Graph-Based RDF Knowledge Graph Research

Lei Zou
Peking University, China

北京大学

Collaborators

Prof. Tamer Ozsu, University of Waterloo

Prof. Jeffrey Xu Yu, The Chinese University of Hong Kong

Prof. Lei Chen, Hong Kong University of Science and Technology

Dr. Haixun Wang, Facebook

Collaborators

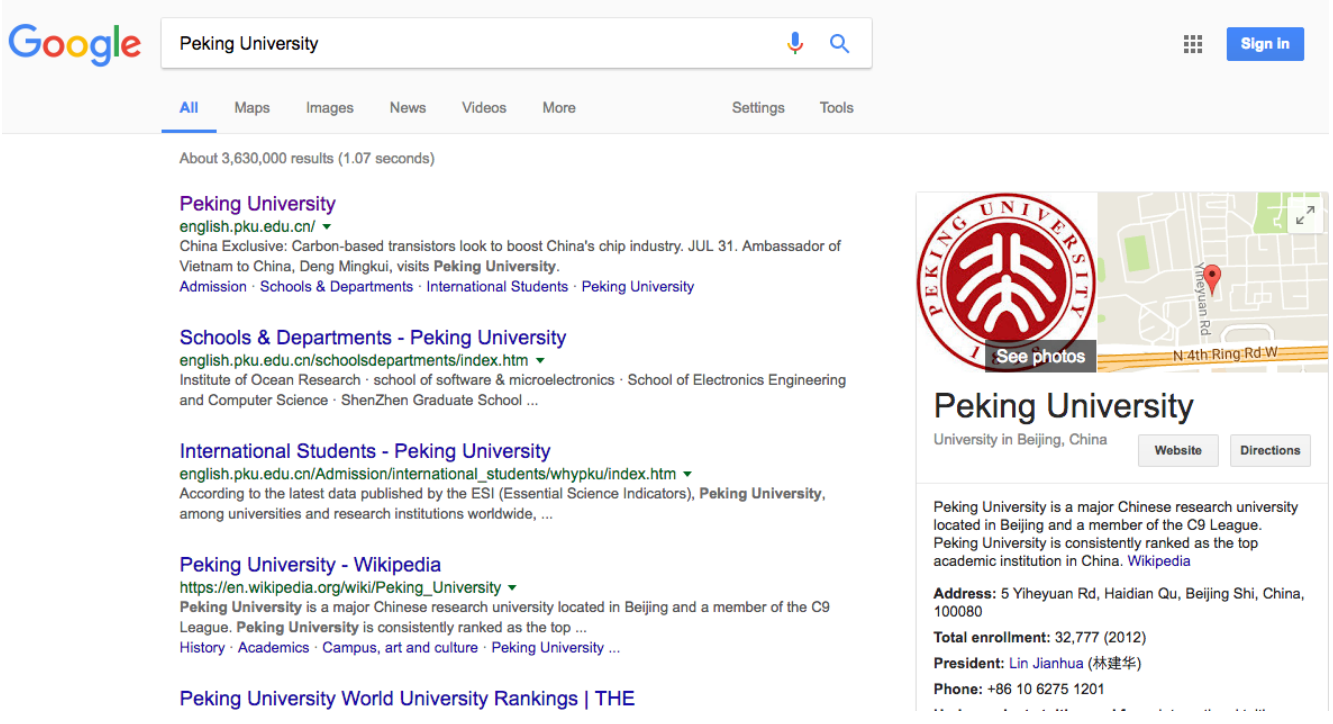
PhD students (including alumni):

Weiguo Zheng, graduated at 2015, post-doc in The Chinese University of Hong Kong;
Peng Peng, graduated at 2016, assistant professor in Hunan University.
Shuo Han
Seng Hu

Master Students (including alumni):
Shuo Yang
Xinbo Zhang

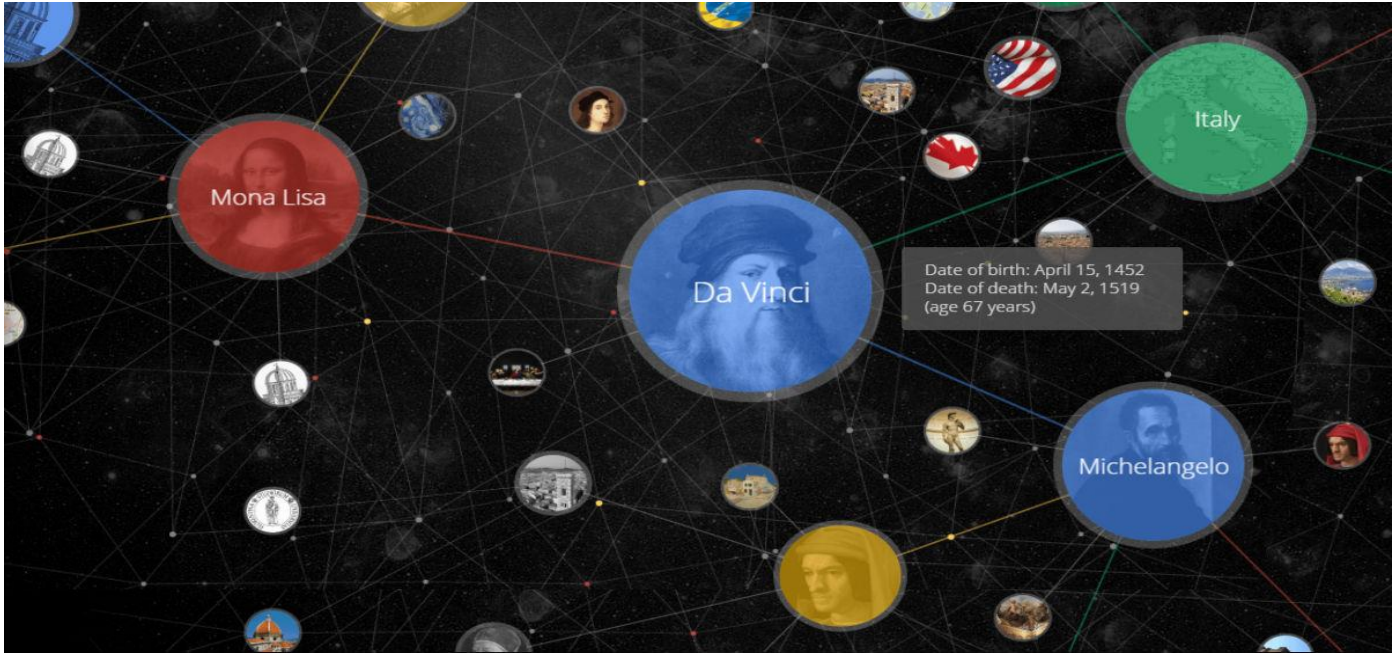
Knowledge Graph

Google launches **Knowledge Graph** project at 2012.



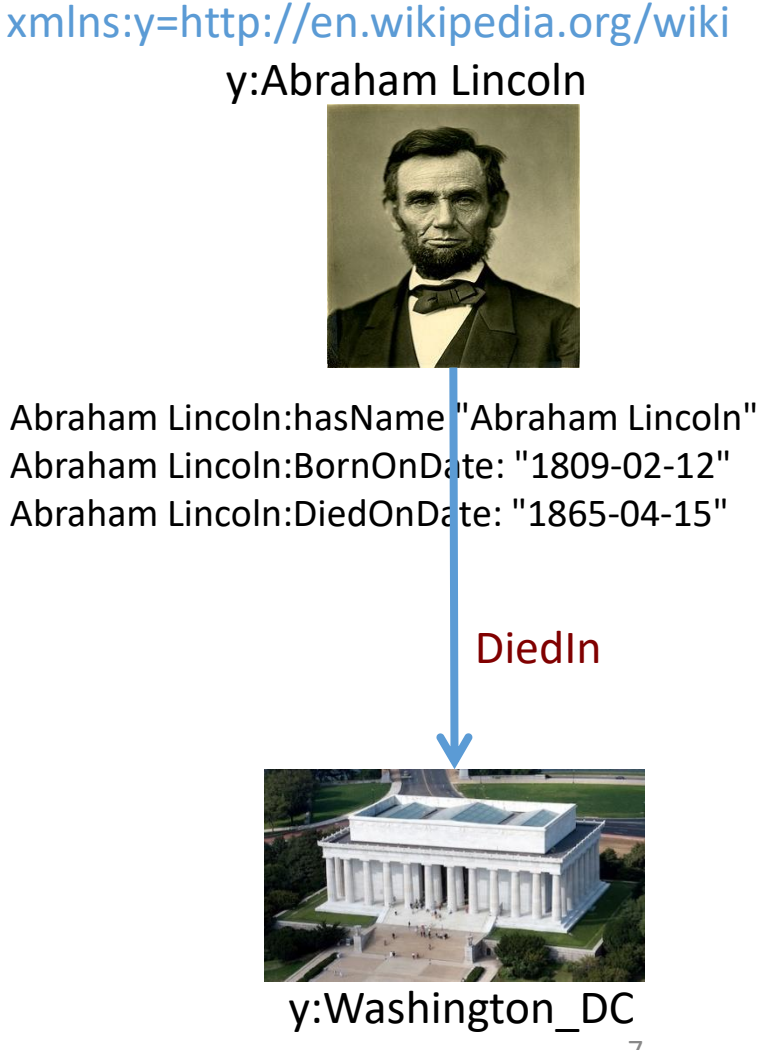
Knowledge Graph

Essentially, KG is a sematic network, which models **the entities (including properties) and the relation between each other.**



RDF Data Model

- RDF is the de-facto standard data format for Knowledge Graph.
- Simple triple format <subject, predicate, object>
- Represent both the properties of entities and relations between entities.



RDF & SPARQL

RDF Datasets

Subject	Predicate	Object
Abraham_Lincoln	hasName	“Abraham Lincoln"
Abraham_Lincoln	BornOnDate	“1809-02-12"
Abraham_Lincoln	DiedOnDate	“1865-04-15"”
Abraham_Lincoln	DiedIn	Washington_DC
Abraham_Lincoln	bornIn	Hodgenville KY
Reese_Witherspoon	bornOnDate	"1976-03-22"
Reese_Witherspoon	bornIn	New_Orleans_LA
New_Orleans_LA	foundingYear	“1718”
New Orleans LA	locatedIn	United_States
United_States	hasName	“United States ”
United_States	hasCapital	Washington_DC
United_States	foundingYear	“1776”

“Finding people who was born in 1976 and his birth place is a city built on 1718.”

SPARQL

```
SELECT ?name
WHERE {
?m <bornIn> ? c i t y .
?m <hasName> ?name .
?m <bornOnDate> ?bd .
? c i t y <foundingYear> ` `1718 ' ' .
FILTER( regex (str (?bd ), “1 9 7 6 ' ' ) )
}
```

Interdisciplinary Research

Database

RDF Database

Data Integration 、 Knowledge Fusion

Natural Language Processing

Information Extraction

Semantic Parsing

Machine Learning

Knowledge Representation (Graph Embedding)

Knowledge Engineering

KB construction

Rule-based Reasoning

Knowledge Engineering

KB construction

[Mendes et al. 12; Suchanek et al. 07; Bollacker]

Leipzig University
University of Mannheim
OpenLink Software

1.1 Billion Triples

Max-Planck-Institute

180 Million Triples

Metaweb Company,
acquired by Google in 2010

2.5 Billion Triples

Natural Language Processing

Semantic Parsing [Zettlemoyer et al., UAI 05]

Transforming natural language (NL) sentences into computer executable complete meaning representations (MRs) for domain-specic applications.

E.g., “Which states borders New Mexico ?”

Lambda-**calculus** [Alonzo Church, 1940]

$$\lambda x.state(x) \wedge borders(x, new\_mexico)$$

“**Simply typed Lambda-calculus** can express varies database query languages such as **relational algebra**, fixpoint logic and the complex object algebra." [Hillebrand et al., 1996]

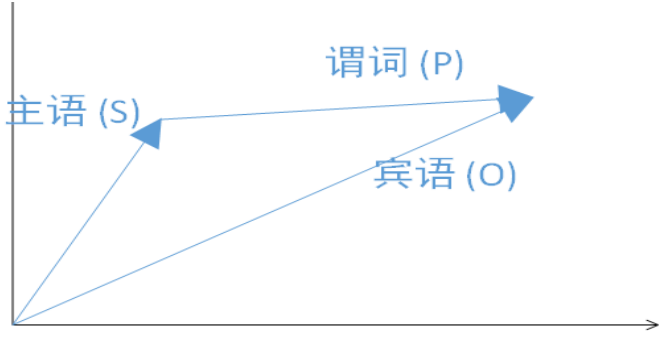
Machine Learning

Knowledge Representation: TransE [Bordes et al., NIPS 13]

- For each triple (Subject,Predicate,Object), “Predicate” as a **translation** from Subject to Object
- Each Subject/Predicate/Object in KG maps to a multidimension vectors
- Objective: S+P=O

S	P	O
China	Capital	Beijing
Canada	Capital	Ottawa
.....	.....	.....

$$\Gamma = \sum_{(s,p,o)\in S} \sum_{(s',p',o')\notin S} [r + d(s+p,o) - d(s'+p',o')]_+$$



$$\text{Beijing} - \text{China} \approx \text{Ottawa} - \text{Canada} = \text{Capital}$$

Database

A Fundamental Problem: How to store RDF data and answer SPARQL queries

Subject	Predicate	Object
Abraham_Lincoln	hasName	“Abraham Lincoln"
Abraham_Lincoln	BornOnDate	“1809-02-12"
Abraham_Lincoln	DiedOnDate	“1865-04-15"”
Abraham_Lincoln	DiedIn	Washington_DC
Abraham_Lincoln	bornIn	Hodgenville KY
Reese_Witherspoon	bornOnDate	"1976-03-22"
Reese_Witherspoon	bornIn	New_Orleans_LA
New_Orleans_LA	foundingYear	“1718”
New Orleans LA	locatedIn	United_States
United_States	hasName	“United States”
United_States	hasCapital	Washington_DC
United_States	foundingYear	“1776”

DBpeida and Freebase have more than billions of triples

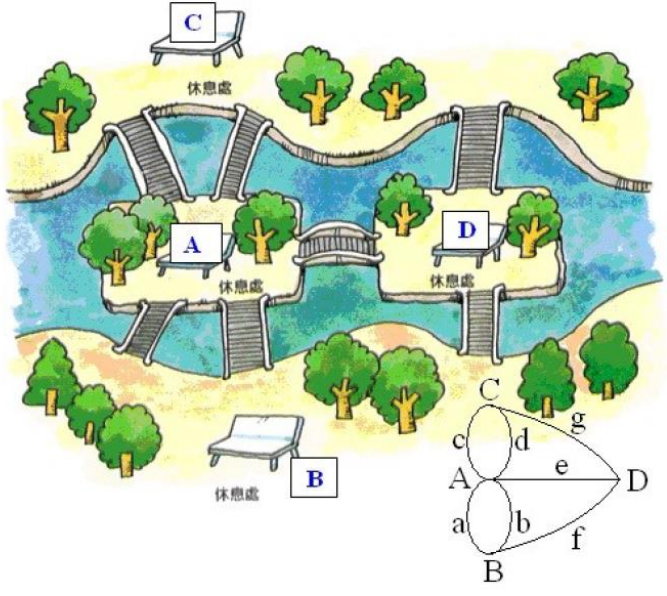
SPARQL

```
SELECT ?name
WHERE {
?m <bornIn> ? c i t y .
?m <hasName> ?name .
?m <bornOnDate> ?bd .
? c i t y <foundingYear> ` `1718 ' ' .
FILTER( regex (str (?bd ), “1 9 7 6 ' ' ) )
}
```

How to answer SPARQL efficiently.

