Lee- KG

BspG(— — —)

1. replace Bsp, "transform"
2. replace unions to disjoin
3. transform expressions from

union(...)

join(?, ...)

leftJoin(->)

Bsp: return true or false

[
sh:path wikibase:geoLatitude
sh:maxCount 1
sh:minCount 1
sh:datatype xsd:double

find inverse constraints

Subgraph query.

übung -

Allgemein. understanding

// what means **normalized?**:

<u>unify the forms</u>, units make data more friendly for auto... processing

↓

<u>converting</u> units to base units

> RDF dump (T)

> can be used in nlp efficiently.

"刘德华——用纯属合字吗?"

> <u>语义分析</u>.

> neo4j: graph database

<u>这些</u><u>关系</u>网络图:

> what shall be returned by quires?

↳ 实体或关系. object or relationships

> import wikidata to neo4j ! (+)

> neo4j py api (ext)

> pipeline :

NLP, NER. extraction.

↓

graph db query

↓

return result.

> neo4j schema ? (ext)

> 应用缓存 : redis / levelDB ——> Neo4j

> neo4j used with mongoDB

节点属性中存 mongoDB indexID.

> "关系理解"

> ER : entity, relationship

> KG: based on <u>notation</u>.
  DL: based on <u>math computation</u>

> RDF, Resource Description Framework.

> Serialization of RDF? ✓
  xml, n-Triples, Turtle....

> statement: <u>spo</u>

> rdf has expression limit. ✓
  → rdfs → owl
  ↑ (an extension)
  (with schema)
  ("<u>class</u>")

> 海量数据补全. ※ ✓

> wikidata → rdf: simplicity

> backend of wikipedia? php.

> wiki: 人类知识网络系统.

> nlp: 文本 mining. GFP.V.

> snippets also possible → py

↑ quick testing results.

S1: some methods for analysing networks and graphs. ✓

> graph theory ✓

> The Hype.

Def> facts, rules ← collection of

↳ knowledge

91 nodes. connections.

KG: a collection of knowledge.

KG is a dataset.

- structured
- normalized
- connected

- explicit
- connected
- non-hierarchical
- large

→ Wikidata, Yago2, Freebase, DBpedia, OSM, typical KG.

(q) → whatever typical.

→ Wikipedia, Relational DB

unstructured text

not normalized

focus on connections

q: why not KG?

h> community detection.

h.> Text mining.

p> what are **arcs**, **nodes**, ...

// [redacted] S.

q> what is KG. S.Op18.

pra> collection of facts, rules..
express relationships or connections

h> is a paradigm

q:> features of KG?
KG is a special dataset
S.Op19.

q> is wikiData KG? Yago2? OSM?

q> why not Facebook Social Graph?

q> WordNet Op28. not a KG.

q> Why are they not KG?:

① Wikipedia

② Relational DB:

pro>

no focus on connecting unstructured text, connections sub-ordinary.

q> Ref of a simple undirected Graph? (edges, nodes?) of it?

Ch 32.

pro> G(V, E) ← set of edges

↓

set of vertices (nodes)

undirected edge called arcs

Q> Def of Directed Graph.

t = ? Op 34

arrows from to ?

Ans from source vertex to target vertex arrows.

e,

$v_1 \rightarrow v_2$

Q:> graph with self-loop ?

multi-graph ?

vertex, edge - labeled graph. ?

in-degree ? Op 35-36

Ans

number of edges that pointing towards it.

in-degree.

Q> What is directed path?
simple path?
notation of a d. path?

up37

proof>

$v_0 \xrightarrow{e_1} v_1 \xrightarrow{e_2} \cdots \xrightarrow{e_n} v_n$

Sequence of consecutive edges.

Simple path: without repeat of vertices other than first and last one.

Q> connected graph?
Strongly connected graph?

up38.

Q> adjacency matrix?

pre> $G = \langle V, E \rangle$, $|V| \times |V|$ matrix

h> Adjacency matrix for undirected graphs are symm.

h> matrix ⟶ multi-graphs
labelled-graphs.
possible.

Q:> how? <u>op 39</u>

h. Q:> possible representation of a Graph?

- adj matrix
- adj list
- combine / other options

h> allowing repeated lines
↳ multi-graph.

h> the list does not list V.
↳ can be modified acc.

Q> pros and cons in the repres. of list and matrix. which always been used in KG, and why? (op 41)

pro> matrix → good for dense graphs.

h> why RDF, we are going to exchange graphs bet. apps.

Q> what is RDF stands for.
op 67

h> originally built for <u>web</u> <u>data exchange</u>

h: 图数据的交换，每不违可以处理了似的.

h> focus on RDF 1.1, <u>open standard</u>.

Q> What does RDF support?
multi-graphs (restricted)
What does it mean? <u>ep 51</u>

Q> URI, IRI, URL? <u>ep 54</u>

pro) <u>https://example.org/S/P</u>?-
path. #fragment

scheme — authority

Q> Form of URL? <u>ep 55</u>

Q> What is urn used for?
http / ftp ... / irc / file
meaning? <u>ep 56</u>

RDF uses <u>IRIs</u>?

