Knowledge Graphs

G. Falquet

Source:

Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutierrez, C., Gayo, J. E. L., Kirrane, S., Neumaier, S., Polleres, A., Navigli, R., Ngomo, A.-C. N., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., & Zimmermann, A. (2022). Knowledge Graphs. *ACM Computing Surveys*, *54*(4), 1–37. https://doi.org/10.1145/3447772

Contents

Hogan, A. et al. (2022). Knowledge Graphs. *ACM Computing Surveys*, *54*(4), 1–37. https://doi.org/10.1145/3447772

- Definitions
- Data models and query languages
- Representations of schema, identity, and context
- Quality dimensions by which a knowledge graph can be assessed
- Deduction in KG
- Induction in KG
  - Graph analytics
  - Graph embeddings
  - Graph neural networks
  - Symbolic learning: rule and axiom mining

Definitions

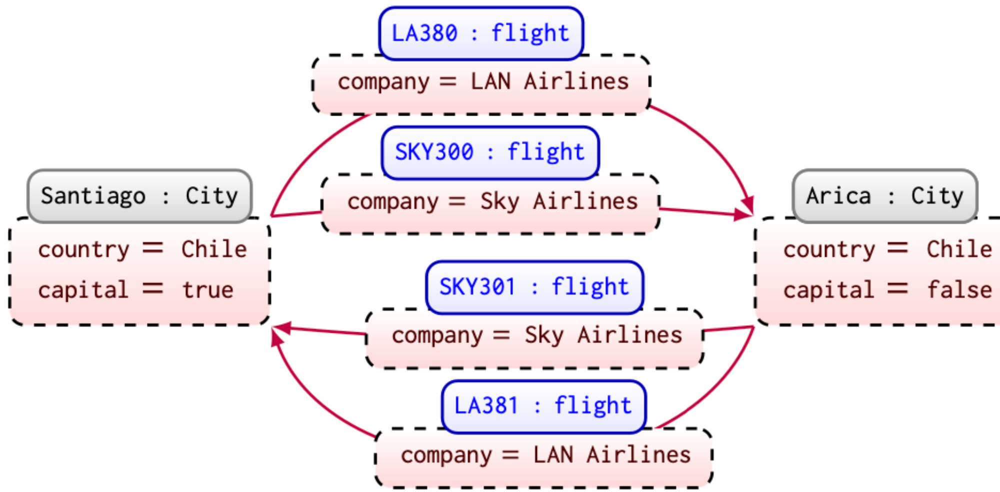
A knowledge graph as a graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent relations between these entities.

- Knowledge may be composed of simple statements,
  - "Santiago is the capital of Chile",
- or quantified statements
  - "all capitals are cities".
- Simple statements can be accumulated as edges in the data graph.
- Quantified statements require a more expressive way to represent knowledge – such as ontologies or rules

Data models and query languages

- Directed edge-labelled graphs
  - e.g. RDF
- Graph datasets
  - e..g RDF datasets

- **Property graphs**
  - property-value pairs
    - on nodes
    - on edges
  - typed nodes and edges



Hogan et al. (2022)

- Translation without loss of information DELG ⟷ PG

Schema

Semantic schema

- e.g. RDF Schema

Table 1. Definitions for sub-class, sub-property, domain and range features in semantic schemata

Feature	Definition	Condition	Example
Subclass	$c \xrightarrow{\text{subc. of}} d$	$x \xrightarrow{\text{type}} c$ implies $x \xrightarrow{\text{type}} d$	City $\xrightarrow{\text{subc. of}}$ Place
Subproperty	$p \xrightarrow{\text{subp. of}} q$	$x \xrightarrow{p} y$ implies $x \xrightarrow{q} y$	venue $\xrightarrow{\text{subp. of}}$ location
Domain	$p \xrightarrow{\text{domain}} c$	$x \xrightarrow{p} y$ implies $x \xrightarrow{\text{type}} c$	venue $\xrightarrow{\text{domain}}$ Event
Range	$p \xrightarrow{\text{range}} c$	$x \xrightarrow{p} y$ implies $y \xrightarrow{\text{type}} c$	venue $\xrightarrow{\text{range}}$ Venue

Hogan et al. (2022)

Schema

Validating schema

- to represent diverse, incomplete data at large-scale → OWA
- in some scenarios → guarantee that (part of) the data graph is "complete".