Graph

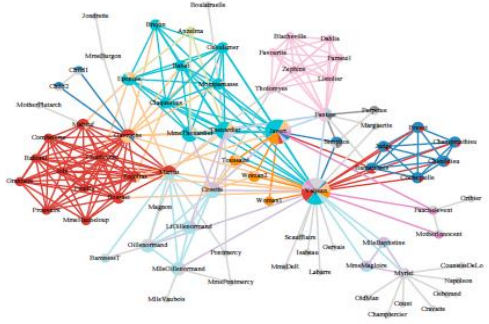
[Seven Bridges of Knigsberg, 1736] The problem is to devise a walk across each of the seven bridges once and only once to touch every part of the town; or this walk does not exist.



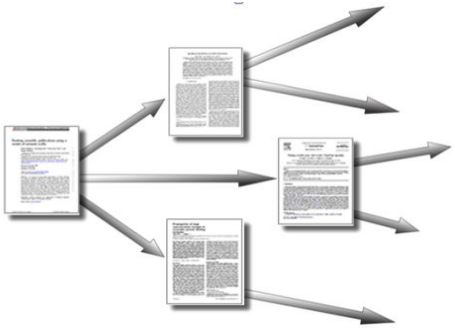
Leonhard Euler,
[1707-1783]

Graph

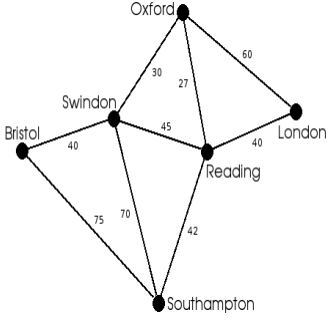
Graph is everywhere:



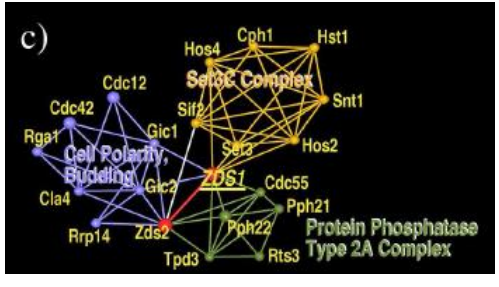
Social Network



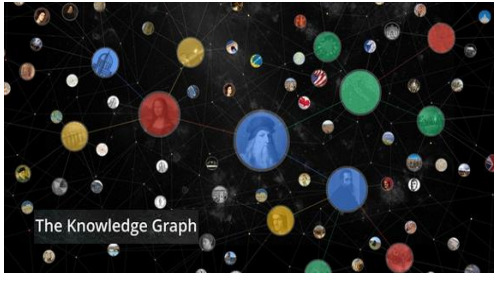
Citation Network



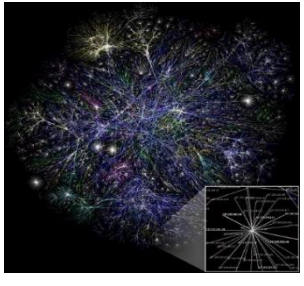
Road Network



Protein Network



Knowledge Graph



Internet

Graph computing is different from traditional computing task.

	TOP 500 The List.	GRAPH 500
Benchmark	Solving a dense n by n system of linear equations Ax = b	BFS search over a large graph
Measure	floating point computing power (TFlops/s).	GTEPS (giga-traversed edges per second).
Applications	Engineering computing	data-intensive workloads

Graph computing is different from traditional computing task.

TOP 10 Sites for November 2017

For more information about the sites and systems in the list, click on the links or view the complete list.

1-100 | 101-200 | 201-300 | 301-400 | 401-500

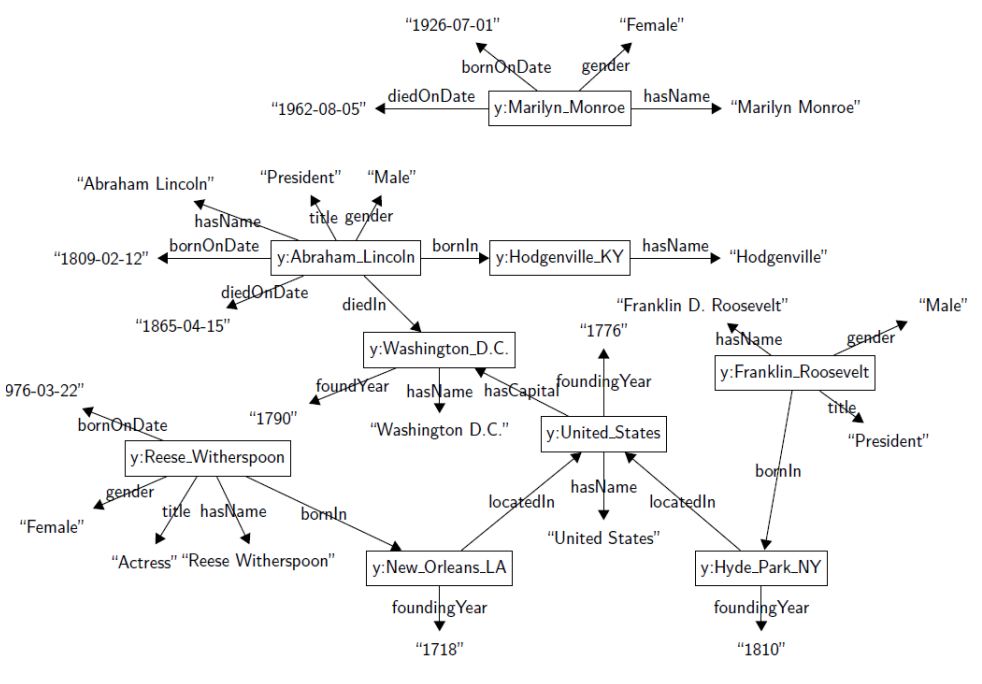
Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway , NRCPC National Supercomputing Center in Wuxi China	10,649,600	93,014.6	125,435.9	15,371
2	**Tianhe-2 (MilkyWay-2)** - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.200GHz, TH Express-2, Intel Xeon Phi 31S1P , NUDT National Super Computer Center in Guangzhou China	3,120,000	33,862.7	54,902.4	17,808
3	**Piz Daint** - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interconnect , NVIDIA Tesla P100 , Cray Inc. Swiss National Supercomputing Centre (CSCS) Switzerland	361,760	19,590.0	25,326.3	2,272
4	**Gyoukou** - ZettaScaler-2.2 HPC system, Xeon D-1571 16C 1.3GHz, Infiniband EDR, PEZY-SC2 700Mhz , ExaScaler Japan Agency for Marine-Earth Science and Technology Japan	19,860,000	19,135.8	28,192.0	1,350
5	**Titan** - Cray XK7, Opteron 6274 16C 2.200GHz, Cray Gemini interconnect, NVIDIA K20x , Cray Inc. DOE/SC/Oak Ridge National Laboratory United States	560,640	17,590.0	27,112.5	8,209
6	**Sequoia** - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom , IBM DOE/NNSA/LLNL	1,572,864	17,173.2	20,132.7	7,890

Top Ten from November 2017 BFS

RANK	MACHINE	VENDOR	INSTALLATION SITE	LOCATION	COUNTRY	YEAR	NUMBER OF NODES	NUMBER OF CORES	SCALE	GTEPS
1	K computer	Fujitsu	RIKEN Advanced Institute for Computational Science (AICS)	Kobe Hyogo	Japan	2011	82944	663552	40	38621.4
2	Sunway TaihuLight	NRCPC	National Supercomputing Center in Wuxi	Wuxi	China	2015	40768	10599680	40	23755.7
3	DOE/NNSA/LLNL Sequoia	IBM	Lawrence Livermore National Laboratory	Livermore CA	USA	2012	98304	1572864	41	23751
4	DOE/SC/Argonne National Laboratory Mira	IBM	Argonne National Laboratory	Chicago IL	USA	2012	49152	786432	40	14982
5	JUQUEEN	IBM	Forschungszentrum Juelich (FZJ)	Juelich	Germany	2012	16384	262144	38	5848
6	ALCF Mira - 8192 partition	IBM	Argonne National Laboratory	Chicago IL	United States	2012	8192	131072	36	4212

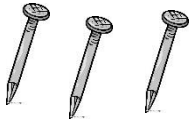
Knowledge “GRAPH”

Subject	Predicate	Object
Abraham_Lincoln	hasName	“Abraham Lincoln"
Abraham_Lincoln	BornOnDate	“1809-02-12"
Abraham_Lincoln	DiedOnDate	“1865-04-15"”
Abraham_Lincoln	DiedIn	Washington_DC
Abraham_Lincoln	bornIn	Hodgenville KY
Reese_Witherspoon	bornOnDate	"1976-03-22"
Reese_Witherspoon	bornIn	New_Orleans_LA
New_Orleans_LA	foundingYear	“1718”
New Orleans LA	locatedIn	United_States
United_States	hasName	“United States”
United_States	hasCapital	Washington_DC
United_States	foundingYear	“1776”



Graph-based RDF Data management

KG problems



- SPARQL Query Evaluation
- Natural Language Question Answering over KG
- Keyword Search over KG
- Semantic Search
- Ontology-based Document Retrieval

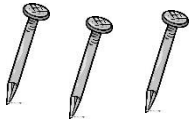
Graph Techniques

- Subgraph Matching
- Bipartite graph matching
- Similarity Subgraph Search
- Random walk-based Similarity Computing

Graph-based RDF Data management

Our Solution

KG problems



Graph Techniques

SPARQL Query Evaluation

Natural Language Question Answering over KG

Keyword Search over KG

Semantic Search

Ontology-based Document Retrieval

Subgraph Matching

Bipartite graph matching

Similarity Subgraph Search

Random walk-based Similarity Computing

Subgraph Matching-based SPARQL Query Evaluation

A Fundamental Problem: How to store RDF data and answer SPARQL queries

Subject	Predicate	Object
Abraham_Lincoln	hasName	“Abraham Lincoln"
Abraham_Lincoln	BornOnDate	“1809-02-12"
Abraham_Lincoln	DiedOnDate	“1865-04-15"”
Abraham_Lincoln	DiedIn	Washington_DC
Abraham_Lincoln	bornIn	Hodgenville KY
Reese_Witherspoon	bornOnDate	"1976-03-22"
Reese_Witherspoon	bornIn	New_Orleans_LA
New_Orleans_LA	foundingYear	“1718”
New Orleans LA	locatedIn	United_States
United_States	hasName	“United States”
United_States	hasCapital	Washington_DC
United_States	foundingYear	“1776”

DBpeida and Freebase have more than billions of triples

SPARQL

```
SELECT ?name
WHERE {
?m <bornIn> ? c i t y .
?m <hasName> ?name .
?m <bornOnDate> ?bd .
? c i t y <foundingYear> ` `1718 ' ' .
FILTER( regex (str (?bd ), “1 9 7 6 ' ' ) )
}
```

How to answer SPARQL efficiently.

Existing Solutions: Resorting to **RDBMS** techniques

Subject	Predicate	Objects
Abraham_Lincoln	hasName	“Abraham Lincoln"
Abraham_Lincoln	BornOnDate	“1809-02-12"
Abraham_Lincoln	DiedOnDate	“1865-04-15"”
Abraham_Lincoln	DiedIn	Washington_DC
Abraham_Lincoln	bornIn	Hodgenville KY
Reese_Witherspoon	bornOnDate	"1976-03-22"
Reese_Witherspoon	bornIn	New_Orleans_LA
New_Orleans_LA	foundingYear	“1718”
New Orleans LA	locatedIn	United_States
United_States	hasName	“United States”
United_States	hasCapital	Washington_DC
United_States	foundingYear	“1776”

SPARQL

```
SELECT ?name
WHERE {
?m <bornIn> ? c i t y .
?m <hasName> ?name .
?m <bornOnDate> ?bd .
? c i t y <foundingYear> ` `1718 ' ' .
FILTER( regex (str (?bd ), “1 9 7 6 ' ' ) )
}
```

SQL

```
SELECT T2 . o b j e c t
FR
W
AND T2.prope          sName"
AND T3.proper         "bornOnDate "
AND T1.subject=T2.subject
AND T2.subject=T3.subject
AND T1.object=T4.subject
AND T4.propety=“foundingYear “
AND T4.object=" 1718 "
AND T3.object LIKE '%1976%'
```

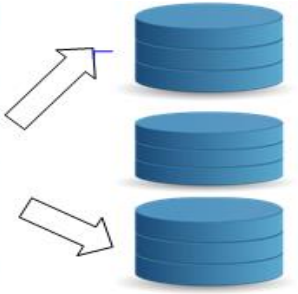
Too many self-joins

Existing Solutions (based on RDBMS techniques)

- **Property Table** Jena [Wilkinson et al., 2003] ,FlexTable [Wang et al., 2010] , DB2-RDF [Bornea et al., 2013]
- **Vertically partitioned tables** SW-store [Abadi et al., 2009]
- **Exhaustive indexing** RDF-3X [Neumann and Weikum, 2008], Hexastore [Weiss et al., 2008]

Basic Ideas: dividing the large single triple-table into several carefully-designed tables.

