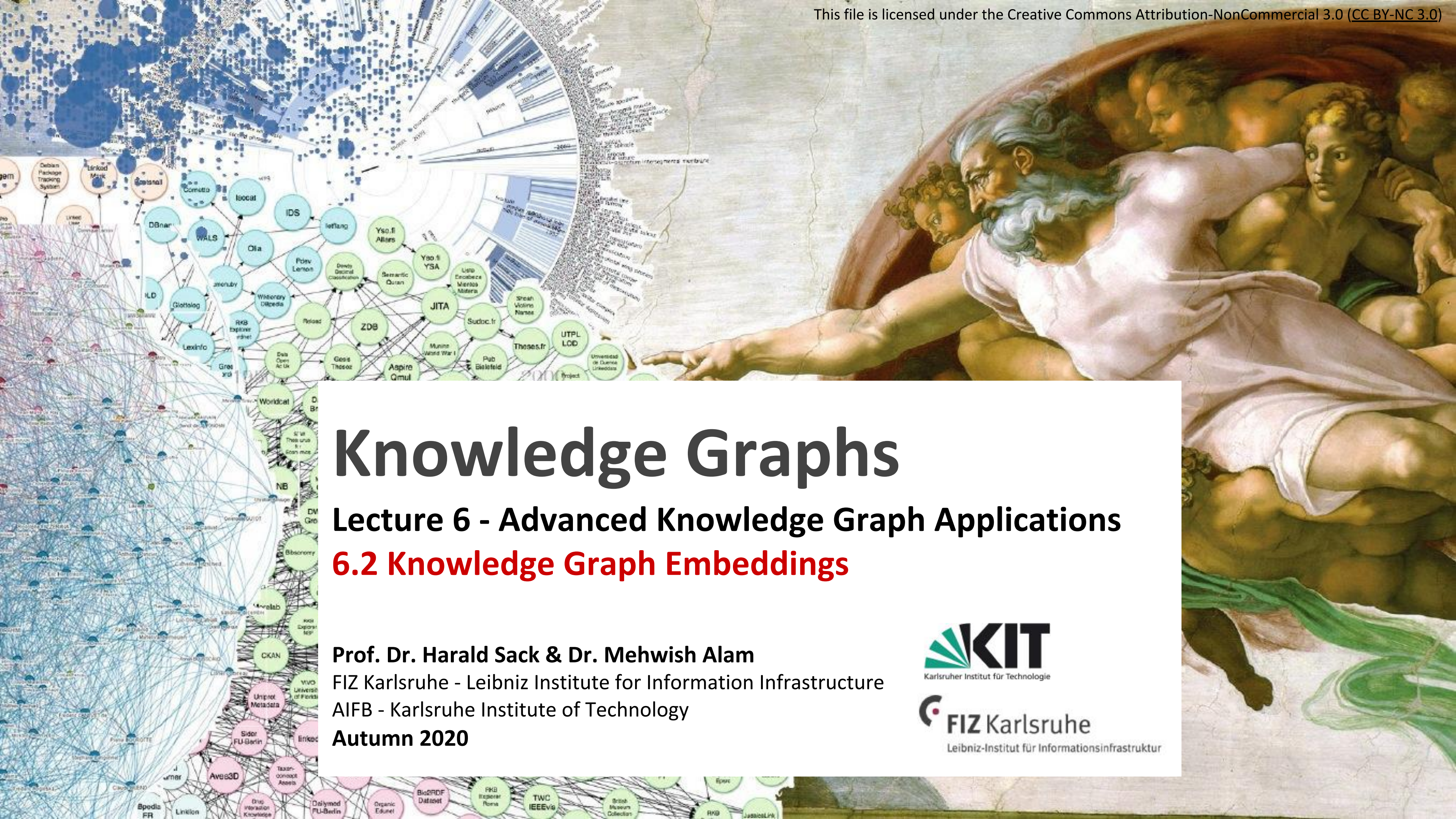


Knowledge Graphs

Lecture 6 - Advanced Knowledge Graph Applications

6.2 Knowledge Graph Embeddings

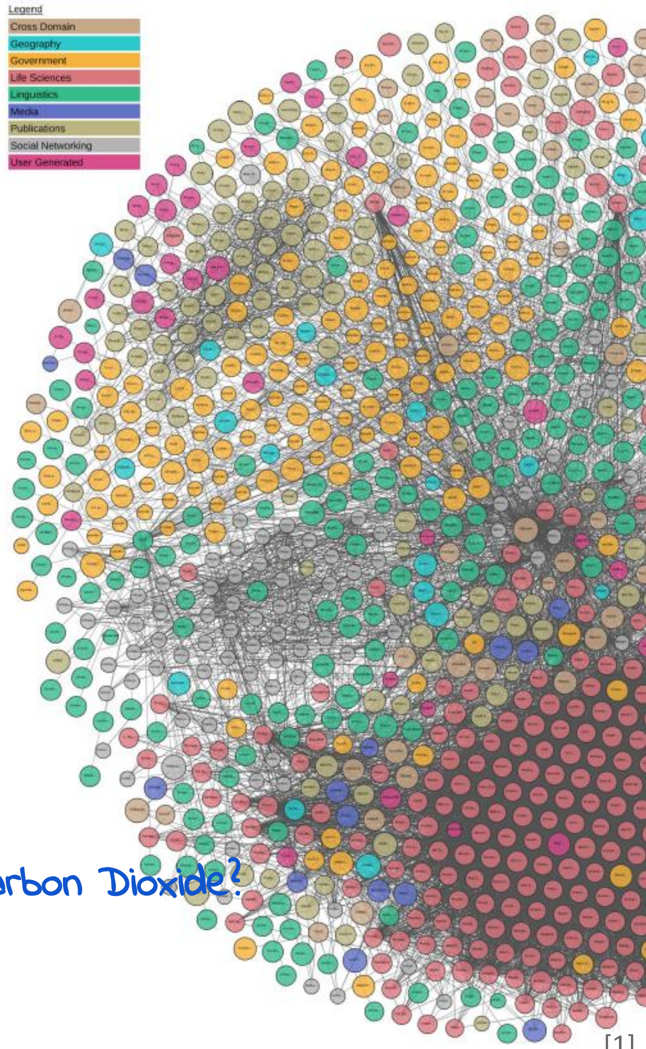
Prof. Dr. Harald Sack & Dr. Mehwish Alam
FIZ Karlsruhe - Leibniz Institute for Information Infrastructure
AIFB - Karlsruhe Institute of Technology
Autumn 2020

Karlsruher Institut für Technologie
FIZ Karlsruhe
Leibniz-Institut für Informationsinfrastruktur