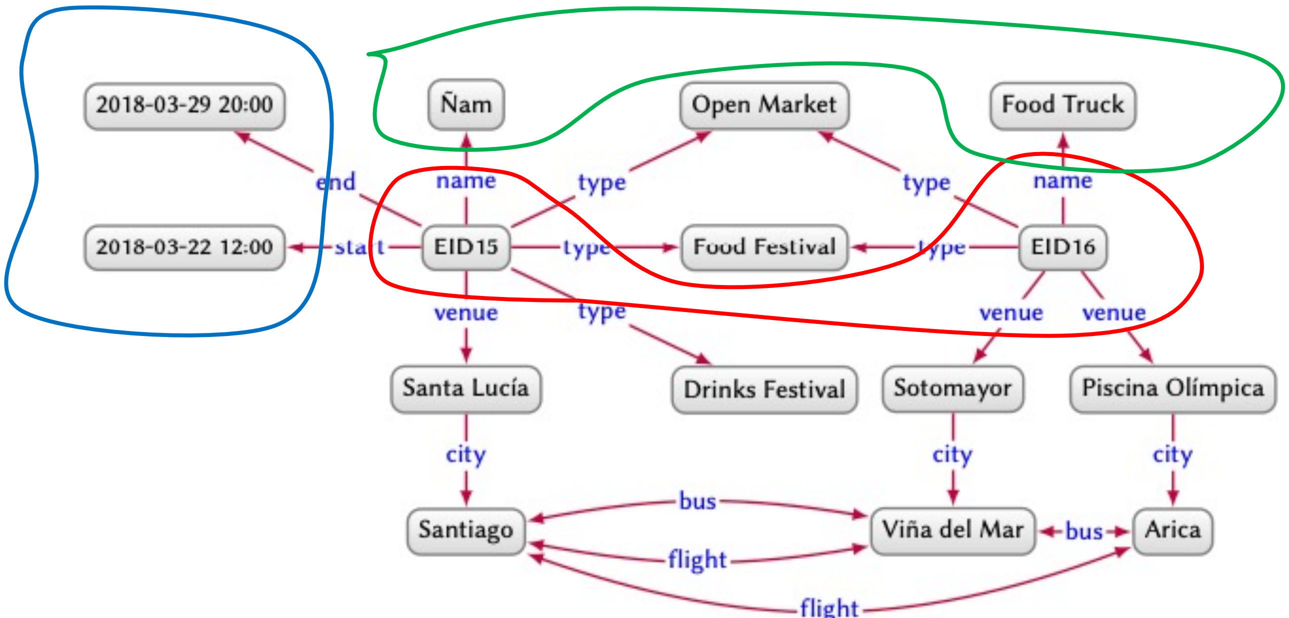
- UML class diagram
- SHACL shapes

Schema

Emergent schema

Quotient graph

- partition node set into equivalence classes
  - based on their context
- replace node x by its class $[x]$, keep the edges
  - simulation $(s\ p\ o) \;\Rightarrow\; [s]\ p\ [o]$
  - bisimulation $(s\ p\ o) \;\Rightarrow\; [s]\ p\ [o]$ iff $\forall x \in [s] \exists z \in [o]: (x\ p\ z)$



Hogan et al. (2022)