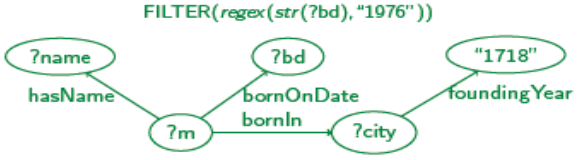
主语	谓词	宾语
Abraham_Lincoln	hasName	"Abraham Lincoln"
Abraham_Lincoln	BornOnDate	"1809-02-12"
Abraham_Lincoln	DiedOnDate	"1865-04-15"^^
Abraham_Lincoln	DiedIn	Washington_DC
Abraham_Lincoln	bornIn	Hodgenville KY
Reese_Witherspoon	bornOnDate	"1976-03-22"
Reese_Witherspoon	bornIn	New_Orleans_LA
New_Orleans_LA	foundingYear	"1718"
New Orleans LA	locatedIn	United_States
United_States	hasName	"United States "@
United_States	hasCapital	Washington_DC
United_States	foundingYear	"1776"

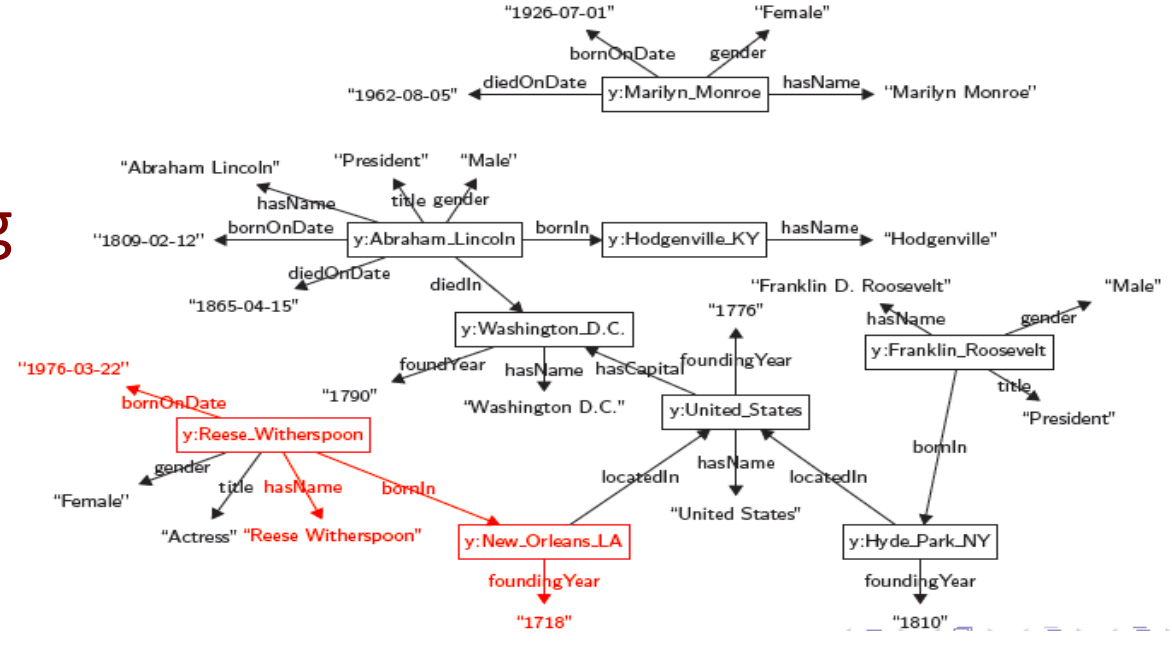


- M. T. Özsu. "A Survey of RDF Data Management Systems", Front. Comp. Sci., 2016.
- Lei Zou, M. T. Özsu. “Graph-based RDF Data Management”, Data Science and Engineering, 2(1): 56-70 (2017)

Our Solution---gStore
[Zou et al., VLDB 11; VLDB J 14]

Answering SPARQL
== subgraph matching





Our Solution---gStore [Zou et al., VLDB 11; VLDB J 14]

Main Techniques:

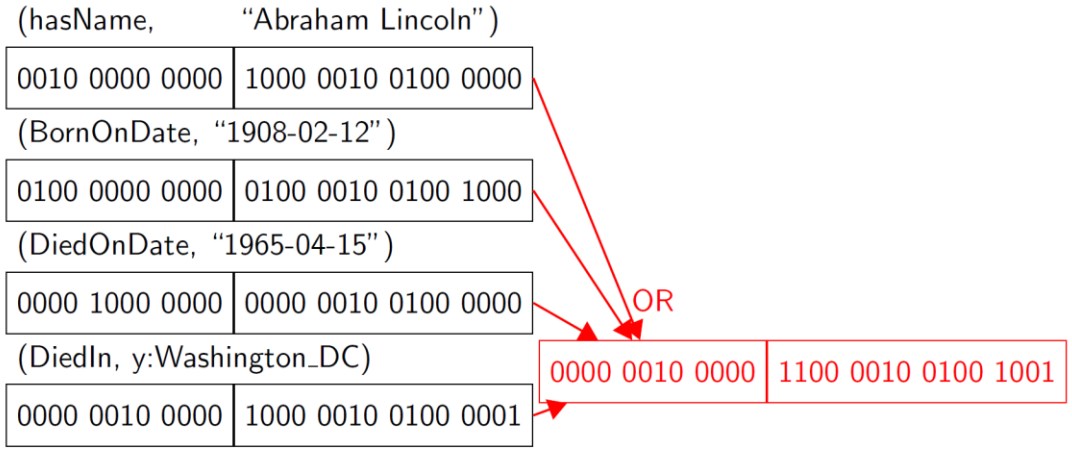
- Store RDF graph G as adjacency lists;
- Neighborhood Structure Summarization—Encoding
- Structure-aware Index– VS*-tree.

Our Solution---gStore

Why Encoding Neighborhood ?

Encoding Technique

vLabel	adjList
y:Abraham_Lincoln	(hasName, "Abraham Lincoln"), (BornOnDate, "1809-02-12"), (DiedOnDate, "1865-04-15"), (DiedIn, y:Washington_DC)

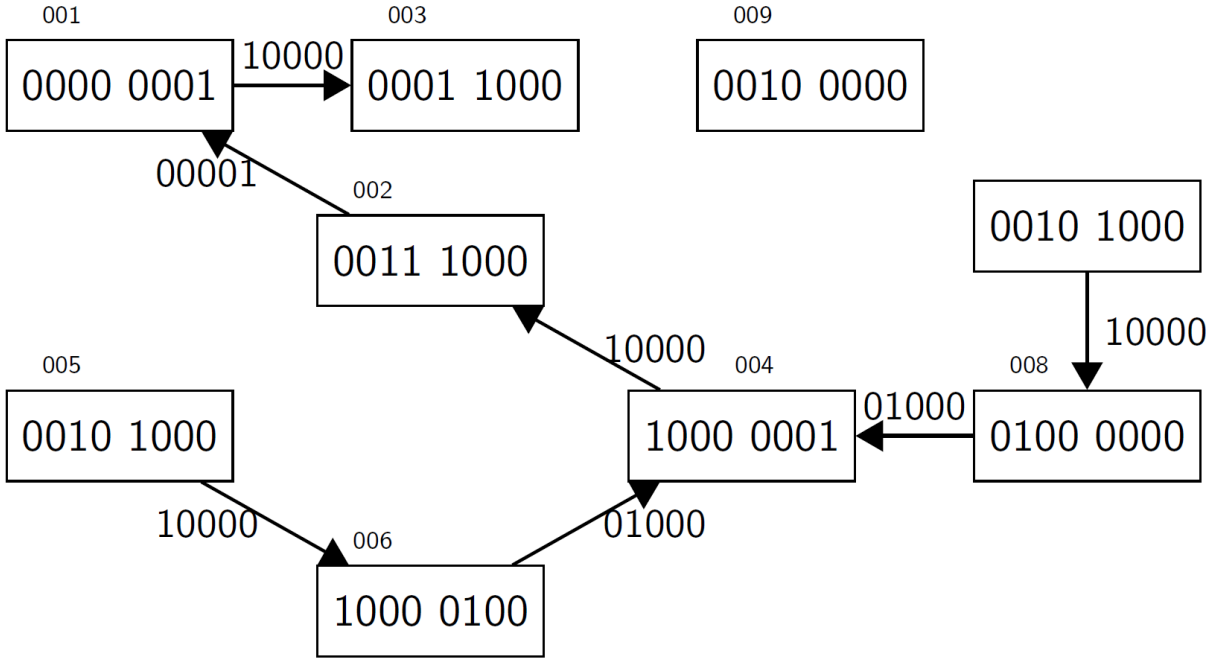


Neighborhood Pruning:

If a vertex u in query graph Q can match a vertex v in data graph G, then any neighbor of vertex u should match one neighbor of vertex v; Otherwise, u cannot match v.

