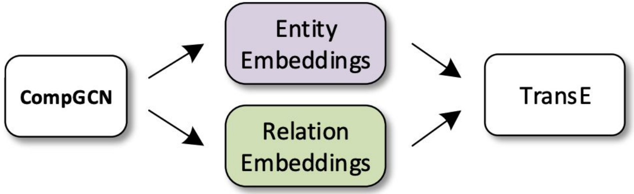
Graph Neural Nets

- Message Passing
- Encoder-Decoder
- Many architectures



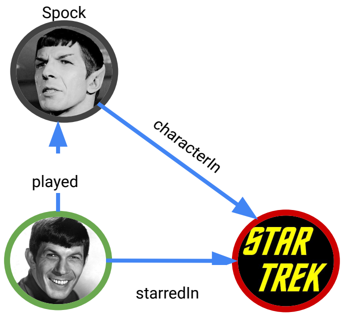
$$\boldsymbol{h}_v = f\left(\sum_{(u,r)\in\mathcal{N}(v)} \boldsymbol{W}_{\lambda(r)} \phi(\boldsymbol{x}_u, \boldsymbol{z}_r) \right)$$

CompGCN Update



Vashishth et al. Composition-Based Multi-Relational Graph Convolutional Networks. ICLR 2020

KGE - Training



Optimization

Loss Function

Negative Sampling

Entity matrix

$$\mathbf{E} : \mathbb{R}^{|E| \times n}$$

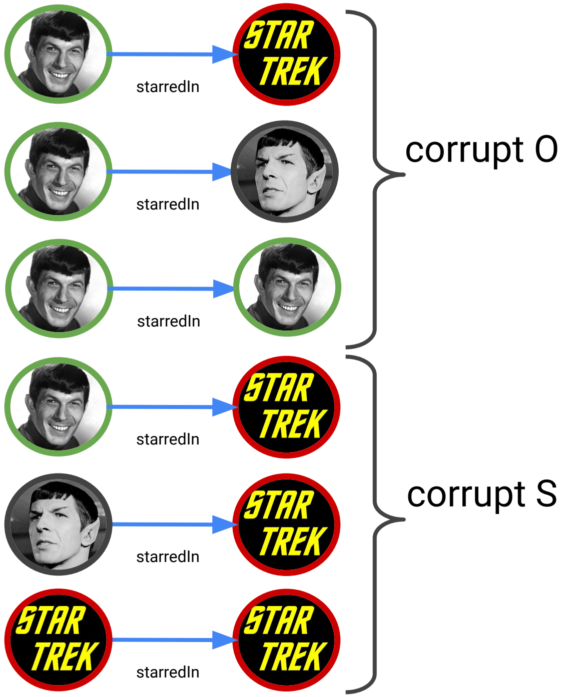
Spock = [0.1, 0.2, 0.3]
Leonard Nimoy = [0.4, 0.8, 0.1]
Star Trek = [0.22, 0.34, 0.87]

Relations matrix

$$\mathbf{W} : \mathbb{R}^{|k| \times n}$$

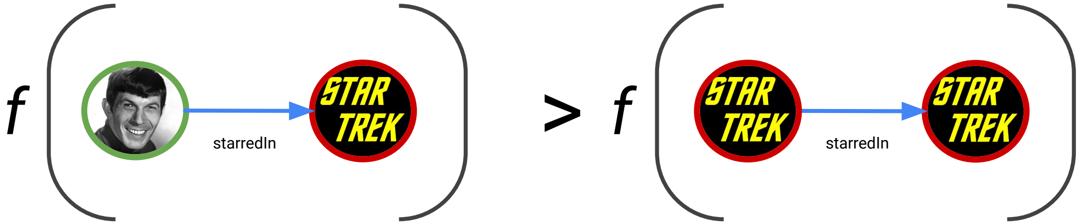
characterIn = [0.1, 0.1, 0.6]
played = [0.2, 0.3, 0.4]
starredIn = [0.9, -0.2, 0.1]

KGE - Training - Negative Sampling + Margin Loss



$$L(\Omega) = \sum_{(e_1,r,e_2)\in T} \sum_{(e'_1,r,e'_2)\in T'} \max\{S_{(e'_1,r,e'_2)} - S_{(e_1,r,e_2)} + 1, 0\}$$

Negative sampling: incorrect triples should have lower (higher) score than correct triples