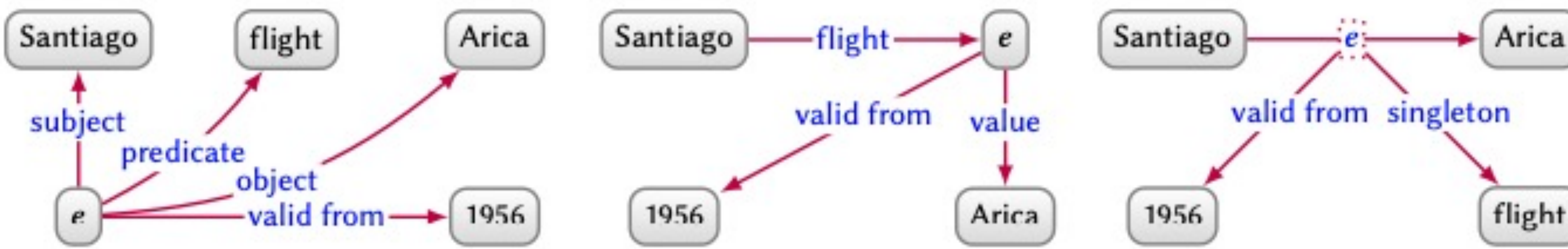
Context

- Facts considered true with respect to a context (scope of truth)
  - temporal
  - geographic
  - provenance

  Often left implicit, e.g. temporal context = now

- Representation
  - direct (with TIME, PROV, ... ontologies)
  - reification
  - higher arity
  - annotation

Reification techniques

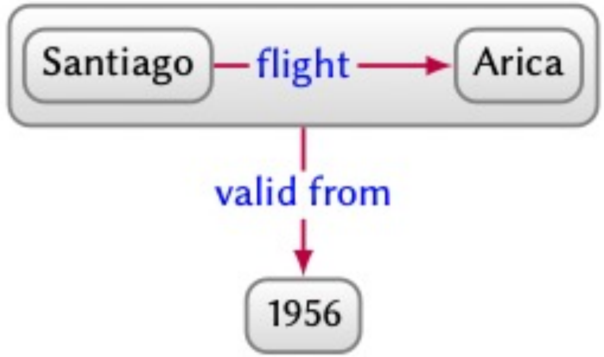


(a) RDF Reification (b) *n*-ary Relations (c) Singleton properties

Hogan et al. (2022)

Higher-arity: RDF*

<<:Santiago :flight :Arica>> :valid_from 1956



Remark

<<:Taylor :spouse :Burton>> :from 1968 ; to 1978 .

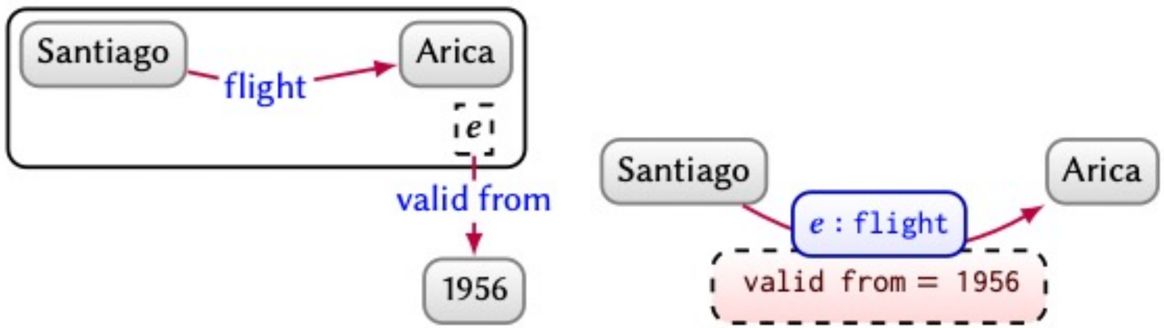
<<:Taylor :spouse :Burton>> :from 1981 ; to 1983 .

→

<<:Taylor :spouse :Burton>> :from 1968 ; from 1981 ; to 1978 ; to 1983 .

.