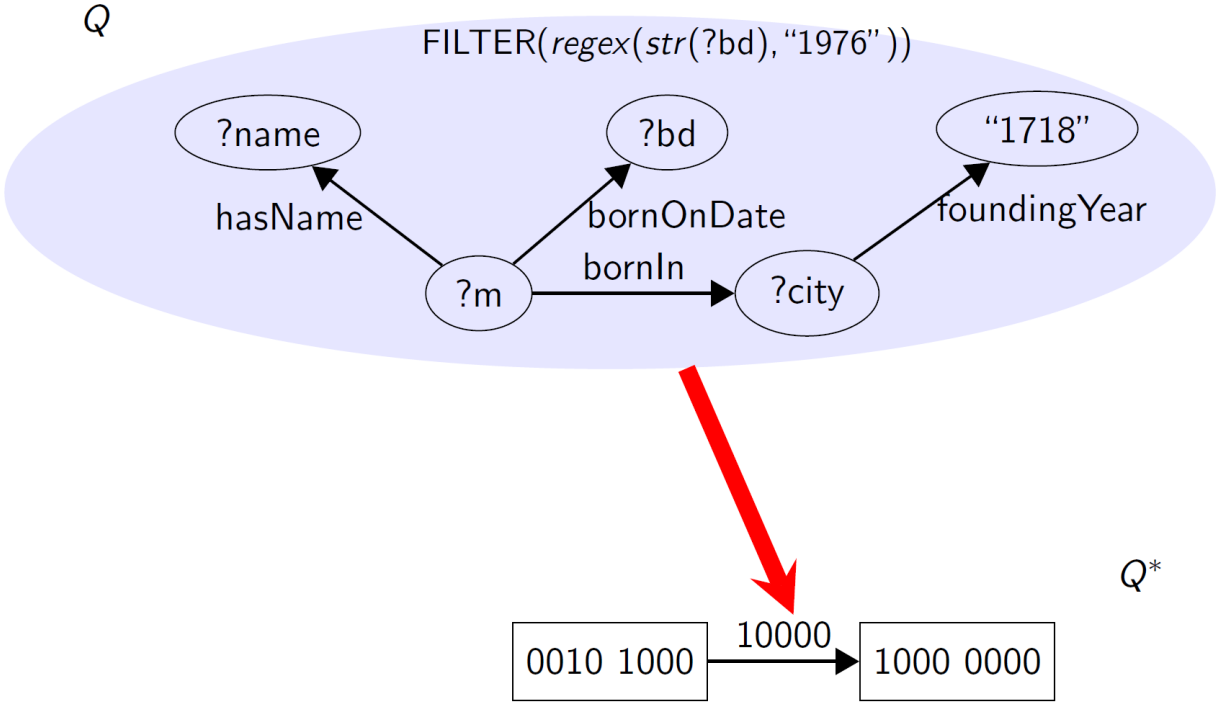
Our Solution---gStore

2. Construct Data Signature Graph G^*



Our Solution---gStore

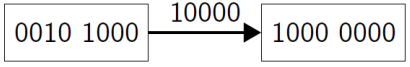
3. Encode Q to Get Signature Graph Q^*



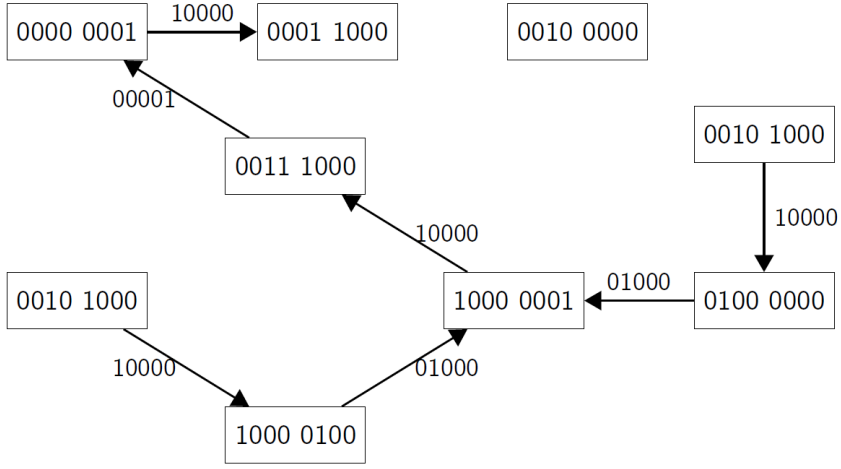
Our Solution---gStore

4. Filter-and-Evaluate

Query signature graph Q^*



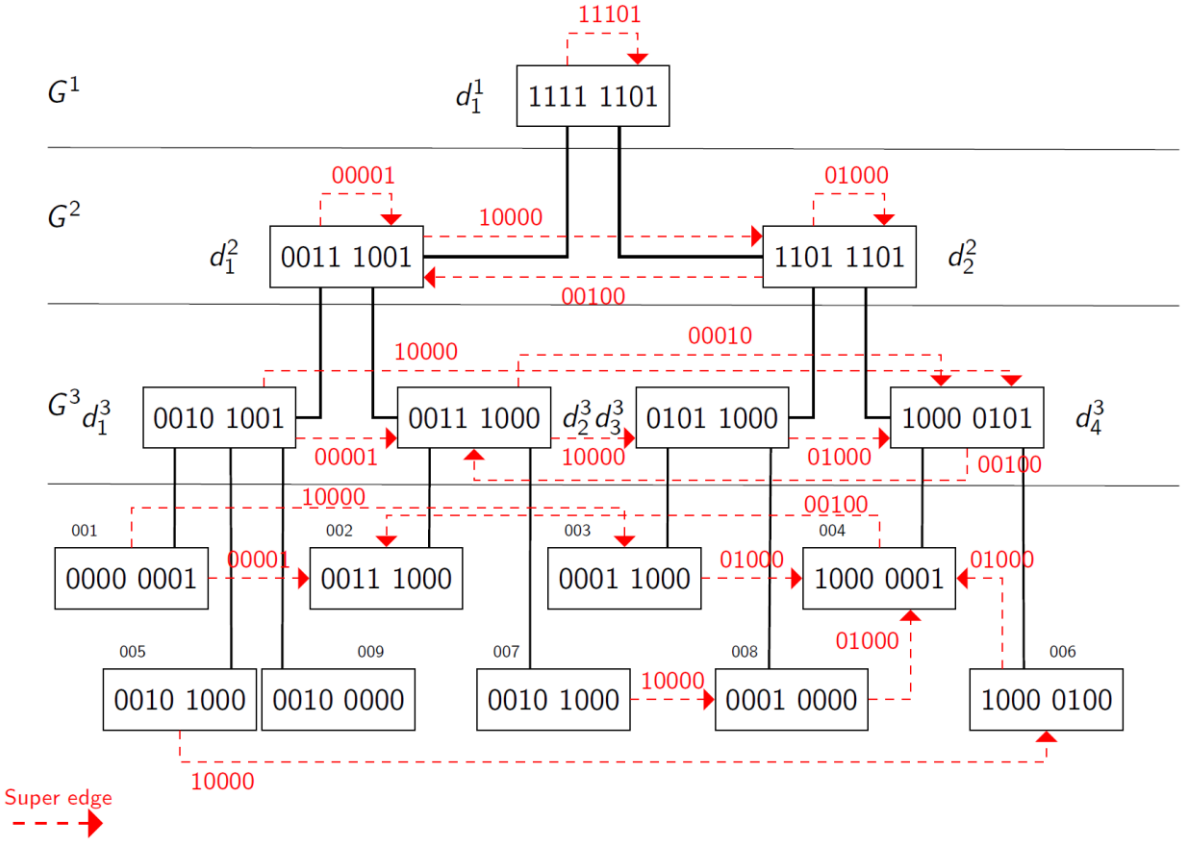
Data signature graph G^*



Find matches of Q^* over signature graph G^* → Verify each match in RDF graph G

Our Solution---gStore

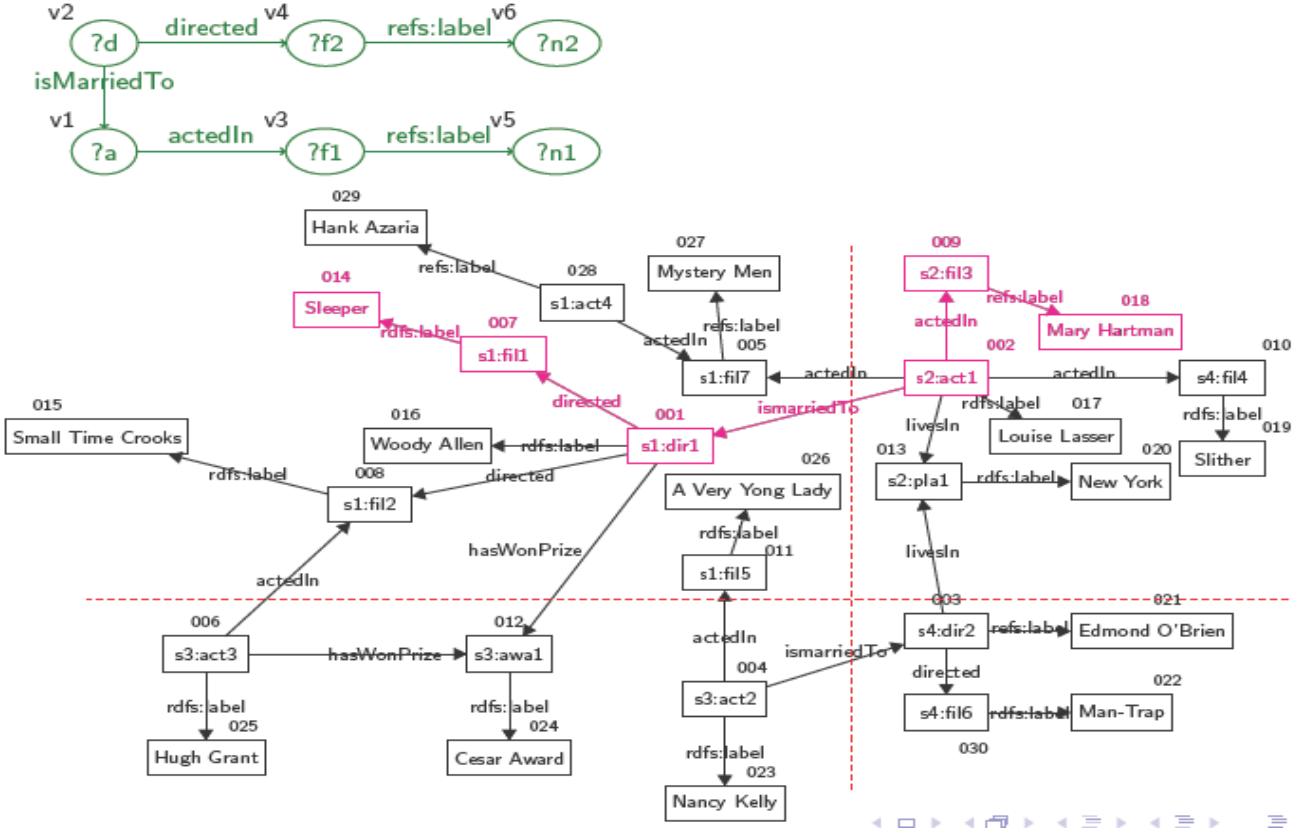
VS-tree



gStore-D: Distributed RDF System

[Peng P, et al., VLDB J 16]

Challenges: How to find "crossing matches"

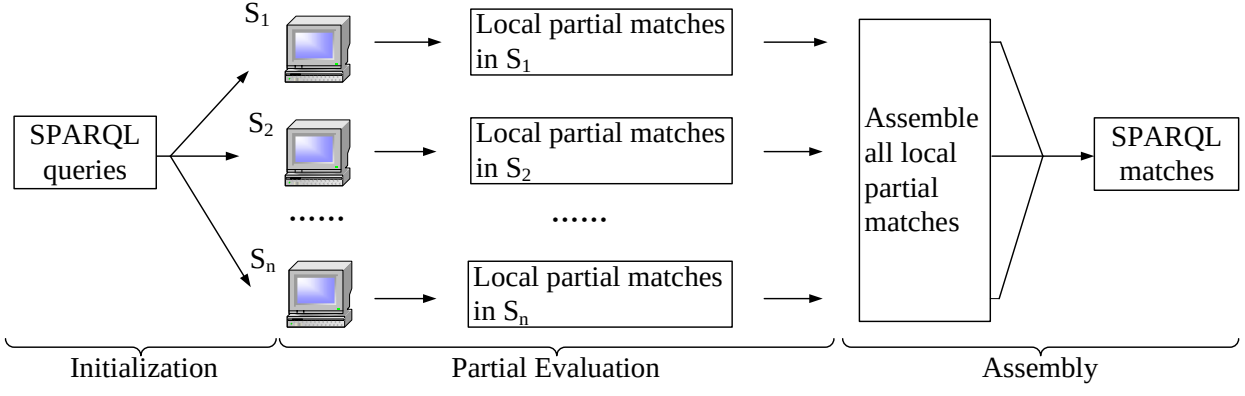


gStore-D: Distributed RDF System

[Peng P, et al., VLDB J 16]

Main Techniques:

- Partial Evaluation and Assembly-based Solution;
- Optimized Assembly Strategy in the distributed circumstance



gStore-D: Distributed RDF System

[Peng P, et al., VLDB J 16]

Background: Partial Evaluation [Jones, 1996; Fan et al., 06; Shuai et al., 2012]

$$f(x) \Rightarrow f(s,d) \Rightarrow f''(f'(s),d) \Rightarrow \text{Final Results}$$

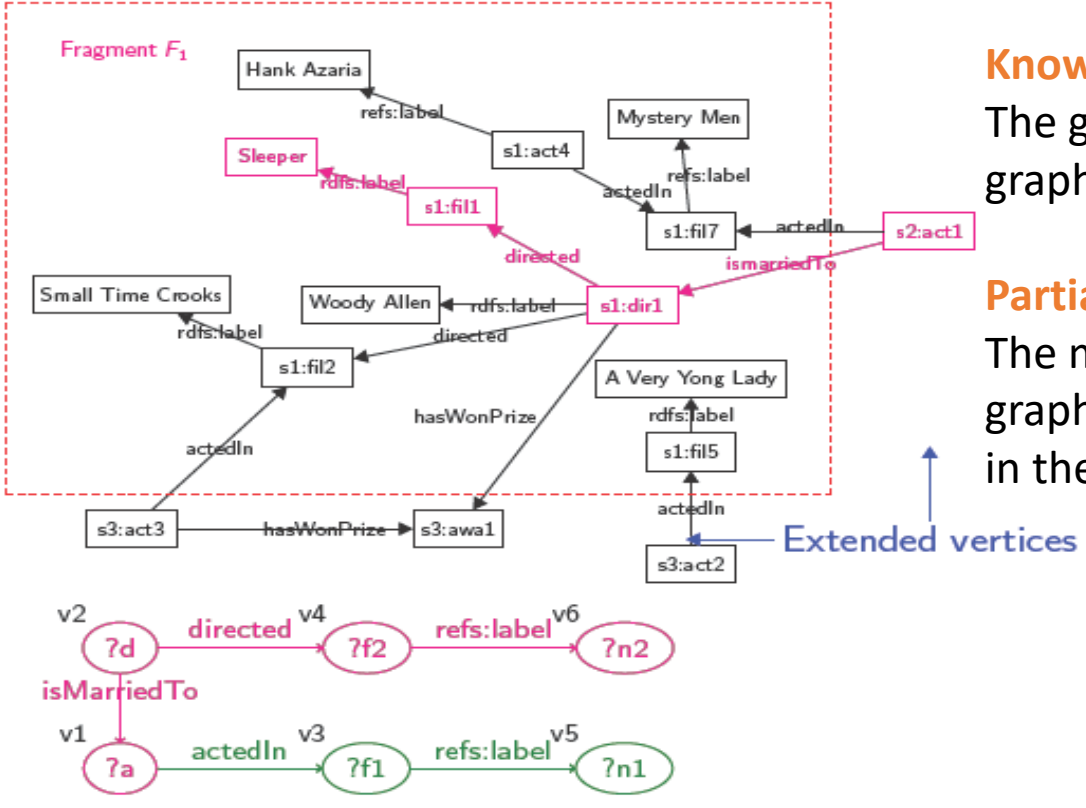
Known Input

Unknown Input

Partial Results

gStore-D: Distributed RDF System

Which are "known inputs" and "partial results" ?



Known inputs:

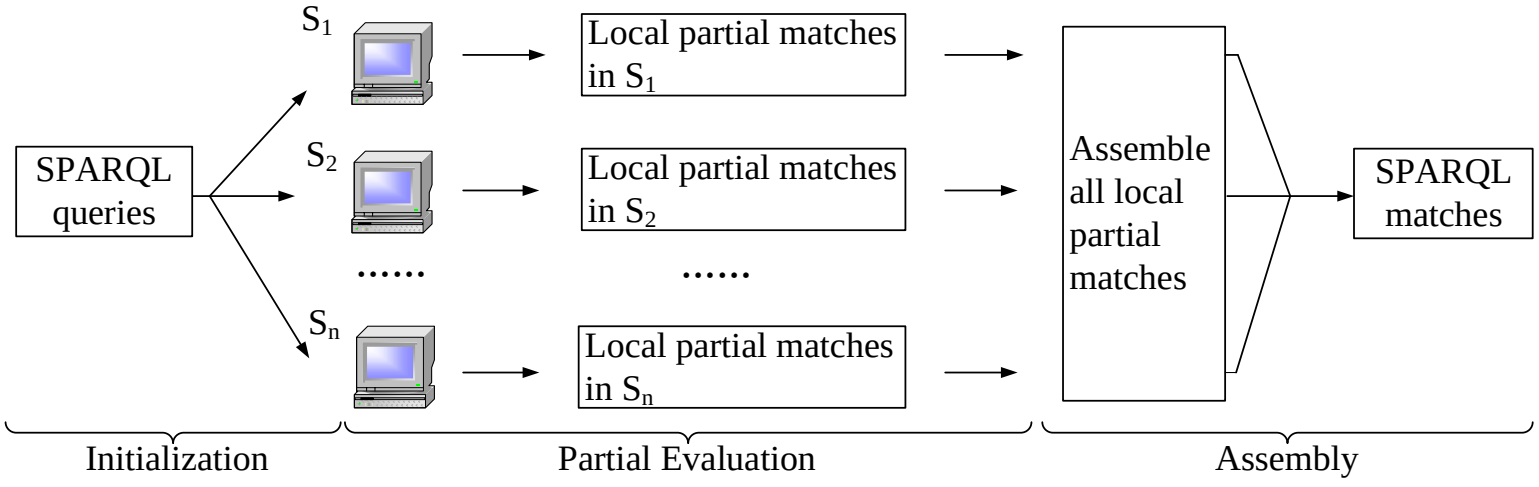
The graph at its own site and the query graph Q.

Partial Results:

The maximal partial matches of query graph Q over its own partial data graph in the site.

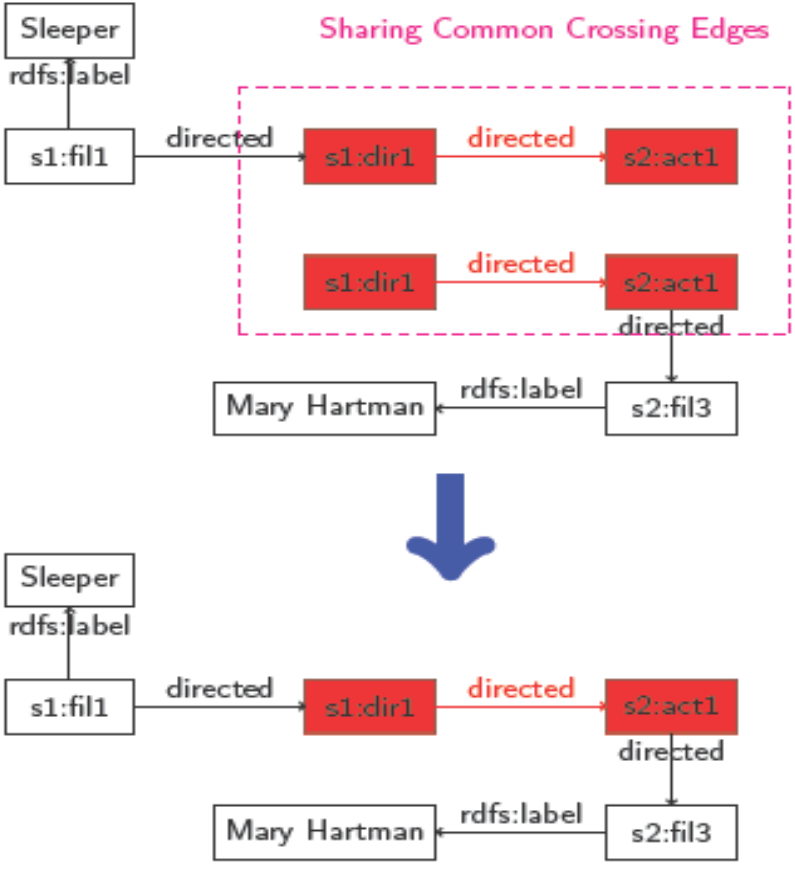
gStore-D: Distributed RDF System

[Peng P, et al., VLDB J 16]



gStore-D: Distributed RDF System

Assembly



Our System

gStore

Codes： More than 140,000 lines C++, coding from scratch

Project Address:

https://github.com/Caesar11/gStore/

including all codes; user manual; benchmarking test report; system demo video.

Licenses: BSD

API: C++, Java, Phython, PHP and HTTP Rest

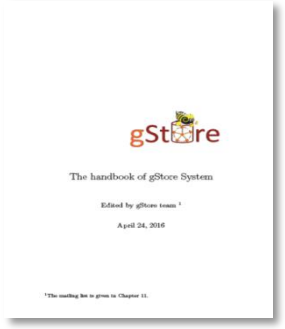
Supporting SPARQL 1.1 (including UNION, OPTIONAL, FILTER, GROUP BY, BIND)

Our System

gStore

Capability: A single site can support big KG with more than FIVE billion edges (e.g., supporting the full version of DBpdida and freebase in a single machine)

Performance: see our system performance report in github.



Endpoints:

http://dbpedia.gstore-pku.com

http://freebase.gstore-pku.com

The Third Part Comments

【Vijay Ingalalli, Dino Ienco, Pascal Poncelet, Serena Villata: Querying RDF Data Using A Multigraph-based Approach. **EDBT 2016**: 245-256】

- LIRMM, IRSTEA
- CNRS, I3S Laboratory

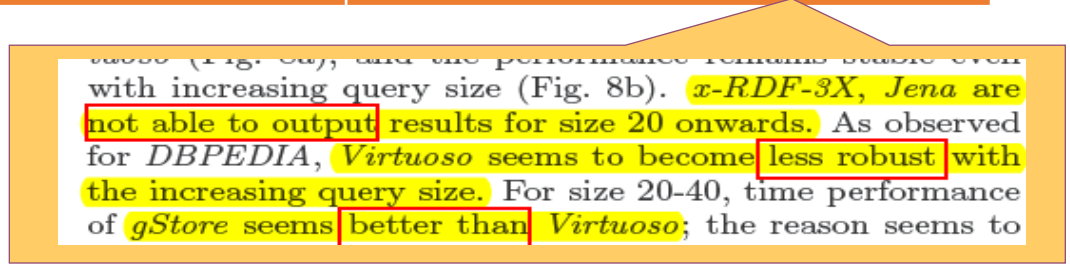
DBpedia	
33 Million Triples	4 Million Vertices

Comparative Systems	Systems' Features	Comments
Apache Jena	Open Source RDF Database; original from HP Lab	"x-RDF-3x, Jena are not able to output results for size 20 onwards".
x-RDF-3x	Influential academic system, from Max-Planck-Institute	
Virtuoso	Commercial System	"Virtuoso seems to become less robust with the increasing query size"
gStore (Our System)	Open Source System at Github 【Zou et al., VLDB 2011】	"the time performance of gStore seems better than Virtuoso"

gStore	Virtuoso	RDF-3x
11.96 (sec)	20.45 (sec)	>60 (sec)

Average Time (seconds) for a sample of 200 complex queries on DBPEDIA.