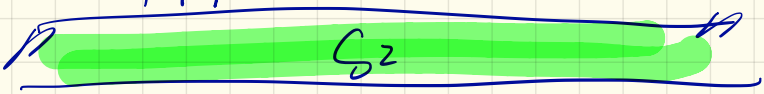
h> many IRIs may refer to the same thing.

h> RDF features.

modelling in RDF.

RDF 1.1 spec.

RDFs. → extented RDF



Q> what kind of graph is standard in RDF graph?

S2bpd.

pra> founder

named-after.

directed
edge-labeled.

Q> how comes IRIs in RDF?

X2. Opos

Q> why use domain information in IRIs? Opos

Q> avoid existing web pages?

h> maybe IRIs return useful info content via https. ?
why doing this

h> Guidlines for creating rdf.
Struc.
why create rdf? reasoning for purpose., do sth meaningful within dataset.

Q> IRIs should display to users? X

Q> IRIs should return some useful info how ? Op/o.

X3.

prac) { fragments
Http redirects.
Content negotiation

Op 11.

Q.> detail about Content nego.: how to get a turtle instead of JSON file ?
↳ use "Accept header"

Q.> why not use IRIs to represent data values too?

x3 ~~op 16~~

pros

1. same everywhere.
2. builtin understanding
3. more "interpreted" — "not extensive" than IRIs.

"更好解释的" do not needed, otherwise may be more unnecessary "interpretations"

h.> "lexical value" ^^ datatype-IRI

pra> "547" ^^ xsd:int, "1873-01-31" ^^ xsd:date

Q.> what does "xsd" mean here?

datatype based on XML schema.

Q.> components of a Datatype in RDF:

x3 op 19.

Q.> What is the lexical space of integer under W3C standard XML schema?

finite-length strings of decimal digits (0-9) and a leading sign.

h> many standard data types in RDF: XML Schema.

RDF 标准中支持 XML Schema 数据类型这是一种良好实用.

Q> value space and Lexical space of decimal (xsd:decimal)

value space: Arbitrary-precision decimal numbers.

Lexical space: decimal strings

Q:> how values in Lexical space can be transferred to value space? 对应

use value mapping

Q> xsd:string. the mapping for it is identical. ?

pre> typical datatypes:

int, integer, double, float, boolean, ----

h.Q: what is the diff bet. int and integer in xsd ↑ bounded?

h> xsd:date, xsd:dateTime portion, time, year.

h> fun: see: rdf:, datatype section.

h> two literals can have the same value without being the same rdf term:

"1" ^^ xs:integer

"01" ^^ xs:integer

h> Literals used for values.

Q> what means 'ill-typed'?
op23

Q> how to encode strings? relationships?
↳ IRIs

Q> how about encoding values? (web-rdf11)
↳ literals

hQ:> rdf terms, x3?
IRIs, literals, b/lank nodes

h> An RDF graph is a set of rdf triples? ✓
rdf triples can be expressed with terms? ✓

Q> "string"@lan.
what is this part?
opdx (BCP47)

"chips"@en-UK

5> language-tagged string is widely used in practice to encode human-readable labels.

h> b-nodes: as placeholder
used as auxiliary vertices
↳ today, largely avoided.

Q> formal def of a rdf graph op28

h> Subject: IRI, b-node
predicate: (IRI) must!
Object: IRI, b-node, literal
↳ Optional!
3 possible vals

h> ill-formed literals are allowed when defining a graph.

> RDF serialisations.

↳ Abstract syntax of RDF?

Q> Serialization formats? possible? x5 op32

↳> and some further historical/unofficial formats exists!

↳> meaning 3 types,
- N-triples
- Turtle
- JSON-LD

↳> pointy brackets: <>

↳> bnodes has no global meaning? ✓ — : Skolemized. Only within this doc.

tbd > see examples later. some open datasets exists: dblp

↳> n-triples has official support and official website? ✓

Q> encode a given graph. op34

Q> pros and cons for n-triple representation op35

pro> simple / fast / basic. x2.

cons: { storage space
reading writing unfriendly.

Q> How can turtle shorten IRIs? x3+ op36

Q> What is the propose of BASE keyword? op36.

A> xsd. prefix, rdfs prefix

pro> BASE, PREFIX, semicolon(;) and (,)

Q> What does this bracket mean?
subject [...] op39.

pra) predicate - object pairs inside bnode, which is assumed as subject.

h> numbers can also be written directly, without quotes, but only to defaunt types. (may slightly after from ori:)

h.Q> pro cons of turtle expression

~~op40~~

h> RDF/XML, 在RDF表示上, 人们也走过弯路. XML: <u>hard to parse, to read</u>

Q> why use JSON format?

hQ> compare format: N-triple
turtle
JSON-LD.

(op42)

Q> what does those commonly used prefix mean?
xsd, rdf, rdfs op43.

Q> what are qualified names? op63.
↗
xsd:dateTime.

Q> what is RDF dataset op64

Q> How to enable named graph into RDF dataset?
op65

Q> What is N-Quads / TriG ?