Higher-arity

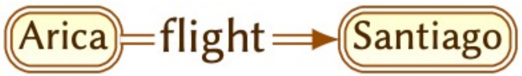


(a) Named graph (b) Property graph

Hogan et al. (2022)

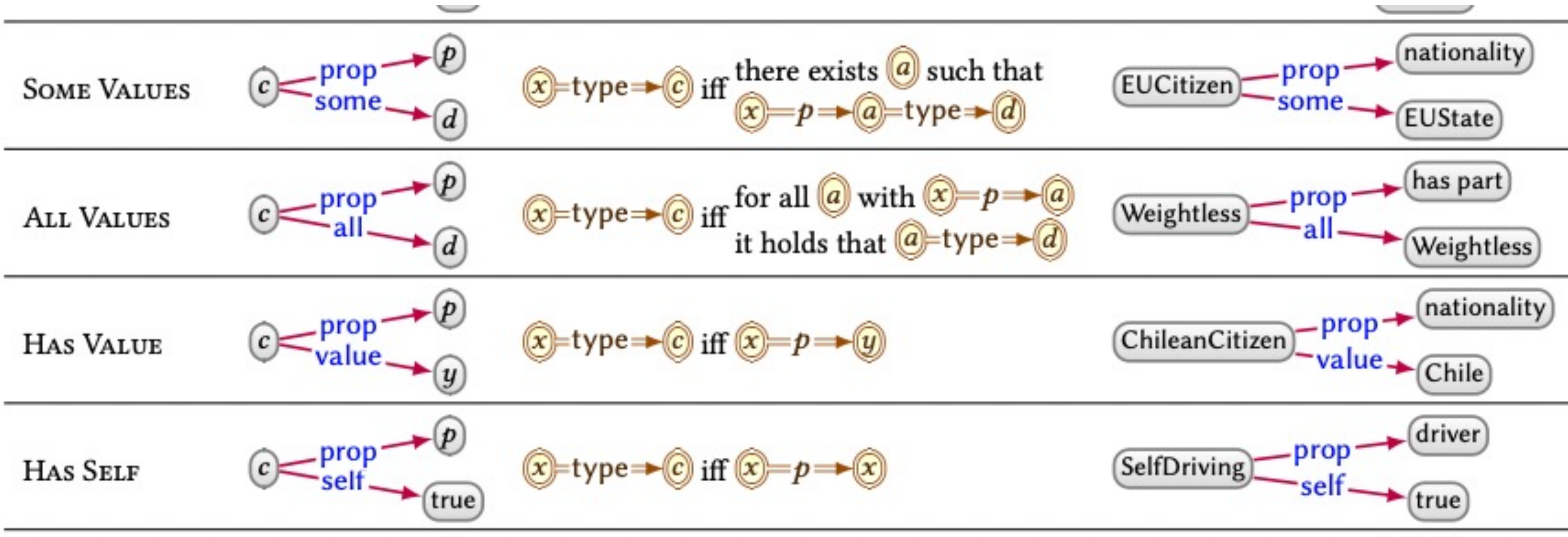
DEDUCTIVE KNOWLEDGE

- Ontologies
  - Interpretation
    - Data graph (nodes, edges) $\rightarrow$ Domain graph (entities, relations)



Feature	Axiom	Condition	Example
Assertion	$x \xrightarrow{y} z$	$x \xrightarrow{y} z$	Chile $\xrightarrow{\text{capital}}$ Santiago
Negation	$n \xrightarrow{\text{type}}$ Neg, $n \xrightarrow{\text{sub}} x$, $n \xrightarrow{\text{pre}} y$, $n \xrightarrow{\text{obj}} z$	not $x \xrightarrow{y} z$	$n \xrightarrow{\text{type}}$ Neg, $n \xrightarrow{\text{sub}}$ Chile, $n \xrightarrow{\text{pre}}$ capital, $n \xrightarrow{\text{obj}}$ Arica
Same As	$x_1 \xrightarrow{\text{same as}} x_2$	$x_1 = x_2$	Región V $\xrightarrow{\text{same as}}$ Región de Valparaíso
Different From	$x_1 \xrightarrow{\text{diff. from}} x_2$	$x_1 \neq x_2$	Valparaíso $\xrightarrow{\text{diff. from}}$ Región de Valparaíso