Knowledge Graphs

Lecture 6: Advanced Knowledge Graph Applications

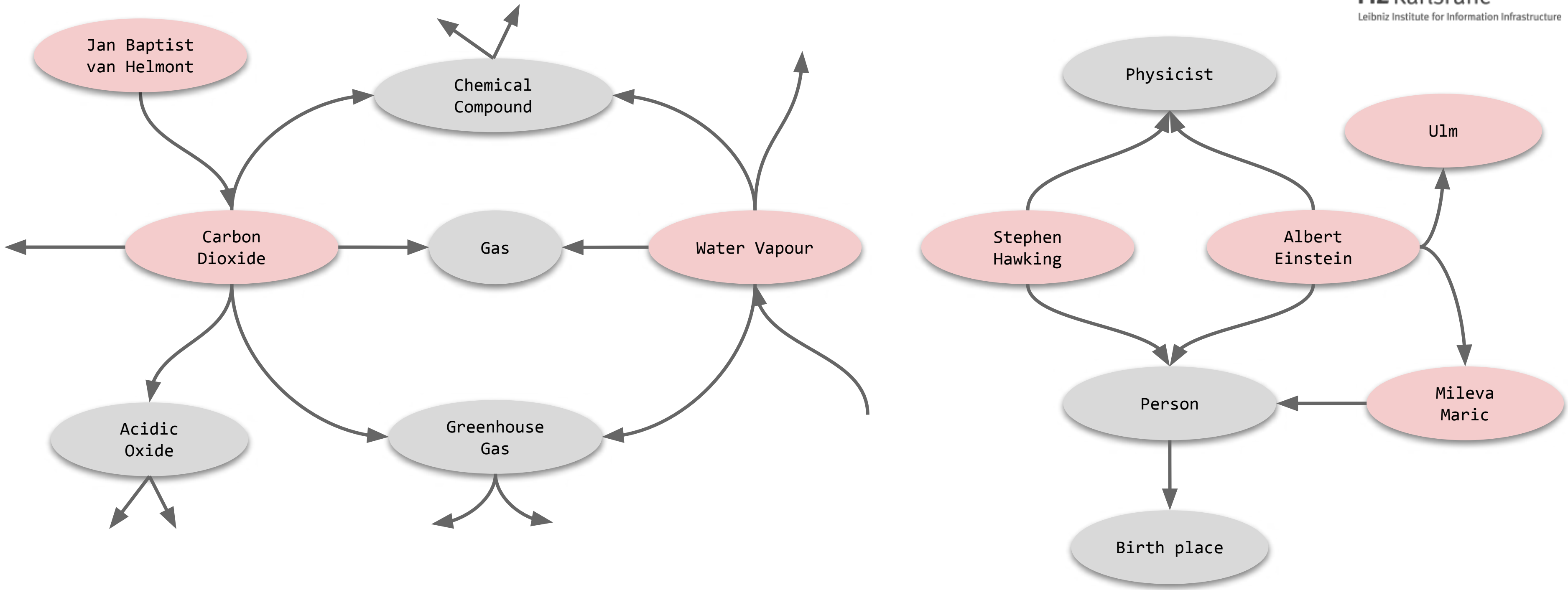
Semantic Similarity



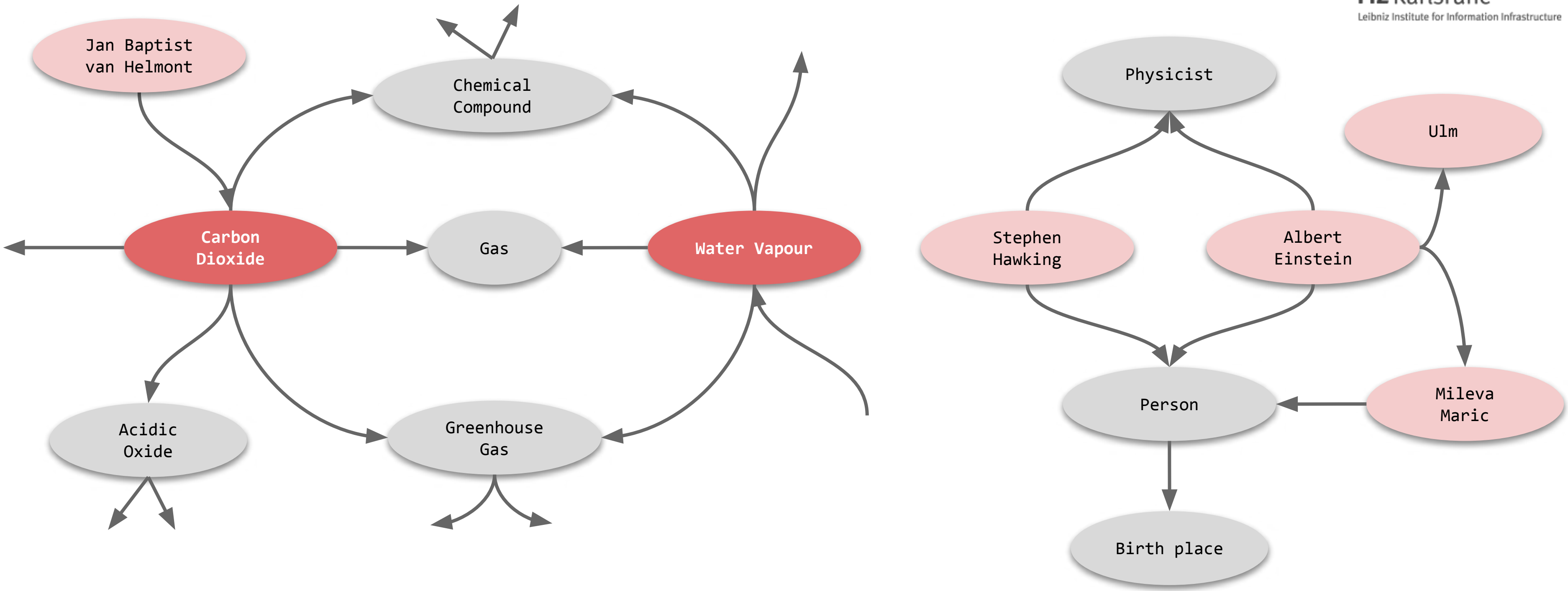
- For search and retrieval systems, **semantic similarity of entities** is an important feature, as e.g.
  - Given an entity find the most similar entities
  - Given an entity find the most similar documents
  - Given a document find the most similar documents, etc.
- **When are two entities (semantically) similar?**
  - If they can be described by the same/similar facts, as e.g.
  - Carbon Dioxide is a Greenhouse Gas and
    water vapour is a Greenhouse Gas
  - Albert Einstein is a Physicist and
    Stephen Hawking is a Physicist
  - **Is Stephen Hawking more similar to Albert Einstein or to Carbon Dioxide?**

[1]