Q> what is rdf properties? the use of rdf:type?
op68.

Q> How to tag a predicate? display names of predicates? define their label?

pra> Property-IRI rdf:type rdf:Property.
use this triple to declear
Resources represented by resources. called property.

Q:> RDF Graphs are hypergraphs ?
with ternary edges? unary ternary
what does a real rdf looks like?
↑ (tagged, labelled)

op52

h> Property(R) may have more details, one can define more informations about it.

A> In the simplest case, declare a property and assign it a label then use it as a predicate.
label / comment / range / domain
op53

h> RDF can support properties without declarations (directly use xxx as predicate)
but not recommand.
Acc as robustness of RDF, which is useful when merging diff. datasets.

R> How this robustness be used? op55.

Q> several syntactic formats exists for exchanging RDF data.
which are they?

S3

Q> What kind of information can be expressed in RDF? Other than Graphs op05

A> of course not all kinds of data. can expressed with RDF format.

pro> CSV files, xml docs, other kind of graphs

A> almost all kinds of graphs can be encoded into RDF, and exchange with libs come with the standard: we do not have to rewrite, imp. the code again, simplify the development.

Q> Challenges when converting files (databases) into RDF DB? x2 op09

pro> higher cavity. + express ordered list

express. (only triples not)

con> RDF2RDF exists: if new features in RDF are needed, we have to convert our old knowledge base / database into the new one.

→ RDF can express n-ary relations. (but) with some effort.

Q> encode this relational table to RDF Graph. op10

A> Sometimes bnodes are needed to do transformation.

Q> What is RDF Reification? op14.

↳ pro> The introduce of bnodes when encoding.

h> In RDB2RDF, table names will be taken into account. ?V

Q> What is R2RML? → mapping Lan.

Q> How to express order information in RDF?
x3 opts. and their pros and cons.

pro> linked lists, properties, sets

Q> What does SPARQL stands for? op23

Q> One can both query and update a DB through SPARQL?

S-Q> Query all res IRIs alone with their labels? op24

h> The core part of QL is query condition WHERE{...}

Q> What is query condition? What syntax supported? variables?

pro>
```
PREFIX rdfs: <http.... >
SELECT ?resources ?label
WHERE {
    ?res  rdfs:label  ?label .
}
```
→ must have, similar to tuples.

h: all triples that has this predicate will be returned that is pattern matching process.

h: rdfs must defined in the database.

S-Q> find up ten people whose daughter is professor op25

A>
```
PREFIX eg: <http://  />
SELECT ?parent
WHERE {
        ?parent  eg:hasDaughter  ?child.
        ?child   eg:occupation   eg:Professor.
}
```

h> { eg: hasDaughter
eg: professor — should be provided.
eg: occupation.

LIMIT 10.   h: varibles should have a meaningful name.

h: make the match explicit.

Q: How to limit the result?   op26

Q: How to count?

S-Q: Count all triples? all predicates?   op27

```
SELECT (COUNT(*) AS ?count)
WHERE {
    ?subject ?predicate ?object .
}
```

→ (?count) 只当后面condition没有出现。(" as placeholder")

→ (?subject ?predicate ?object) 完全是任意匹配，其实等于没说。

S-Q (my) > Count all subjects?

S-Q > Find person with most friends?   op29
Find pairs of siblings?
take use of these: eg: hasFriend
eg: hasChild.

Q > How to express "only two xxx" (child, parent...)   op29.

S-Q > "most" means?   op30

└→ { GROUP BY — .
ORDER BY DESC(—)
LIMIT 1

S-Q > How to ensure that children are diff. from each other?
└ op30

Q> What is solution set modifiers   op31

Q> What is select clause where clause.

p-Q> Most imp. block of a SPARQL query? op32. x4

p-Q> possible blocks instead of SELECT block? x3 op32.

ASK / CONSTRUCT / DESCRIBE

h> RDF terms within query? similar.

Q> The one has most friends.   ~~xxx~~ 8. as output.

```
SELECT   ?person. (COUNT(?f) As ?friendCount)
WHERE {
    ?person  eg:hasFriend  ?friend
} GROUP BY  ?person
  ORDERED BY  DESC(?friendCount)
  LIMIT 1
```

在select中可以定义变量.

Q> What is a variable in SPQL.?
two possibilities for that? op35.

Q> What is Basic Graph Pattern? op36.

Q> SPARQL and RDF has diff support on datatypes? op36
BGP is a set of triple patterns.

h> triple patterns describes query conditions
and we are using them to match triples in DB
find matches that fit all triples at once.

h> extension of turtle.

Q> Bnodes are not part of graph? ✓ op38

Q> meaning of bnodes in query patterns? op38/39 x1.

① hg :> (h2) high lev understanding with graph. ← not so imp.

② pra> practice . imm: draw.

h>

Q> type only

mixed . approach . on pc .

⋮

Generat fib copy paste fast understanding

pc.

mixed - write

mixed - type

Q> What means that when one bracket occurs in multiple lines of the result ? op41.

Q> might be diff. from that used on Data. ? ✓