Big Picture in $\mathbb{R}^5$

Transductive | Triples | Supervised | Unimodal | Small

Inductive | Hyper-relational | Unsupervised | Multimodal | Large (sampling)

SETTING

TASK

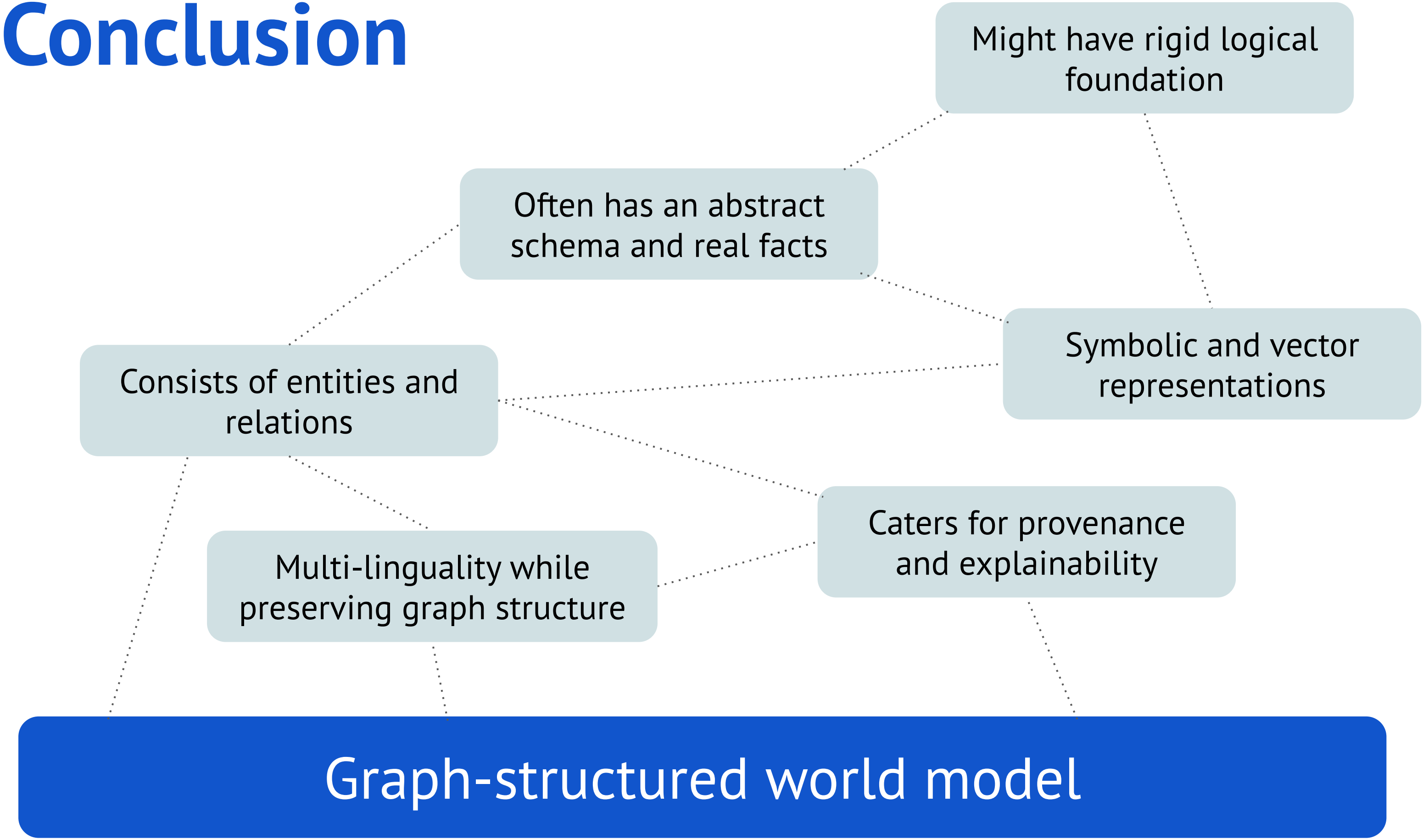
Link prediction | Node classification | Entity Matching | Query Embedding

Theoretical Understanding

Graph Encoder

Knowledge Graph

Conclusion


Might have rigid logical foundation
Often has an abstract schema and real facts
Symbolic and vector representations
Consists of entities and relations
Caters for provenance and explainability
Multi-linguality while preserving graph structure
Graph-structured world model