Representation Learning for Knowledge Graphs: Link Prediction and Entity Alignment

Wei Hu and Zequn Sun

Nanjing University

whu@nju.edu.cn, zqsun.nju@gmail.com

3rd International Workshop on EntitY Retrieval and lEarning (EYRE '20)

Agenda

- Part I: Introduction (10 minutes) ← Wei Hu
- Part II: Link prediction (10 minutes) ← Zequn Sun
- Part III: Entity alignment (15 minutes) ← Zequn Sun
- Part IV: Future directions (5 minutes) ← Wei Hu
- Part V: Q&A (5 minutes) ← Wei Hu & Zequn Sun

Slides: https://github.com/nju-websoft/RrepresentationLearning4KGs

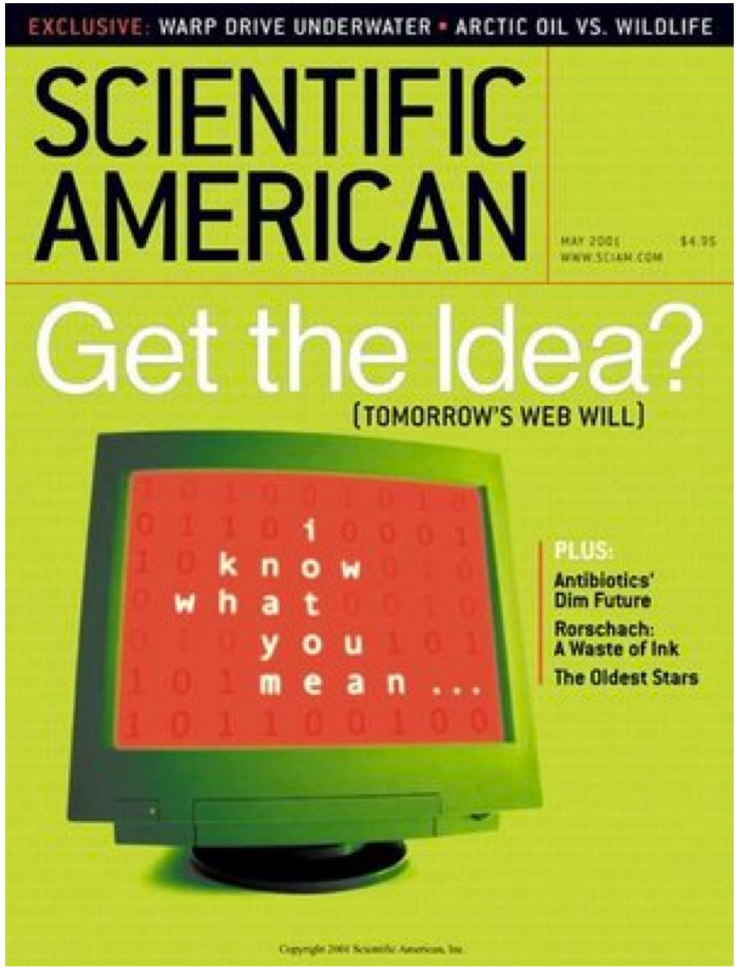
Part I: Introduction

Wei Hu

Semantic Web

- **Semantic Web**
  - Sir Tim Berners-Lee, May 2001
  - Give formal meanings to web information → semantics
    - Web 1.0 → Web 2.0 → Web 3.0 (a web of data)