[Ingalalli et. EDBT 16]


with increasing query size (Fig. 8b). *x-RDF-3X*, *Jena* are not able to output results for size 20 onwards. As observed for *DBPEDIA*, *Virtuoso* seems to become less robust with the increasing query size. For size 20-40, time performance of *gStore* seems better than *Virtuoso*; the reason seems to

gStore Application

- Institute of Microbiology, CAS –
- World Data Center for Microorganisms

# of Triples	# of Entities
3,594,457,749	414,953,654

Bacteria > Terrabacteria group > Actinobacteria > Actinobacteria > Micrococcales > Micrococcaceae > Micrococcus > Micrococcus luteus

细菌 陆生菌 放线菌门 放线菌纲 微球菌目 微球菌科 微球菌属 藤黄微球菌

Overview Taxonomy Genome Feature GO Pathway Literature

Species Information

Taxonomy	Bacteria > Terrabacteria group > Actinobacteria > ... > Micrococcus > Micrococcus luteus
NCBI taxonomy ID	1270
Scientific Name	Micrococcus luteus
Children	Micrococcus luteus CD1_FAA_NB_1 Micrococcus luteus J28 Micrococcus luteus Mu201 Micrococcus luteus NCTC 2665 Micrococcus luteus SK58 Micrococcus luteus str. modasa More
Reference Title In IJSEM	
Type Strains	
Strains	

```
PREFIX annotation:
<http://gcm.wdcm.org/ontology/gcmAnnotation/v1/>
PREFIX taxonomy:
<http://gcm.wdcm.org/data/gcmAnnotation1/taxonomy/>

SELECT ?taxonId ?name
WHERE
{
        ?taxonId  annotation:parentTaxid  taxonomy:1270.
        ?nameId  annotation:taxid  ?taxonId.
        ?nameId  annotation:nameclass  'scientificName'.
        ?nameId  annotation:taxname  ?name.
}
```

"searching **strains** of **Micrococcus luteus**"

gStore Application

- Institute of Microbiology, CAS –
- World Data Center for Microorganisms

# of Triples	# of Entities
3,594,457,749	414,953,654

```
PREFIX annotation:
<http://gcm.wdcm.org/ontology/gcmAnnotation/v1/>
PREFIX taxonomy:
<http://gcm.wdcm.org/data/gcmAnnotation1/taxonomy/>

SELECT (COUNT(?geneid) AS ?num)
WHERE
{
        {       ?taxonid annotation:ancestorTaxid taxonomy:1270.
                ?geneid a annotation:GeneNode.
                ?geneid annotation:x-taxon ?taxonid.
        }UNION
        {       ?geneid a annotation:GeneNode.
                ?geneid annotation:x-taxon taxonomy:1270.
        }
}
```

Number of Gene	54824
Number of Protein	16229

Annotation summary

Proteins with PDB structures	15
Proteins with Pfam assignments	2008
Proteins with GO assignments	32453
Proteins with EC number assignments	680
Proteins with Pathway assignments	2398

Publications and Patents

Publications	
Patents	

"The number of genes related to Micrococcus luteus and its descendants"

gStore Application

- Institute of Microbiology, CAS –
- World Data Center for Microorganisms

# of Triples	# of Entities
3,594,457,749	414,953,654

Genome

Export Excel

Organism Name	Genome Accession	Description
Micrococcus luteus str. modasa	AMYK02000110	Micrococcus luteus str. modasa contig_110, whole genome shotgunsequence.
Micrococcus luteus NCTC 2665	CP001628	Micrococcus luteus NCTC 2665, complete genome.
Micrococcus luteus SK58	ADCD01000097	Micrococcus luteus SK58 ctg1119142780327, whole genome shotgunsequence.
Micrococcus luteus str. modasa	AMYK02000273	Micrococcus luteus str. modasa contig_273, whole genome shotgunsequence.
Micrococcus luteus str. modasa	AMYK02000081	Micrococcus luteus str. modasa contig_81, whole genome shotgunsequence.
Micrococcus luteus str. modasa	AMYK02000252	Micrococcus luteus str. modasa contig_252, whole genome shotgunsequence.
Micrococcus luteus str. modasa	AMYK02000060	Micrococcus luteus str. modasa contig_60, whole genome shotgunsequence.
		Micrococcus luteus str.

" Searching for the genes and descriptions related to Micrococcus luteus and its descendants"

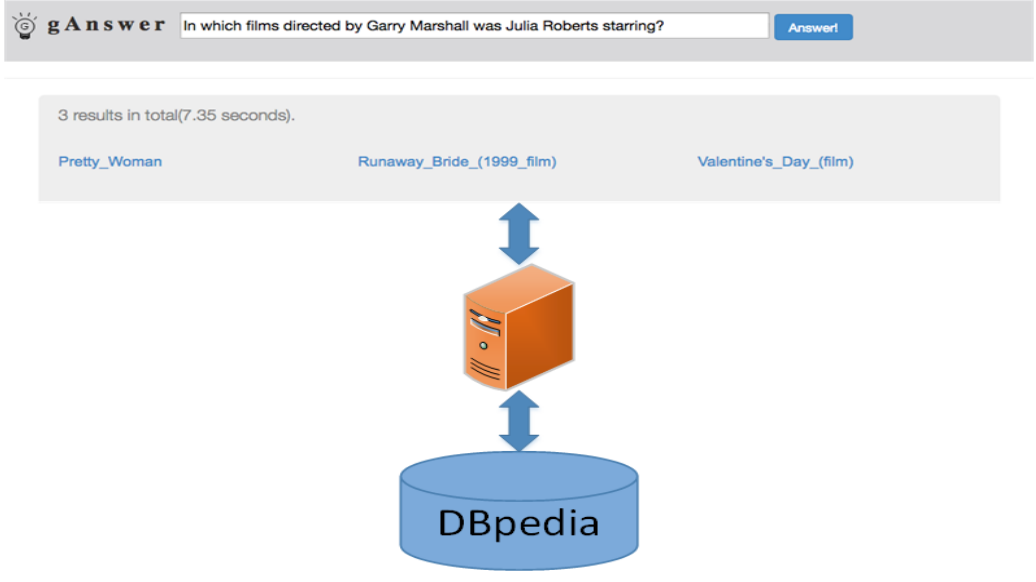
```
PREFIX annotation:
<http://gcm.wdcm.org/ontology/gcmAnnotation/v1/>
PREFIX taxonomy:
<http://gcm.wdcm.org/data/gcmAnnotation1/taxonomy/>

SELECT ?taxonid ?name ?genomeid ?description ?strain
WHERE
{
        ?taxonid annotation:ancestorTaxid taxonomy:1270.
        ?nameId a annotation:TaxonName.
        ?nameId annotation:taxid ?taxonid.
        ?nameId annotation:nameclass 'scientificName'.
        ?nameId annotation:taxname ?name.
        ?genomeid a annotation:GenomeNode.
        ?genomeid annotation:x-taxon ?taxonid.
        ?genomeid annotation:definition ?description.
        optional{?genomeid annotation:strain ?strain.}
}
```

Subgraph Matching-based Natural Language Question/Answering

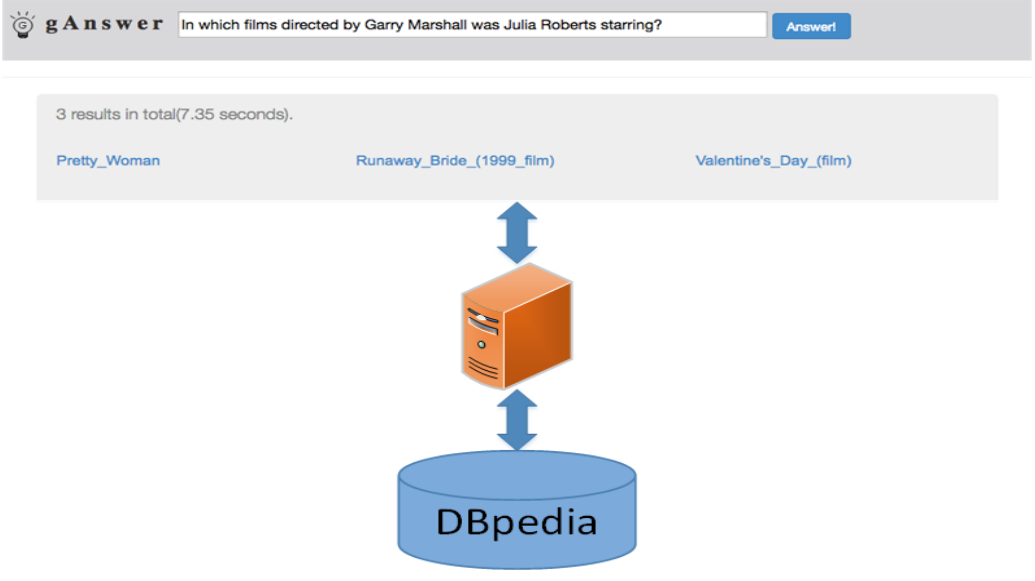
KG-based Question/Answering

- SPARQL syntax are too complex for ordinary users
- RDF KG is“schema-less”data, not like schema-first relational database.



KG-based Question/Answering

- An **Easy-to-Use** Interface to Access Knowledge Graph
- It is interesting to both **academia** and **industry**.
- **Interdisciplinary research** between database and NLP (natural language processing) communities.



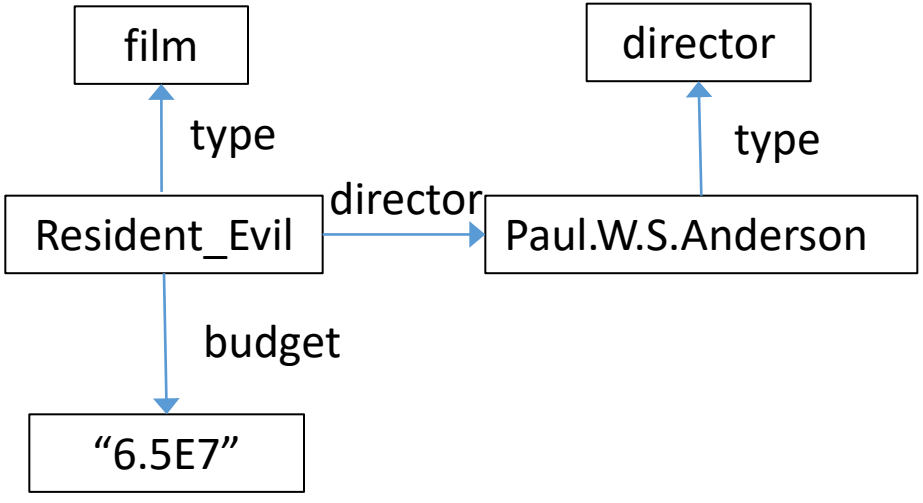
KG-based Question/Answering

- Information Retrieval-based
  - Generate candidate answers
  - Ranking

- Semantic Parsing-based
  - Translate NLQ to logical forms
  - Executing

KG-based Question/Answering

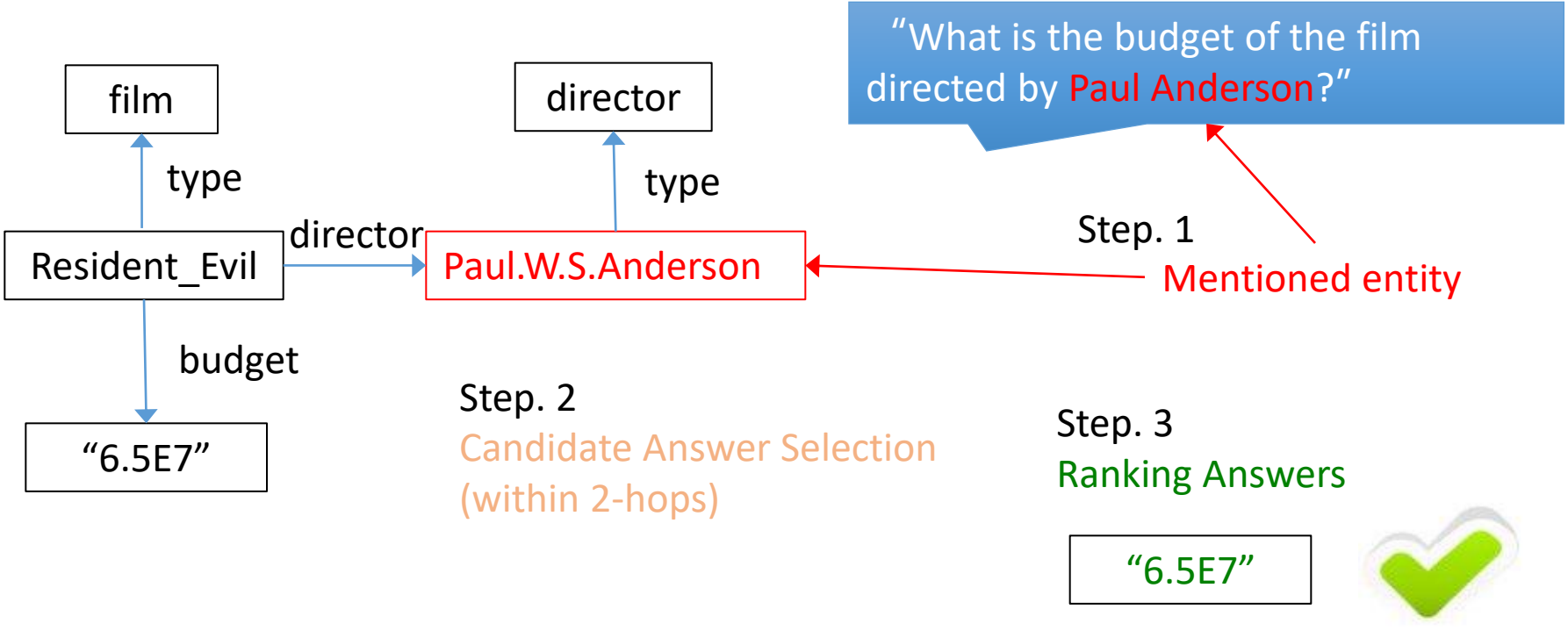
- Information Retrieval-based



“What is the budget of the film directed by Paul Anderson?”