Semantic Similarity

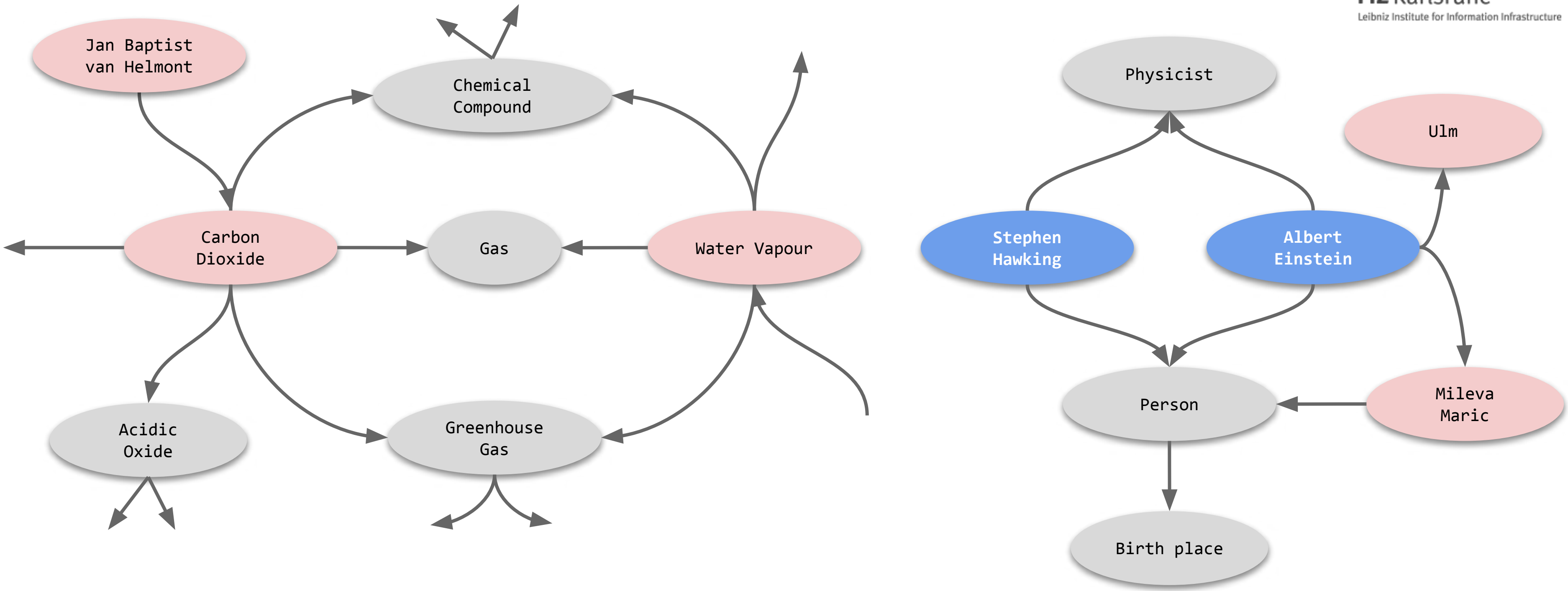


Semantic Similarity



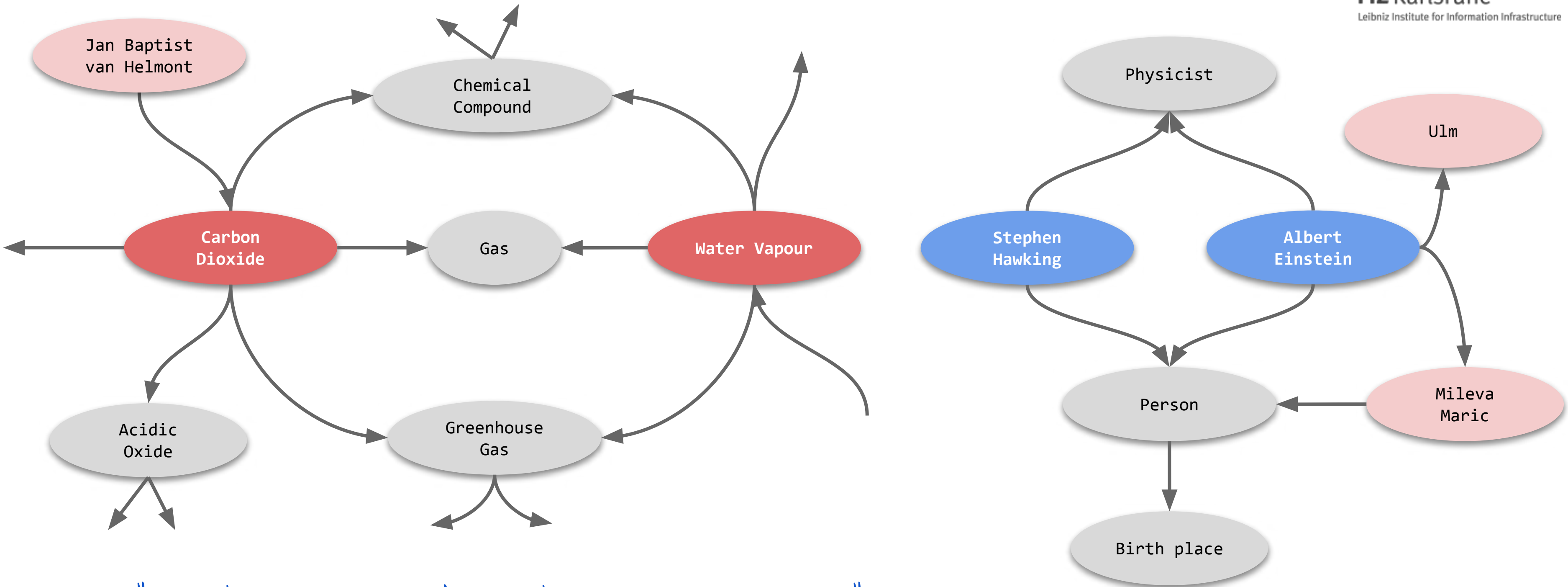
- Carbon Dioxide and water vapour share similar (structural) context in the graph