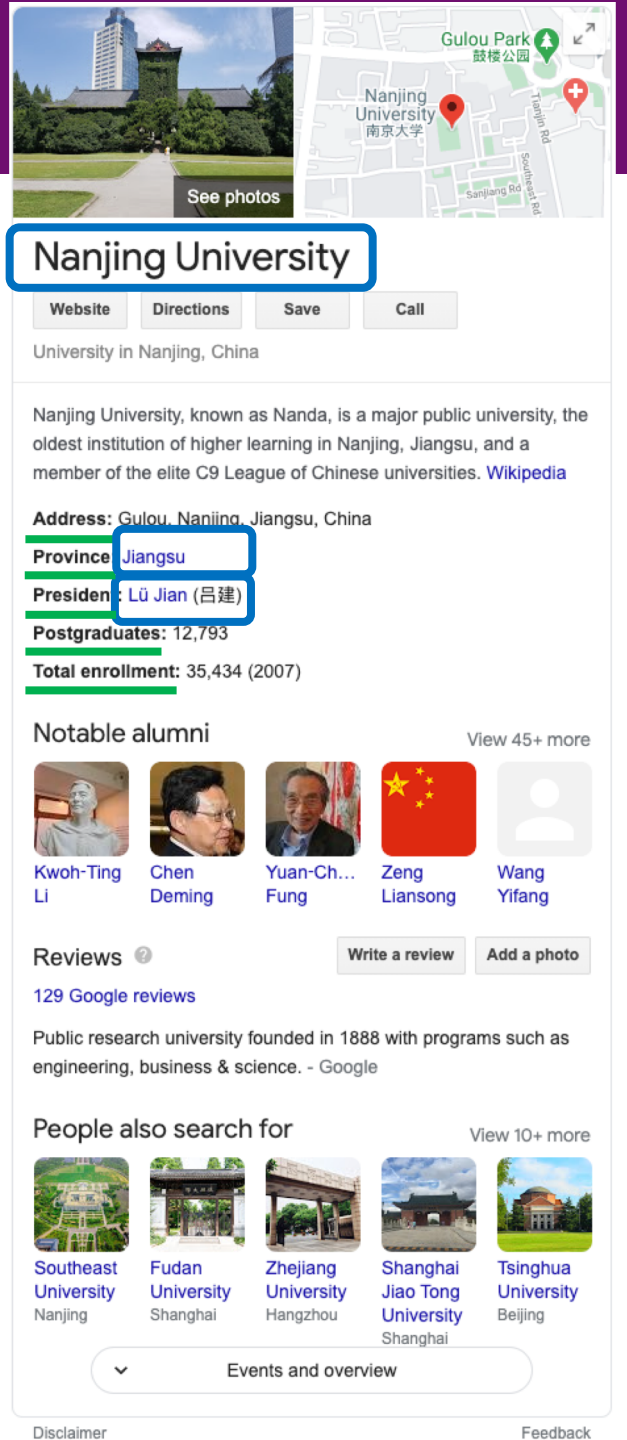
- Semantic Web is about
  - Common formats for
    - Integration and combination of data drawn from diverse sources
  - Language for
    - Recording how the data relates to real-world objects



Knowledge Graphs (KGs)

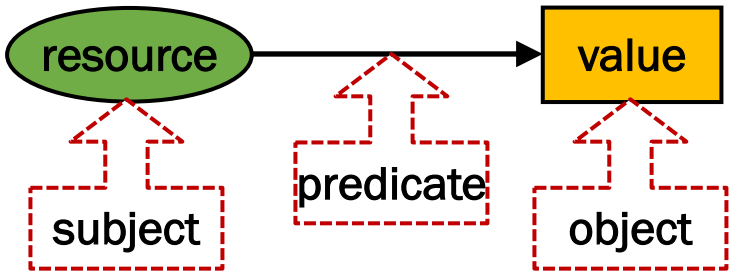
- **Knowledge Graph** (KG) is a knowledge base used by Google to enhance its search engine's search results with semantic-search information gathered from a wide variety of sources
  - Google, May 2012
  - The world is not made of strings, but is made of **things**