KG-based Question/Answering

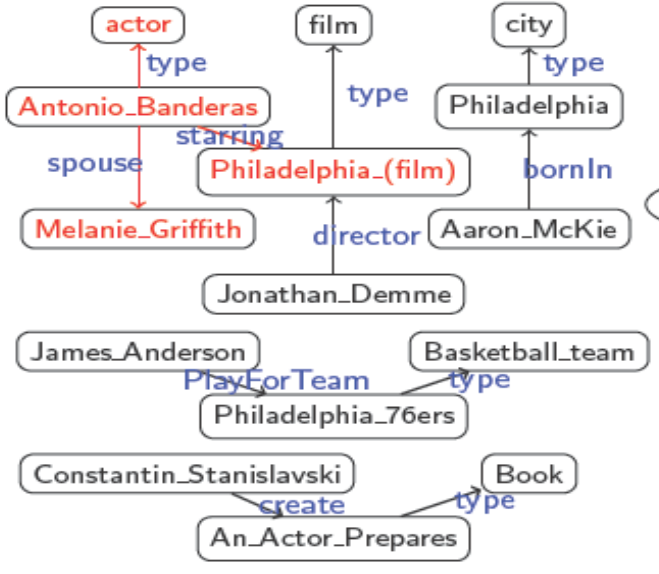
- Information Retrieval-based

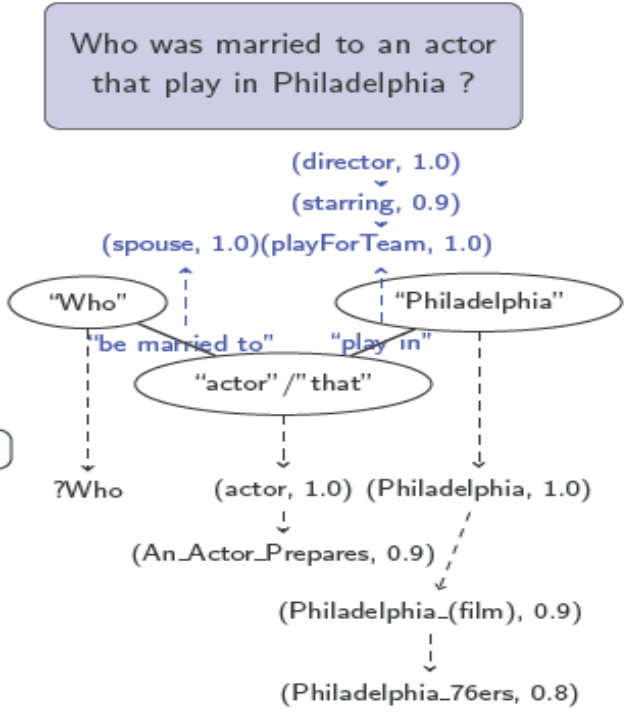


Our Approach- Data Driven

gAnswer [Zou et al, SIGMOD 14]

- Using graph matching-based method
- Graph Matching-based Disambiguation
- Combing Disambiguation and Query together





Our Approach- Data Driven Solution

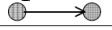
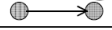
gAnswer

Using two Dictionaries

Entity Name Dictionary: Entity Mention Extraction and Linking

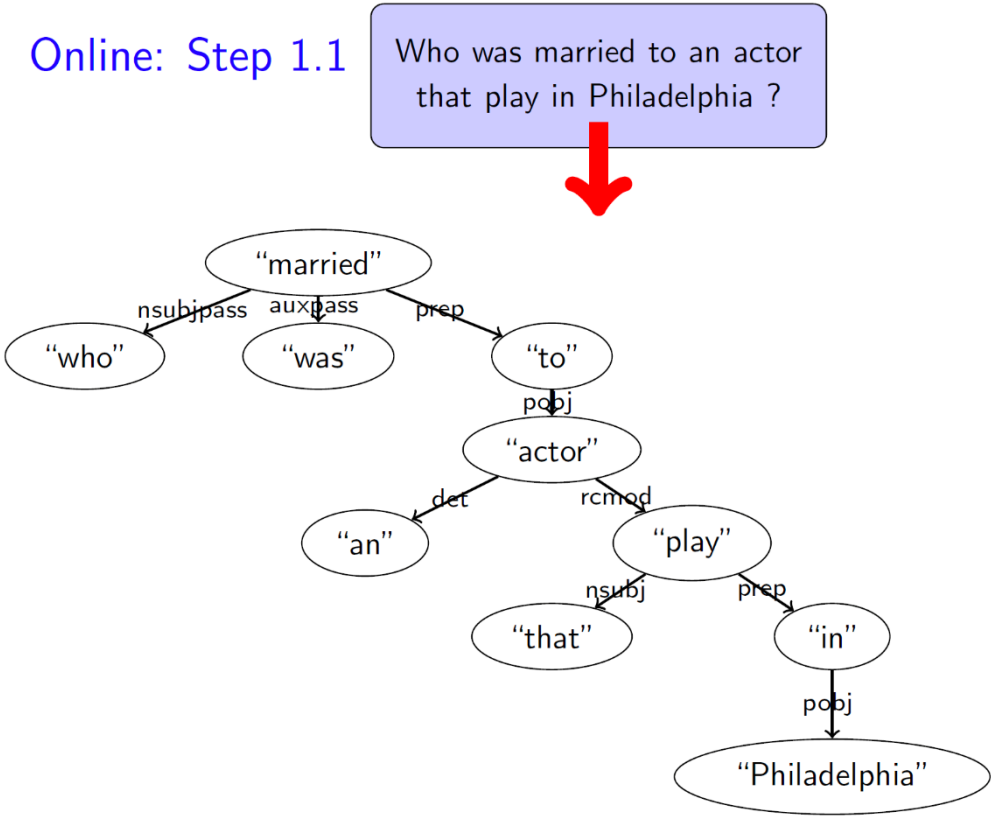
Relation Paraphrasing Dictionary:

Relation Mention Extraction and Mapping

Relation Phrases	Predicates or Predicate Paths	Confidence Probability
"be married to"		1.0
"play in"		0.9
"play in"		0.5
"uncle of"		0.8
… …	… …	… …