Semantic Similarity



- Stephen Hawking and Albert Einstein share similar (structural) context in the graph

Semantic Similarity



- "You shall know a node by the company it keeps"
- i.e. similar nodes can be identified by having the same/similar environment (context)
- adjacency based similarity

Semantic Similarity

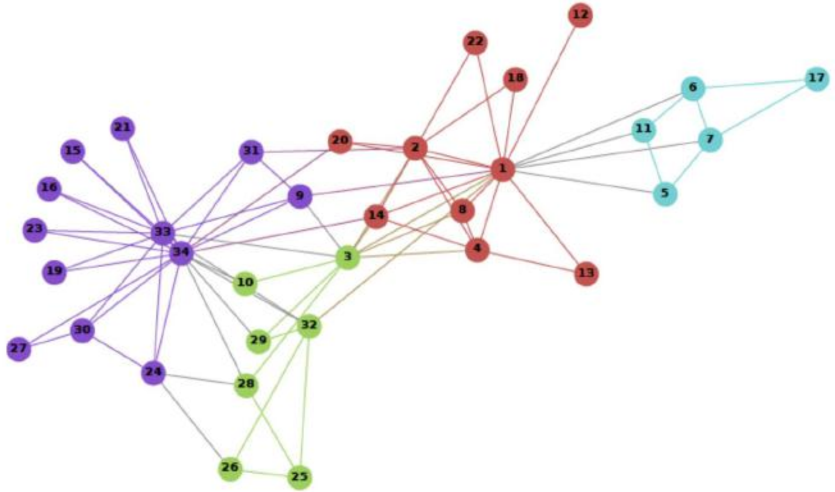
- In a Knowledge Graph,
  - **similar entities** are represented by nodes that are connected to **similar/same facts**
  - i.e. that are connected to **similar graph structures**
  - To identify **similar entities**, we have to identify **similar graph structures**

- **Problem:**
  - Algorithms to determine semantic similarity in graphs are of high complexity, i.e. with large KGs, as e.g. Wikidata, they don’t work efficiently.

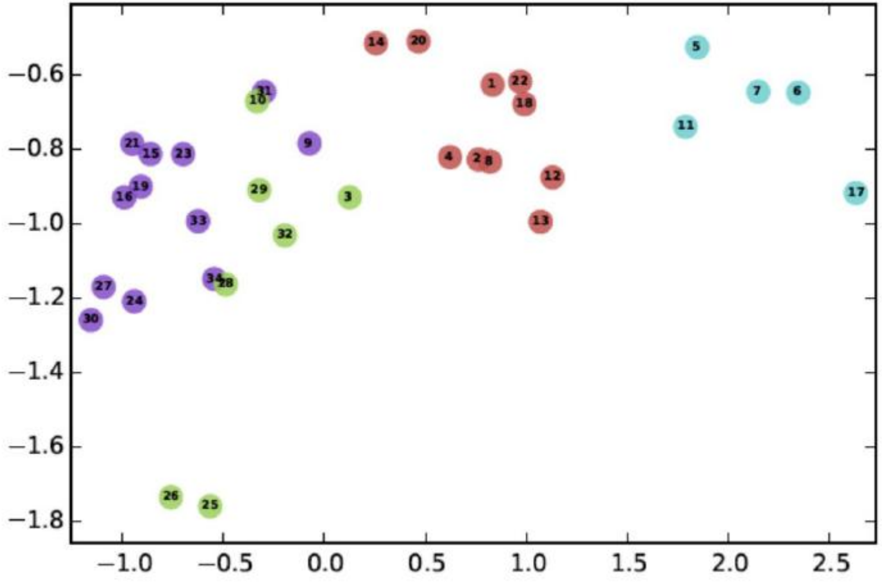
- **Idea**:
  - Approximate the problem by transferring it from graph structures to vector spaces That are easier to handle.

Vector Representations

(Knowledge) Graph Representation



Vector Space Representation



Idea: Find embedding of nodes in a low-dimensional vector space
so that **"similar" nodes in the graph** have **vector embeddings that are close together**