  - **Nodes:** entities and concepts
  - **Edges:** attributes and relations

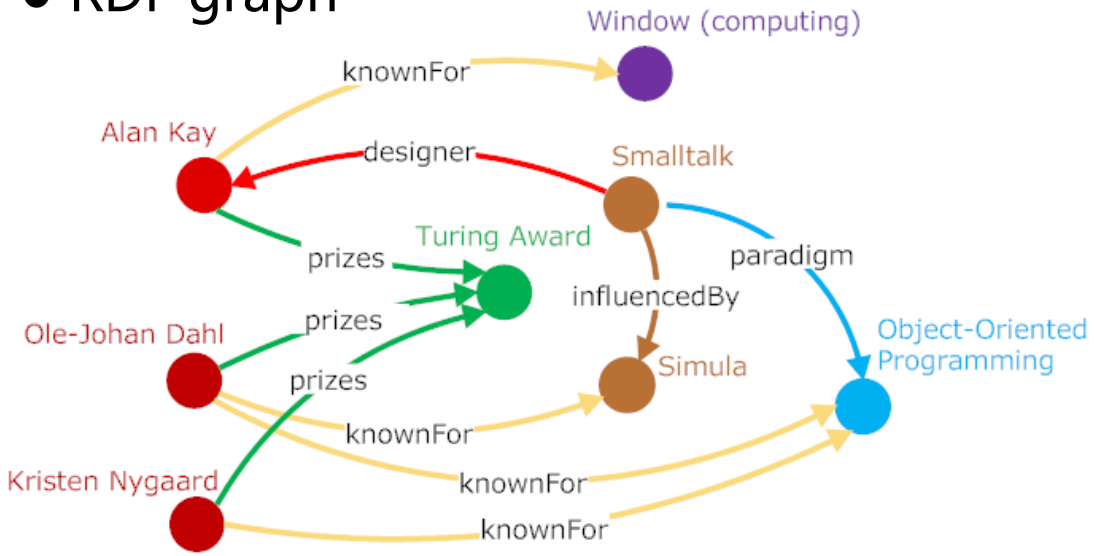


Resource Description Framework (RDF)

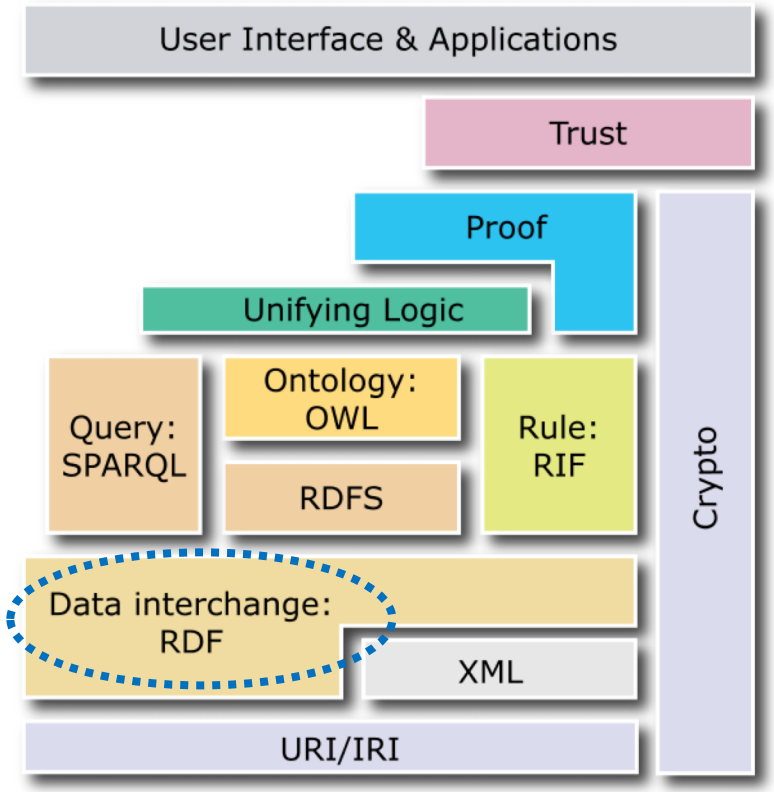
- RDF triple



- RDF graph



- Semantic Web layer cake



Large-scale KGs

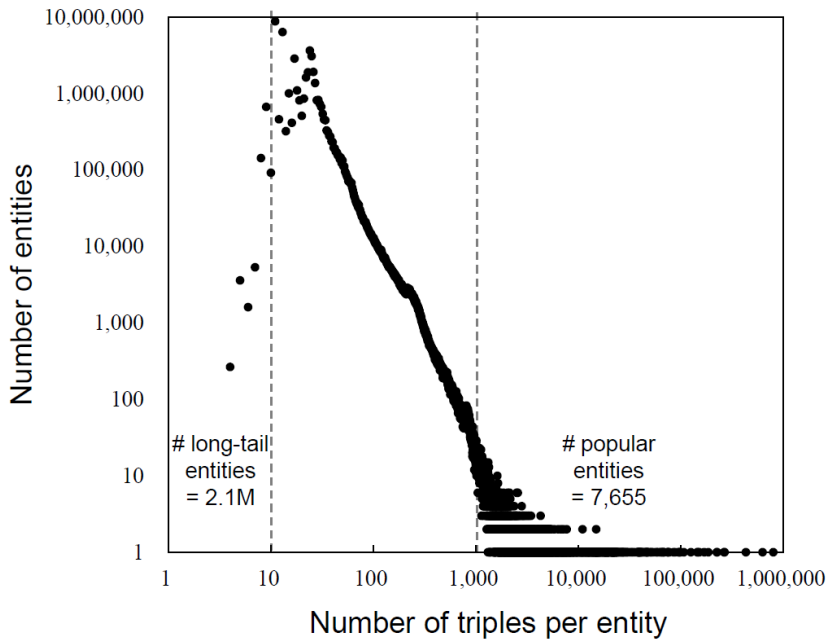
Names	Sizes
DBpedia (2016-10)	English: 6.6 million entities, 13 billion triples, 760 concepts, 3,079 properties 134 languages
YAGO4	64 million entities, 2 billion triples, 116 properties
Freebase	49 million entities, 3 billion triples, 53 thousand concepts, 78 thousand properties
Google	5 billion entities, 500 billion triples
Wikidata, Probase, WolframAlpha …	