Our Approach- Data Driven Solution

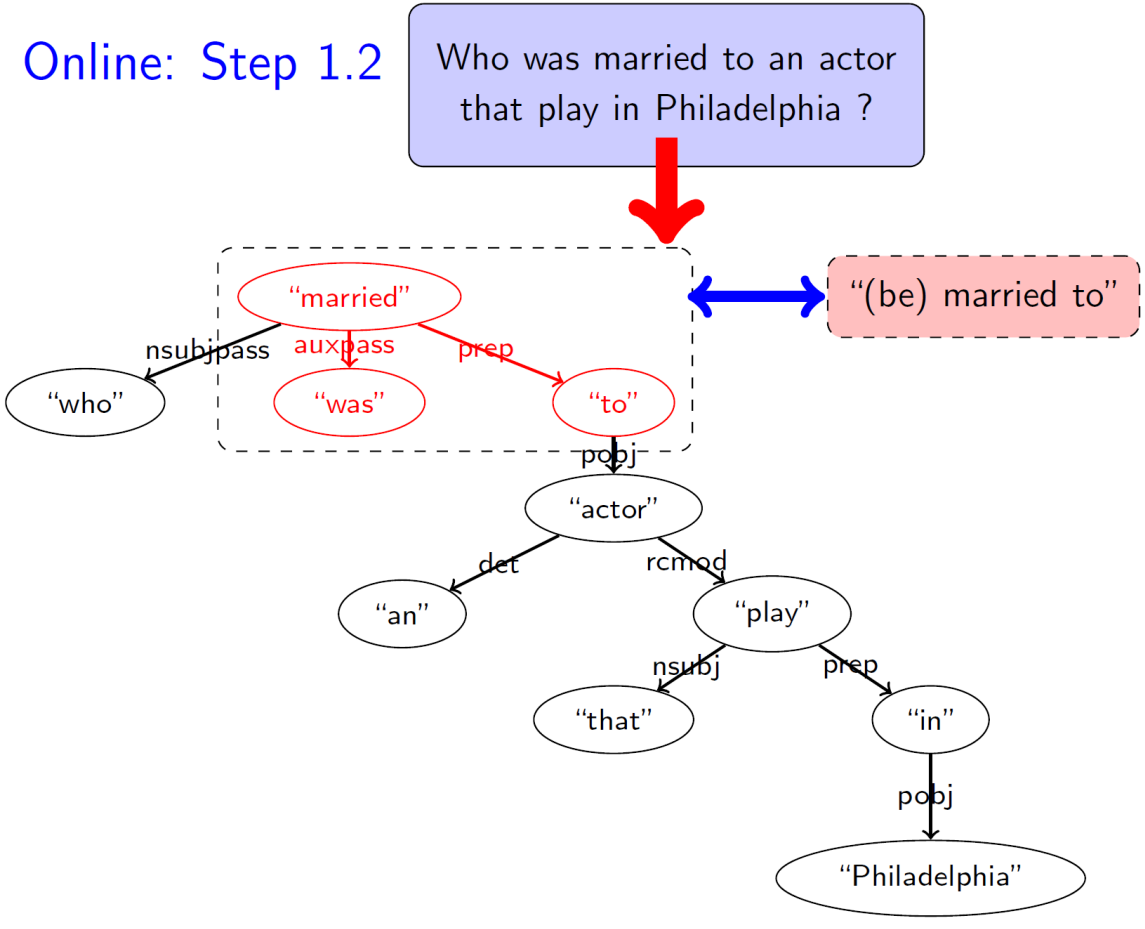
gAnswer



Our Approach- Data Driven Solution

gAnswer

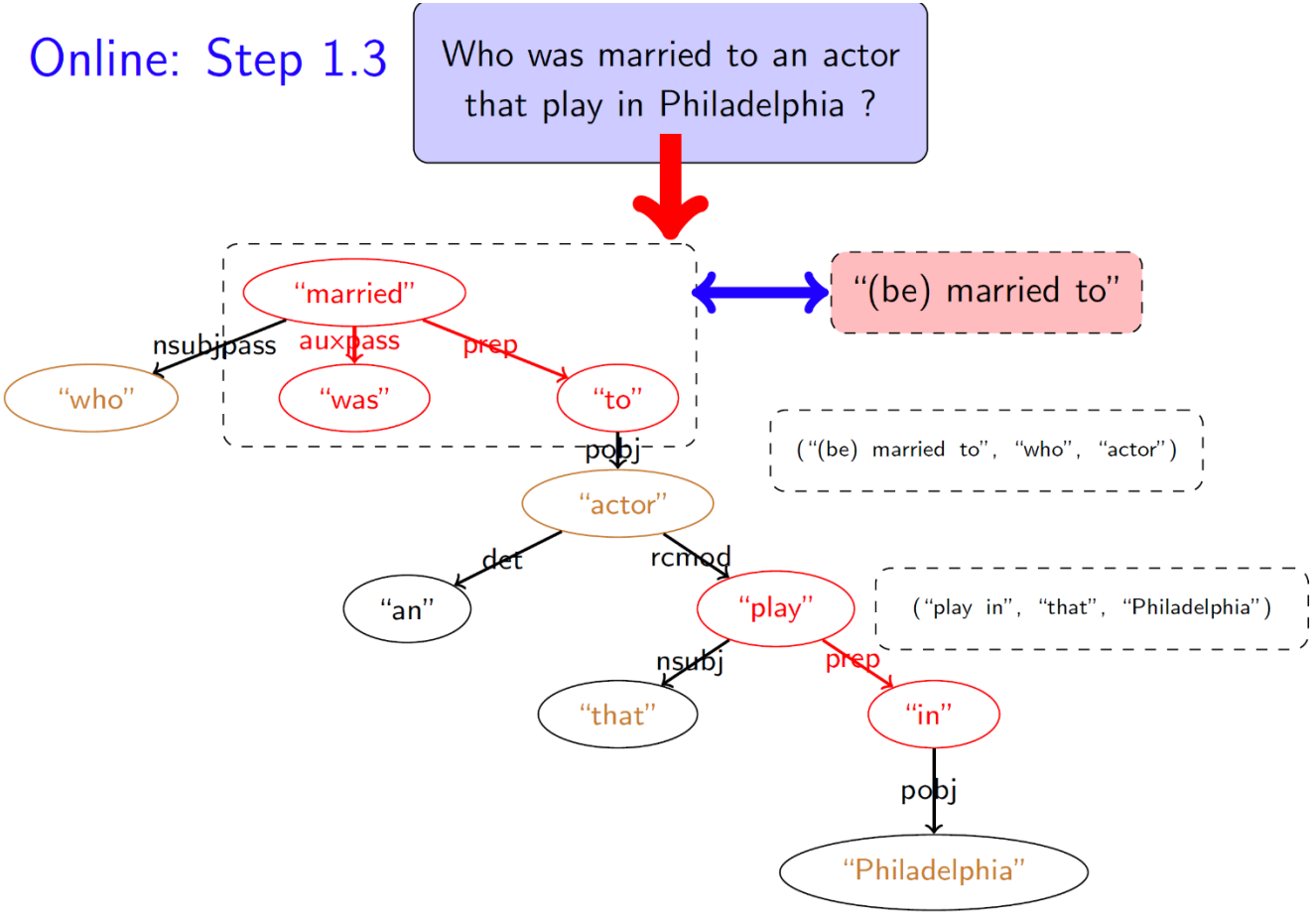
Online: Step 1.2



Our Approach- Data Driven Solution

gAnswer

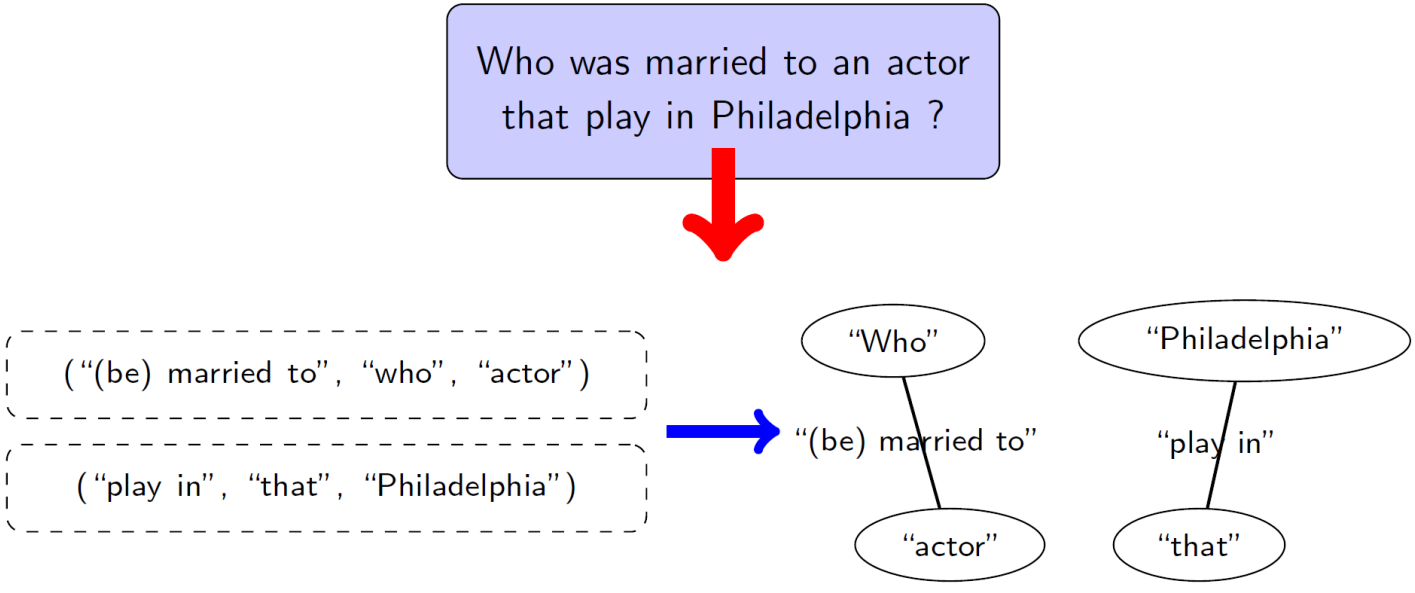
Online: Step 1.3



Our Approach- Data Driven Solution

gAnswer

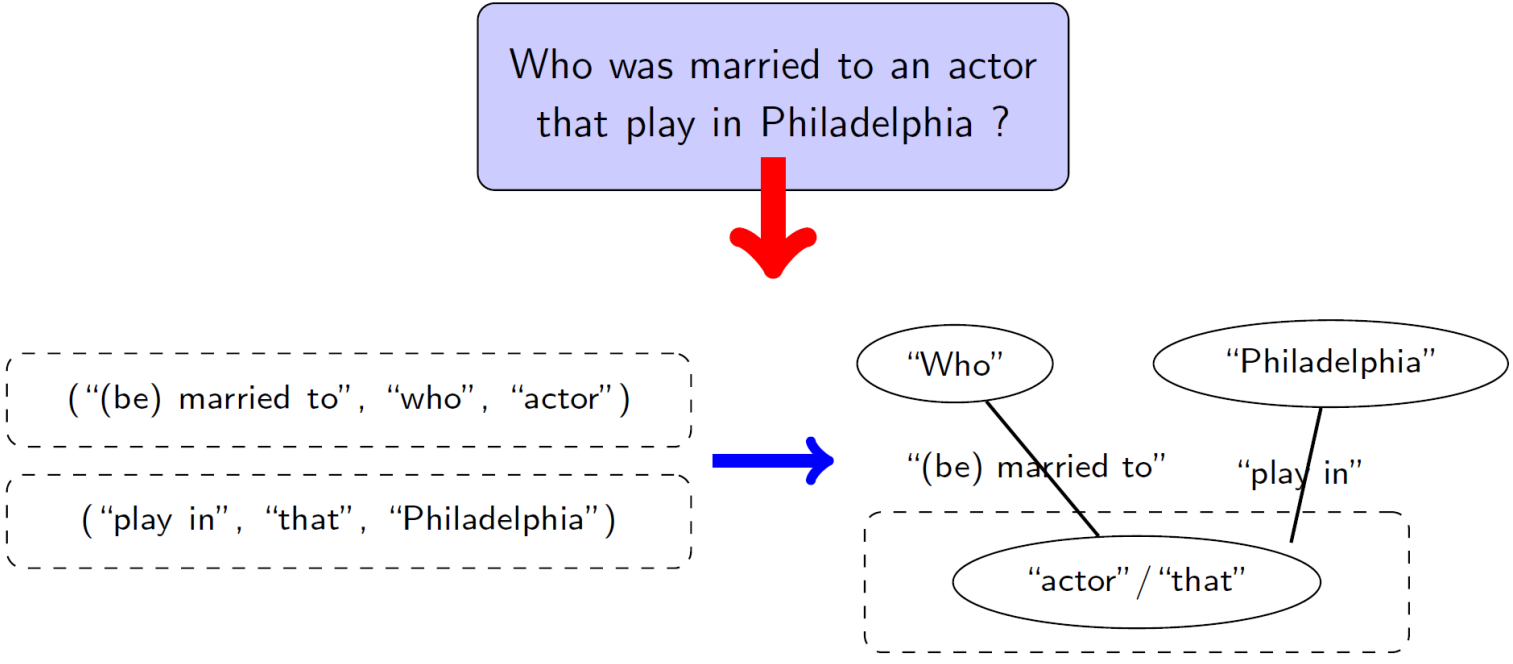
Online: Step 1.4



Our Approach- Data Driven Solution

gAnswer

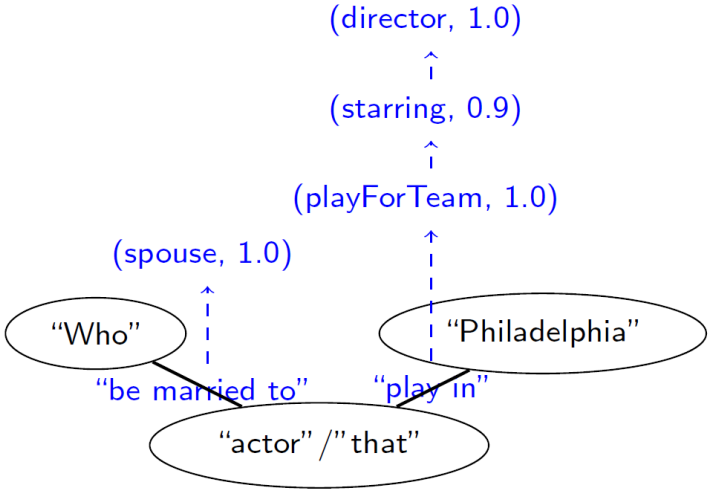
Online: Step 1.4



Our Approach- Data Driven Solution

gAnswer

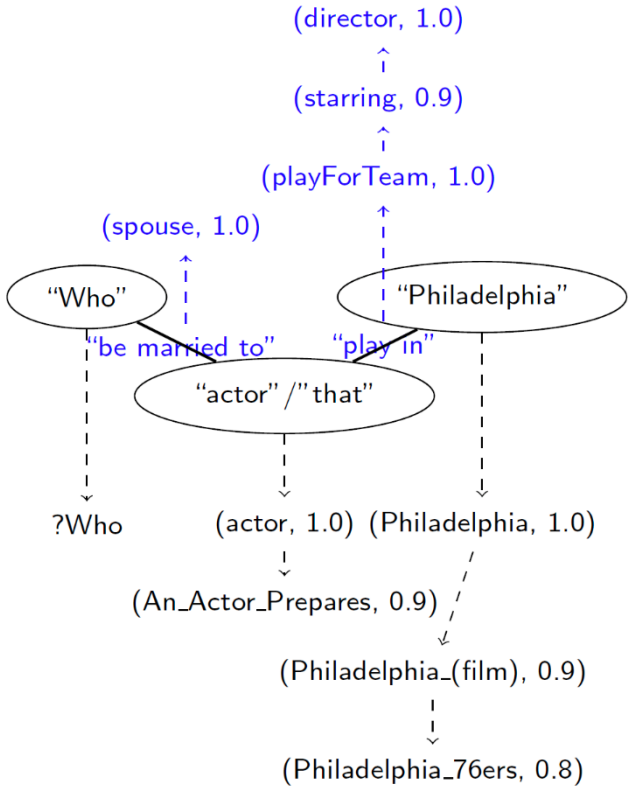
Online: Step 2.1 Mapping Edge



Our Approach- Data Driven Solution

gAnswer

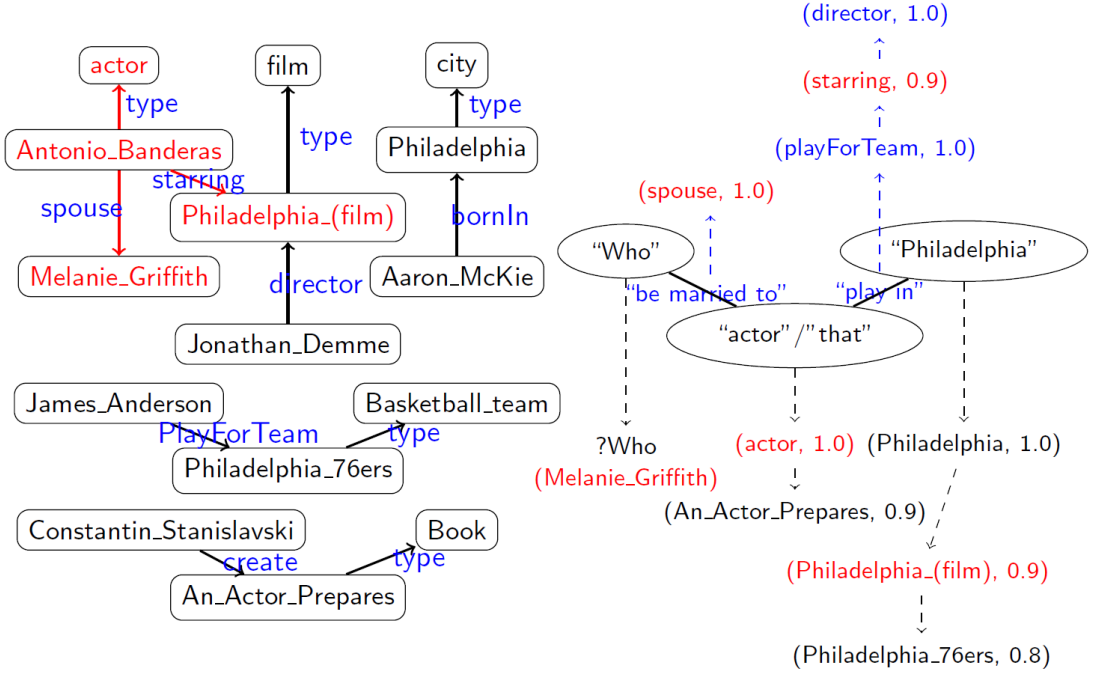
Online: Step 2.2 Mapping Vertices



Our Approach- Data Driven Solution

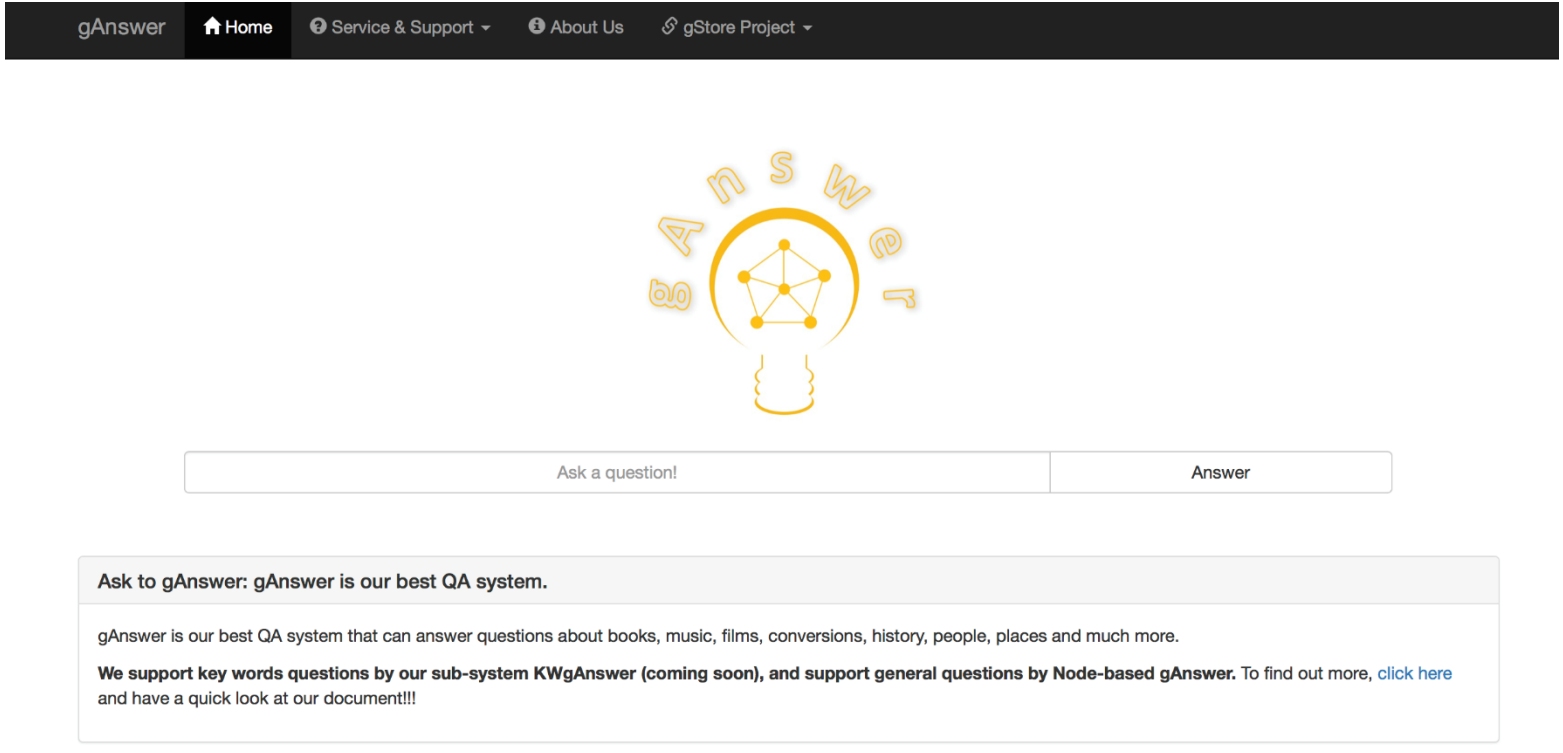
gAnswer

Online: Step 2.3 Finding Top-k Matches



Online Demo

URL: http://ganswer.gstore-pku.com/



Keyword Search Over RDF graphs ---a query graph assembly approach

Motivation

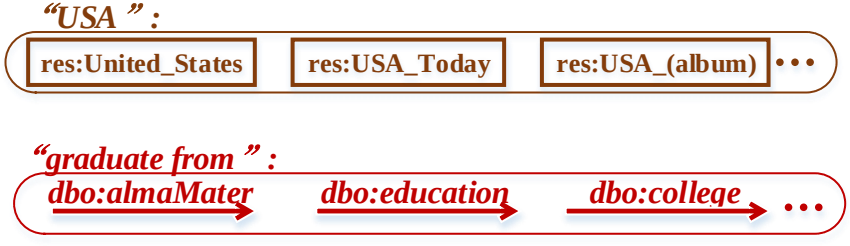
SPARQL vs Keywords

- Easy-to-use RDF query interfaces:
  - Natural Language Query Answering (NL-QA)
    - -- *"Which scientist graduate from a university that located in USA?"*
  - **Keyword Search**
    - -- *"scientist graduate from university USA"*
    - **more concise and flexible**

Challenges

Effectiveness

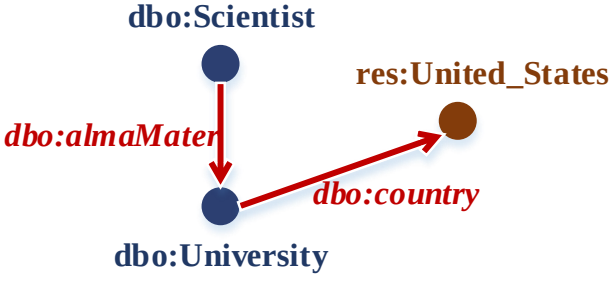
- Understanding the query intention **accurately**
  - ambiguity of keywords – multiple ways to “interpret” a keyword

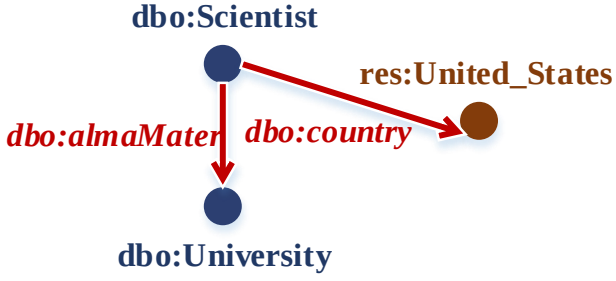


Challenges

Effectiveness

- Understanding the query intention **accurately**
  - ambiguity of keywords – multiple ways to “interpret” a keyword
  - ambiguity of query structures – multiple ways to “assemble” the query



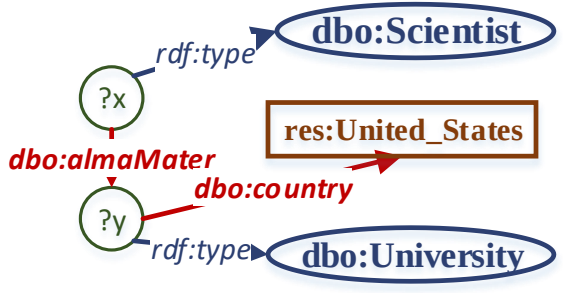


Our Task

- We study the keyword search on RDF graphs.
- Given a keyword token sequence $RQ = \{k_1, k_2, \ldots, k_m\}$, our task is to interpret RQ as a query graph Q.