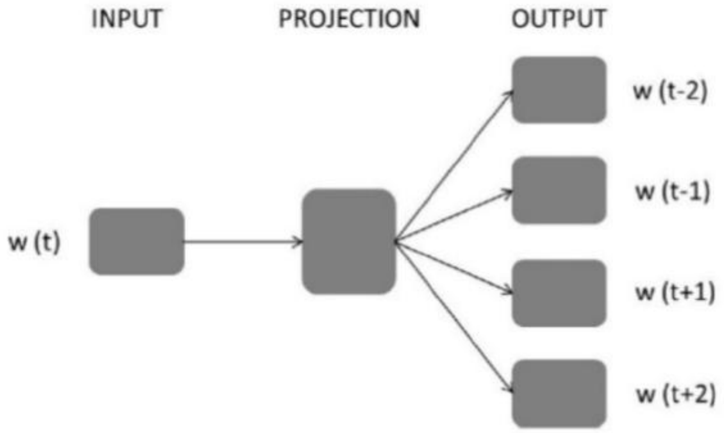
Excursion: Word Embeddings

- **Word Embeddings** map natural language words to a dense vector representation
- **Basic Assumption:** Similar words occur in similar contexts:
  (Carbon Dioxide, Water Vapour, Methane) is one of the driving agents of climate change.
  Climate change is caused by greenhouse gases like (Carbon Dioxide, Water Vapour, Methane)
- **Basic idea:** instead of counting co-occurrences of words, predict the likelihood of the appearance of words in the neighborhood of others
- Train a predictor (neural network) that can predict a word from its context (**CBOW**) or the context from a given word (**Skip Gram**)

Mikolov, Tomas; et al. (2013). "Efficient Estimation of Word Representations in Vector Space". arXiv:1301.3781

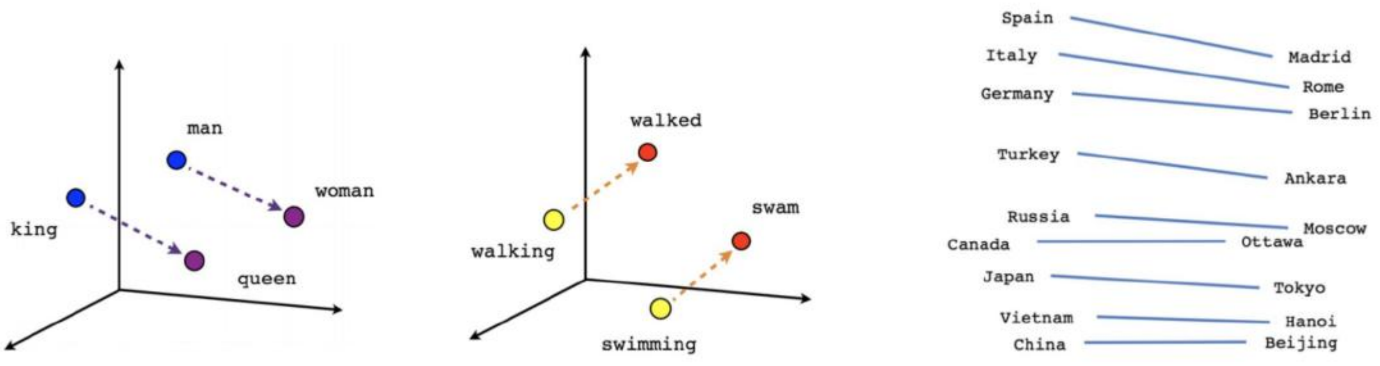
Excursion: Word Embeddings

- **Skip Gram:**
  - Train a neural network with one hidden layer
  - Use output at hidden layer as vector representations
- **Observation:**
  - *Carbon Dioxide, Water Vapour, Methane* will activate similar context words
  - i.e. their output weights at the projection layer have to be similar



Mikolov, Tomas; et al. (2013). "Efficient Estimation of Word Representations in Vector Space". arXiv:1301.3781