Adoption of KGs

- Frank van Harmelen, late 2019, https://www.slideshare.net/Frank.van.Harmelen

Challenges in KG Construction

- KGs are far from complete
  - About 2.1 million entities in Freebase have < 10 facts per entity, meanwhile 7,000+ entities have > 1,000 facts
    - The power-law distribution

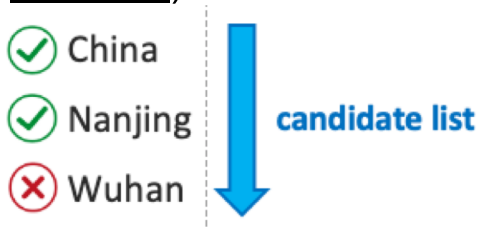


- Two important tasks are
  1. **Link prediction (a.k.a. KG completion):** complete missing facts in **a single KG**
     - E.g., predict **?** in (*Nanjing University*, *located in*, **?**) or (**?**, *located in*, *Nanjing*)
  2. **Entity alignment:** find entities in **different KGs** denoting the same real-world object
     - E.g., *instant immersion spanish deluxe 2.0* ≅ *instant immers spanish dlux 2 ?*

Link Prediction

- Automation is inevitable
  - The cost of curating a fact manually is much more expensive than that of automatic creation, by a factor of 15 to 250 [Paulheim@ISWC2018]

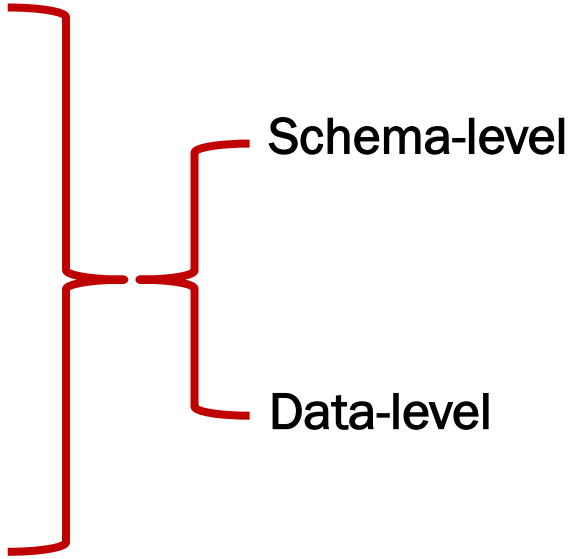
- A ranking problem
  - E.g., predict **?** in (*Nanjing University*, *located in*, **?**)



  - Relevant to several NLP tasks
    - Relation extraction, slot filling …
    - TransE expects, for a triple (h, r, t) , $\mathbf{h} + \mathbf{r} = \mathbf{t}$ [Bordes+@NIPS2013]

Heterogeneity

- Since long long time ago …
  - Syntactic
    - e.g., “Wei Hu” vs. “HU, Wei”
  - Terminological
    - e.g., “notebook” vs. “laptop” vs. 笔记本电脑
  - Semantic
    - e.g., $\mathrm{hasSon}(x, y)$ vs. $\mathrm{hasChild}(x, y) \sqcap \mathrm{Male}(y)$
  - Pragmatic



KGs have reached a scale in billions of triples and are evolving all the time!