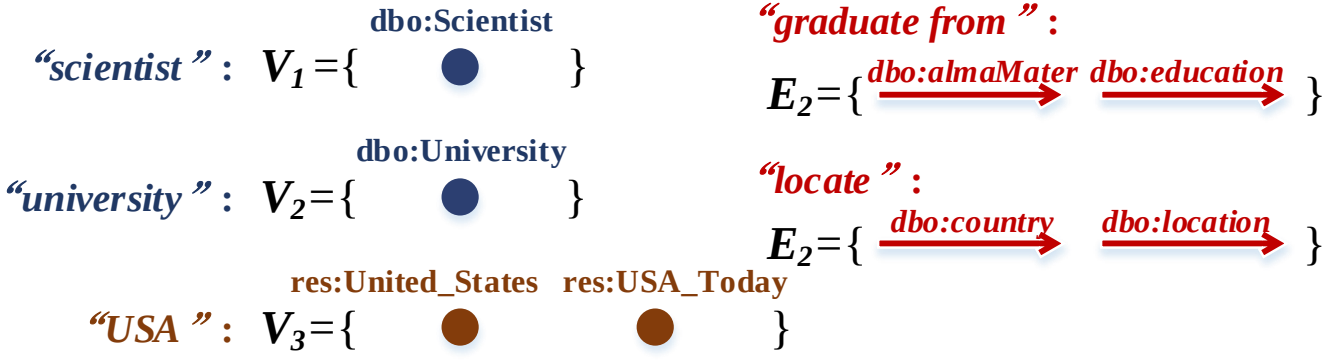
"scientist graduate from university USA"

$RQ \rightarrow Q$



Solution Overview

Query Graph Assembly



Elementary Query Graph Building Blocks

QGA Problem

Definition

- **Query Graph Assembly Problem (QGA):**
  - Given n vertex terms $t_i^v (i = 1, \dots, n)$, each t_i^v is matched to a set V_i of candidate entity/class vertices;
  - and m edge terms $t_j^e (j = 1, \dots, m)$, each t_j^e is matched to a set E_j of candidate predicate edges.
  - A valid *assembly query graph* $Q(V_Q, E_Q)$ must satisfy the following constraints:
    - each set V_i has exactly one vertex in V_Q;
    - each set E_j has exactly one edge in E_Q.

QGA Problem

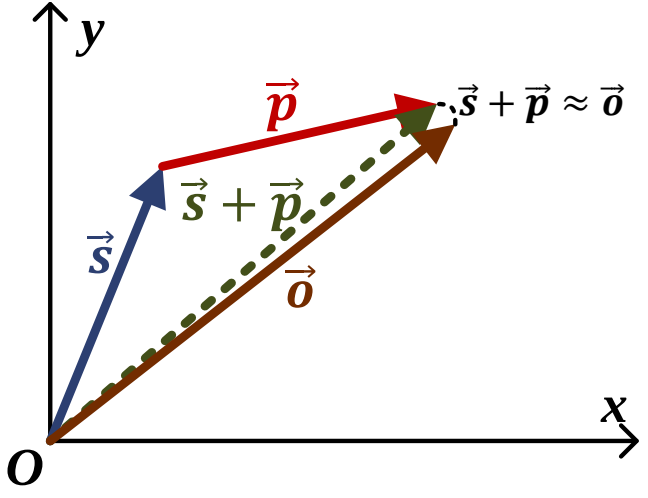
Cost Function

- $cost(Q) = \sum_{e(\langle v_1, v_2 \rangle, p) \in Q} w(\langle v_1, v_2 \rangle, p)$
  - where $w(\langle v_1, v_2 \rangle, p)$ denotes the triple assembly cost.
- The ***query graph assembly*** (QGA) problem is to construct a valid query graph Q with the minimum $cost(Q)$.

Assembly Cost

TransE Model

- $w(\langle v_1, v_2 \rangle, p) = MIN(|\overrightarrow{v_1} + \vec{p} - \overrightarrow{v_2}|, |\overrightarrow{v_2} + \vec{p} - \overrightarrow{v_1}|)$



QGA Problem

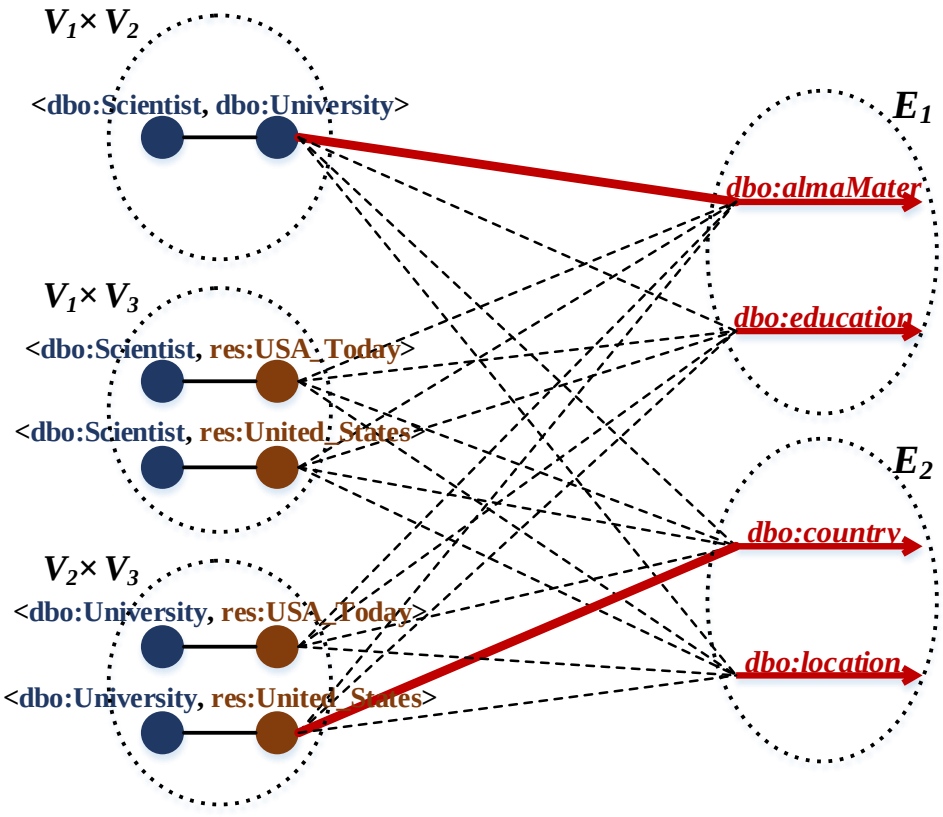
Hardness

- Theorem: The QGA problem is **NP-complete**.
  - *Proof: We reduce **3-SAT** problem to **QGA**.*

Bipartite Graph Model

Grouped Nodes

$V_i \times V_j$



E_i

Experiments

	DBpedia	Freebase
Number of Entities	5.4 million	41 million
Number of Triples	110 million	596 million
Number of Predicates	9708	19456
Size of RDF Graphs (in GB)	8.7	56.9

QALD is a series of evaluation campaigns on question answering over linked data.

		Processed	Recall	Precision	F-1	F-1 Global
CANaLI	(en)	100	**0.89**	**0.89**	**0.89**	**0.89**
NbFramework	(en)	63	0.85	0.87	0.86	0.54
UTQA	(en)	100	0.69	0.82	0.75	0.75
KWGAnswer	(en)	100	0.59	0.85	0.70	0.70
UTQA	(es)	100	0.62	0.76	0.68	0.68
UTQA	(fa)	100	0.61	0.70	0.65	0.65
UIQA (with manual)	(en)	44	0.63	0.54	0.58	0.25
UIQA (without manual)	(en)	36	0.53	0.43	0.48	0.17
SemGraphQA	(en)	100	0.25	0.70	0.37	0.37
PersianQA*	(fa)	100	0.19	0.91	0.31	0.31

QALD-6 Competition Results

IMPROVE-QA: An Interactive Mechanism for RDF Question/Answering Systems

Motivation

WHY? & WHY NOT?

Which actress was born in countries in Europe?

<Marilyn_Monroe> ✗ **WHY**

<Judy_Garland>

<Lana_Tumer>

<Audrey_Hepbum> ✓

<Mariene_Dietrich> ✓

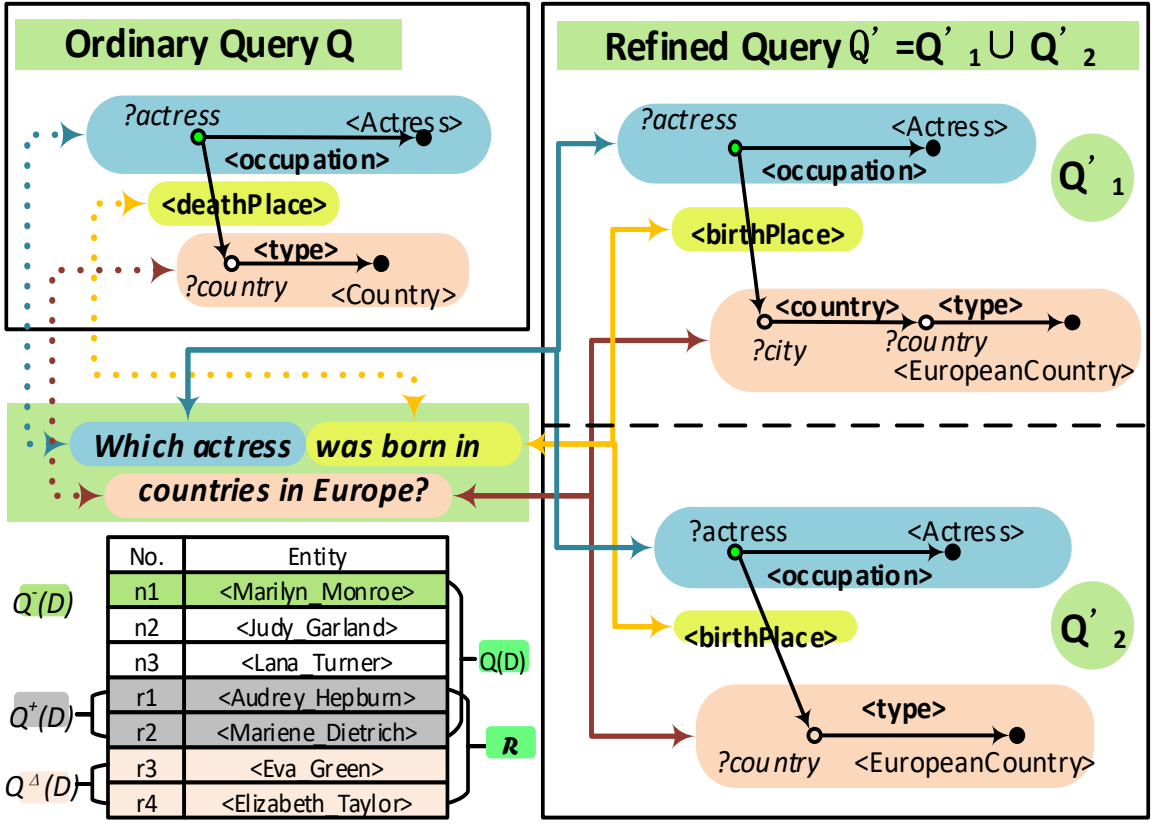
?

<Eva_Green> **WHY NOT**

<Elizabeth_Taylor> **WHY NOT**

IMPROVE-QA [Xinbo Zhang et. al , WWW 17 Poster &SIGMOD 18 demo]

Framework



Demo Group 2

Wednesday 14:00-15:30

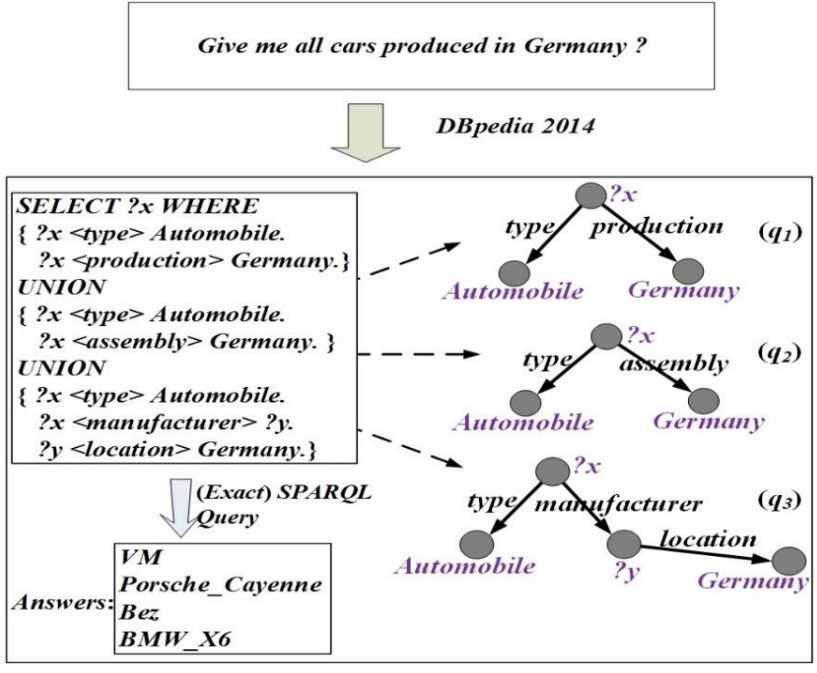
Semantic Search---a graph similarity-based method

Motivation

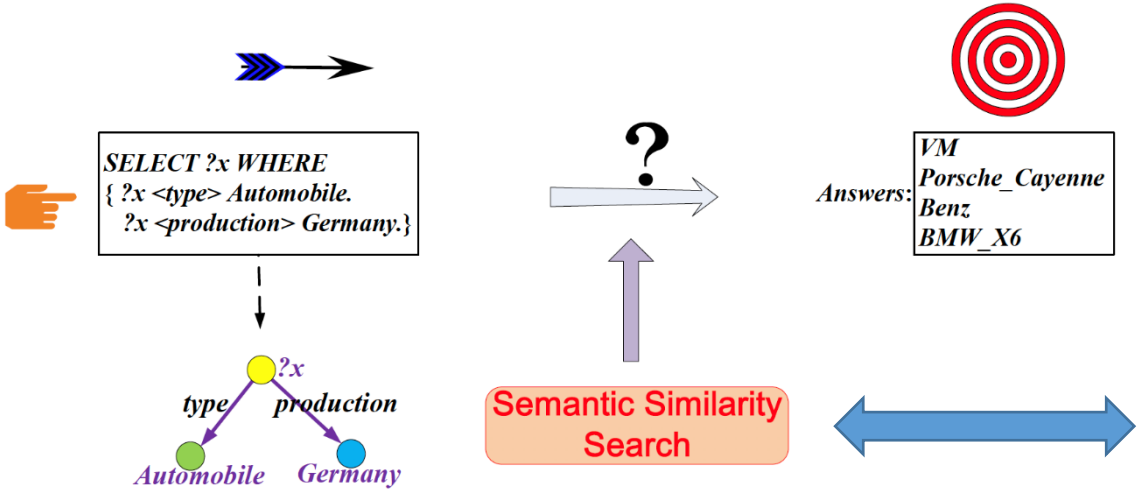
"Schema-less" leads to "Schema variety"

Eg: In DBpedia, "Germanic Vehicles" has at least FIVE different schemas

Enumerating all ?



Semantic Similarity Search [Weiguo Zheng et al., VLDB 2016]



Key Issue:

How to define "Graph Similarity Function" in the context of KG ?

Take-home Message

1. METHODOLOGY

Graph-based KG data management is a **feasible** strategy.

2. TECHNIQUE

We need to re-consider **graph computing techniques** in the context of KG.

Thanks

zoulei@pku.edu.cn