Hogan et al. (2022)



Hogan et al. (2022)

INDUCTIVE KNOWLEDGE

- Graph Analytics
- Knowledge Graph Embeddings
- Graph Neural Networks
- Symbolic Learning

Graph Analytics

- Discovering interesting patterns

- Techniques
  - Centrality computation
    - PageRank, ...
  - Community detection
  - Connectivity
  - Node similarity

Knowledge Graph Emdeddings

- Predicting new edges
- Identifying erroneous edges

- Machine learning techniques $\rightarrow$ Numeric input as vectors

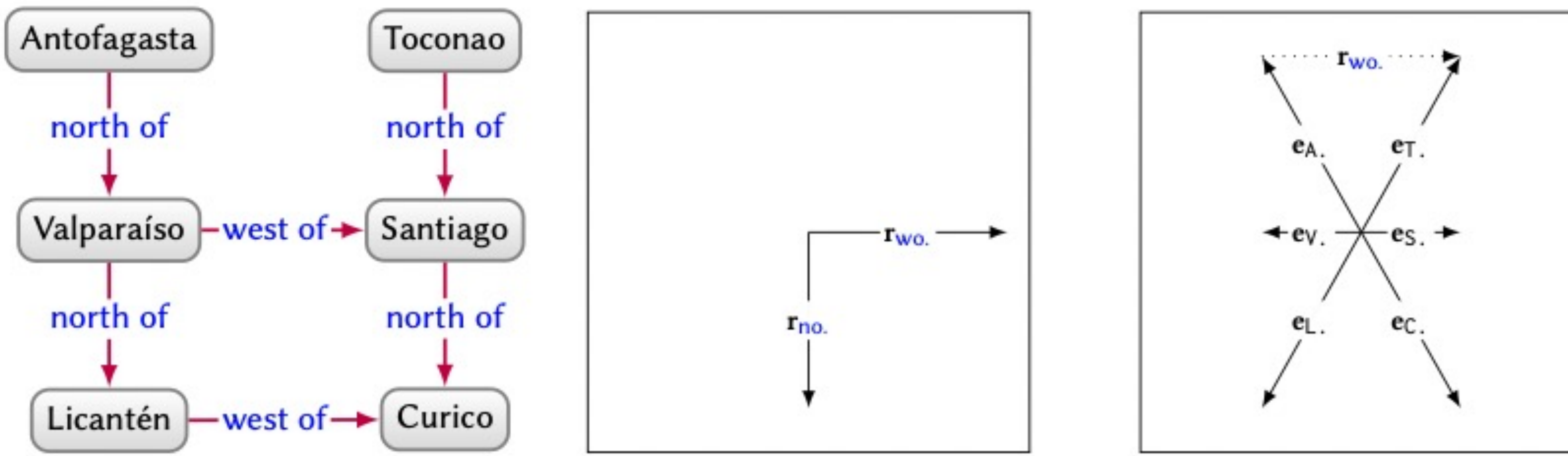
  - How to encode graphs as numeric vectors?

- Graph embedding
  - entity embedding: node $\rightarrow$ d-dimensional vector
  - relation embedding: edge $\rightarrow$ d-dimensional vector

- (s p o) $\rightarrow$ (es rp eo)
  - define a plausibility function for the edge
  - goal: find embeddings that
    - maximize the plausibility of positive edges (in the graph)
    - minimize the plausibility of negative edges (not in the graph)
- Tasks
  - assign a confidence level to edges
  - complete edges with missing labels
  - a basis for similarity measures
    - duplicate detection
    - recommendation

Translational model

- TransE (edges as transformers)
  - from (s p o) learn es, rp, eo
  - goal:
    - on positive examples: es + rp close to eo
    - on negative example: es + rp far from eo



(a) Original graph (b) Relation embeddings (c) Entity embeddings

Hogan et al. (2022)

- Limitations
  - transforms everything
  - (s p o1) (s p o2) $\rightarrow$ tend to define eo1 = eo2
  - cyclical relations $\rightarrow$ 0

- Improvements
  - separate hyperplanes for different relations, ...