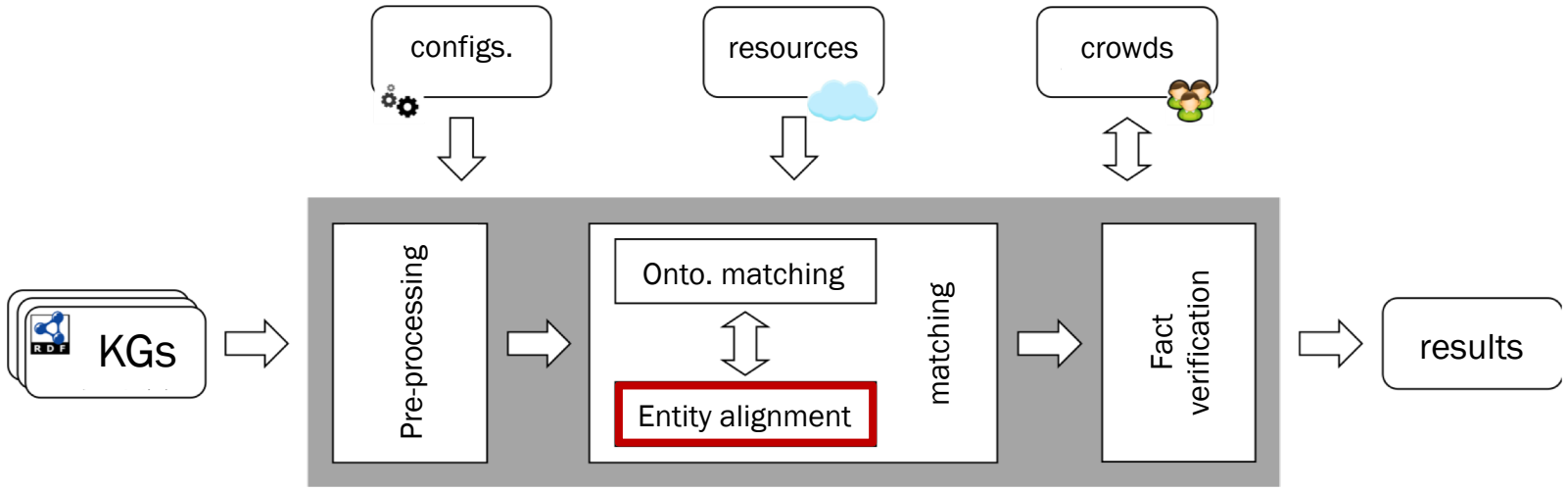
Entity Alignment

- A **KG** is defined as a 7-tuple $G = (\mathcal{E}, \mathcal{R}, \mathcal{A}, \mathcal{V}, \mathcal{N}, \mathcal{X}, \mathcal{Y})$, where
  - $\mathcal{E}, \mathcal{R}, \mathcal{A}$ and $\mathcal{V}$ denote the sets of entities, relations, attributes, literals, respectively
  - $\mathcal{N} \subseteq \mathcal{E} \times \mathcal{V}$ denotes the names of entities, $\mathcal{X} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ denotes the relation triples, $\mathcal{Y} \subseteq \mathcal{E} \times \mathcal{A} \times \mathcal{V}$ denotes the attribute triples

- Given a source KG $G_1 = (\mathcal{E}_1, \mathcal{R}_1, \mathcal{A}_1, \mathcal{V}_1, \mathcal{N}_1, \mathcal{X}_1, \mathcal{Y}_1)$ and a target KG $G_2 = (\mathcal{E}_2, \mathcal{R}_2, \mathcal{A}_2, \mathcal{V}_2, \mathcal{N}_2, \mathcal{X}_2, \mathcal{Y}_2)$, **entity alignment** aims to find a set of identical entities $M = \{(e_i, e_j) \in \mathcal{E}_1 \times \mathcal{E}_2 \mid e_i \cong e_j\}$, where
  - "$\cong$" denotes the equivalence relationship

Traditional Approaches to Entity Alignment

- Equivalence reasoning
  - E.g., Inverse functional property (IFP) relation
    - IFP: a value can only be the value of this property for a single object
      - E.g., $\langle s_1, \boldsymbol{foaf:mbox}, whu@nju.edu.cn \rangle, \langle s_2, \boldsymbol{foaf:mbox}, whu@nju.edu.cn \rangle \rightarrow s_1 \cong s_2$
- Similarity computation
  - Pairwise: string, date, geographic, aggregation …
  - Collective: leverage the relationships of entities to improve accuracy
- Machine learning, crowdsourcing …



End of Part I