Language models for embeddings

- Leverage proven approaches for language embeddings

$$\text{word} \rightarrow \text{vector}$$

- RDF2Vec
  - build "sentences" by performing random walks in the graph
  - input to word2vec

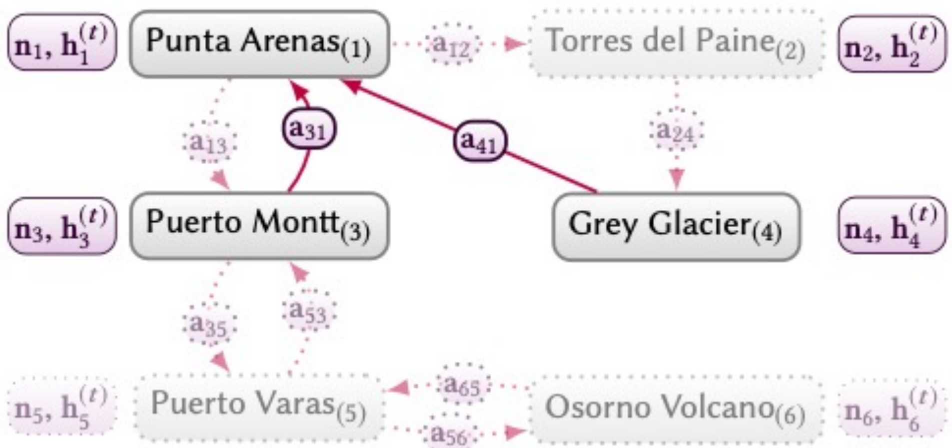
Graph Neural Networks

- Classical NN: homogeneous topology (layers)
- GNN: topology of the data graph

- node $\rightarrow$ feature vector (fixed)
- node $\rightarrow$ state vector
  - parametric transition function, input = neighbour nodes information
  - output function
- execution until a fixpoint is reached

- the function are implemented using neural networks
  - learn the parameters to best approximate the results for the supervised nodes

example



$$\mathbf{h}_x^{(t)} := \textstyle\sum_{y \in \mathrm{N}(x)} f_{\mathbf{w}}(\mathbf{n}_x, \mathbf{n}_y, \mathbf{a}_{yx}, \mathbf{h}_y^{(t-1)})$$
$$\mathbf{o}_x^{(t)} := g_{\mathbf{w}'}(\mathbf{h}_x^{(t)}, \mathbf{n}_x)$$

$$\mathbf{h}_1^{(t)} := f_{\mathbf{w}}(\mathbf{n}_1, \mathbf{n}_3, \mathbf{a}_{31}, \mathbf{h}_3^{(t-1)}) + f_{\mathbf{w}}(\mathbf{n}_1, \mathbf{n}_4, \mathbf{a}_{41}, \mathbf{h}_4^{(t-1)})$$
$$\mathbf{o}_1^{(t)} := g_{\mathbf{w}'}(\mathbf{h}_1^{(t)}, \mathbf{n}_1)$$

$$\ldots$$

Hogan et al. (2022)

Symbolic Learning

- Learn rules or axioms
- Based on standard data mining techniques
  - support
  - confidence

OTHER TOPICS

1. Creation and enrichment of knowledge graphs from external sources.
2. Quality dimensions by which a knowledge graph can be assessed.
3. Techniques for knowledge graph refinement.
4. Principles and protocols for publishing knowledge graphs.