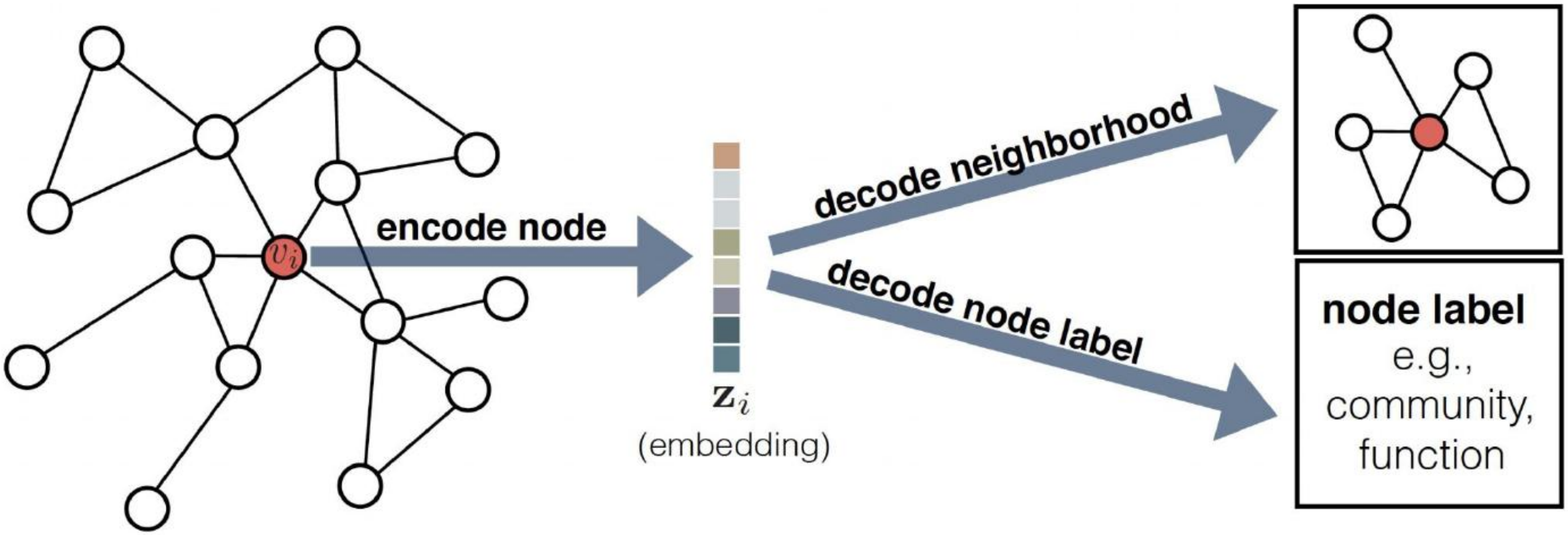
Word Embeddings



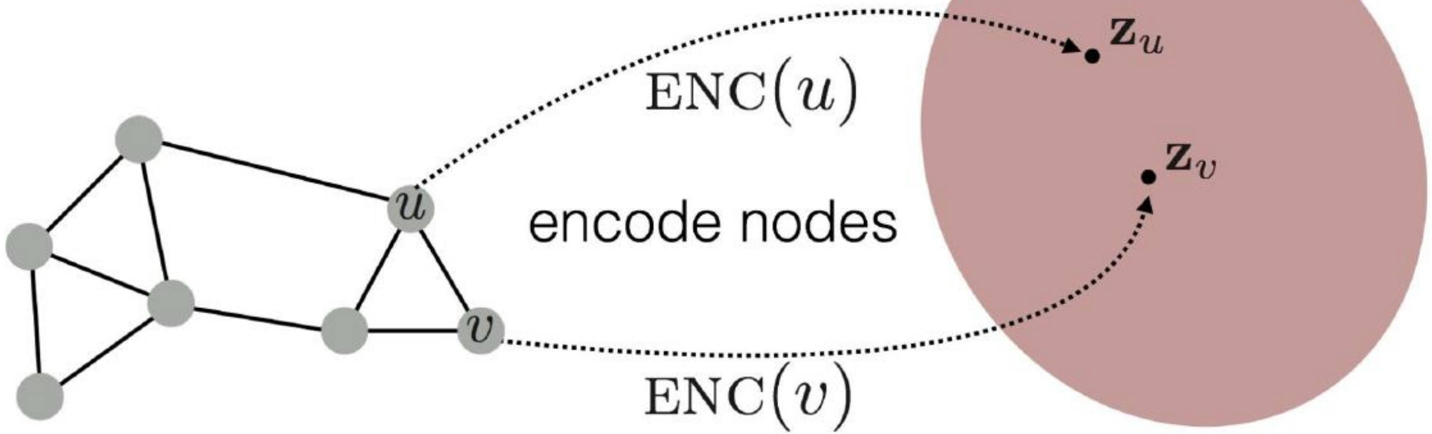
Male-Female Verb tense Country-Capital

- Semantics of words is preserved, i.e. it enables semantic arithmetic operations as e.g. analogies
  - “king” - “man” ≈ “queen” - “woman”
  - “king” - “man” + “woman” ≈ “queen”

Graph Embeddings



Graph Embeddings - Encoder-Decoder Approach



- The goal is to encode the nodes of the graph in a way so that **similarity in the embedding space** (e.g., dot product) **approximates similarity in the original network**.
- ENC: $N \rightarrow \mathbb{R}^d$, $u,v \in N$, $\mathrm{ENC}(u) = z_u \in \mathbb{R}^d$, $\mathrm{ENC}(v) = z_v \in \mathbb{R}^d$
- DEC: $\mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^+$, $\mathrm{DEC}(\mathrm{ENC}(u), \mathrm{ENC}(v)) = \mathrm{DEC}(z_v , z_u) \approx \mathrm{similarity}\ (u,v)$

Learning Graph Embeddings

Karlsruher Institut für Technologie

FIZ Karlsruhe
Leibniz Institute for Information Infrastructure

1) Define an **encoder ENC** (i.e., a mapping from nodes to embeddings)
2) Define a **node similarity function** that specifies how relationships in vector space map to relationships in the original network.
3) Optimize the parameters of the encoder so that:

$$\text{similarity}(u, v) = z_v^T z_u$$

Knowledge Graph Embeddings

Many ways to generate Knowledge Graph Embeddings:

- **Translational Methods**: TransE, TransH, TransR, TransEdge, …
- **Rotation Based**: RotatE
- **Graph Convolutional Networks**: R-GCN, TransGCN
- **Walk-Based Methods**: DeepWalk, RDF2Vec

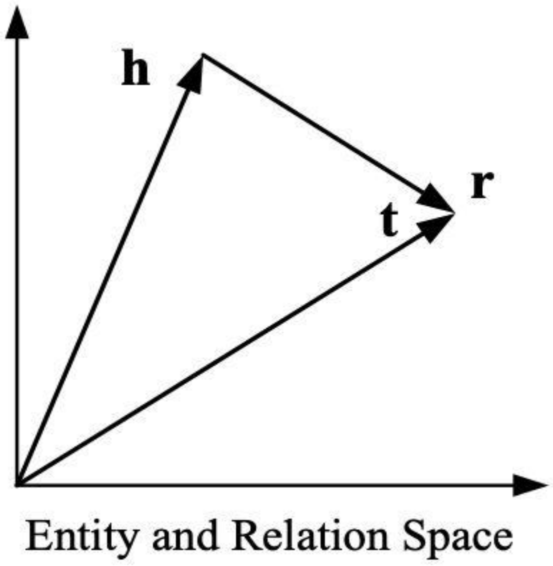
Translational Distance Models

- Exploit distance-based scoring functions
- Measure the **plausibility of a fact** as the **distance between two entities**
- A translation carried out by the relation.
- **Models**: TransE, TransH, TransR, TransD, TransSparse, TransM, TransEdge

Wang et al., Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering (TKDE), 2017.

TransE

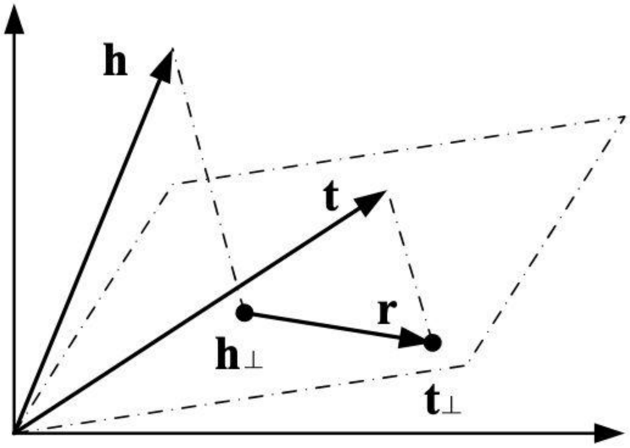
- Entities and relations are embedded into **same vector space**.
- h = head, t = tail, r = relation
- Relation r is considered as translation from h to t
- Learning Assumption **h+r≈t**
- **Problem:** Symmetric functions,
  1-N / N-1 / N-N functions



Bordes et al, Translating Embeddings for Modeling Multi-relational Data, NIPS 2013

TransH

- From original space to Hyperplane
- TransH enables **different roles of an entity in different relations**.
- Entities h and t are projected into specific **hyperplane of relation r**.
- Then predict new links based on translation on hyperplane.



Entity and Relation Space

Wang et al., Knowledge graph embedding by translating on hyperplanes. AAAI, 2014.

Graph Convolutional Network

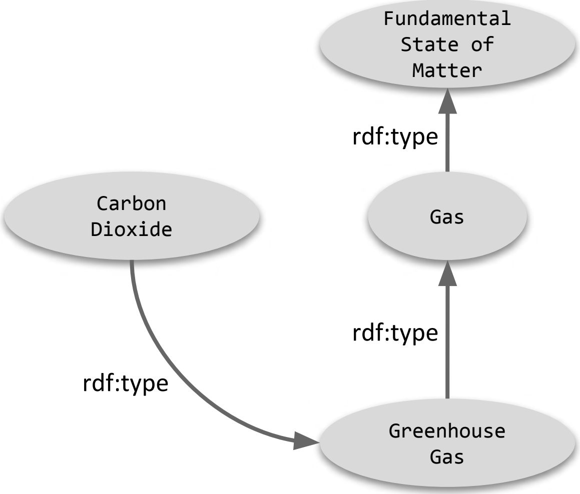
- **Graph Convolutional Networks (GCN)**
  - modeling structured neighborhood information of **unlabeled** and **undirected** graphs with **convolution operations**
- **Relational Graph Convolutional Network (R-GCN)**
  - Models Relational Data using GCN where Knowledge Graphs are considered as **directed labeled multigraphs**.
  - Models in RGCN
    - **Link Prediction:**
      - **an encoder:** an R-GCN producing latent feature representations of entities,
      - **a decoder:** a tensor factorization model exploiting these representations to predict labeled edges

RDF2Vec

- Word2vec operates on sentences, i.e. sequences of words
- **RDF2Vec Basic Idea**:
  - Generate "sentences" from knowledge graph, i.e. sequences of interconnected RDF triples

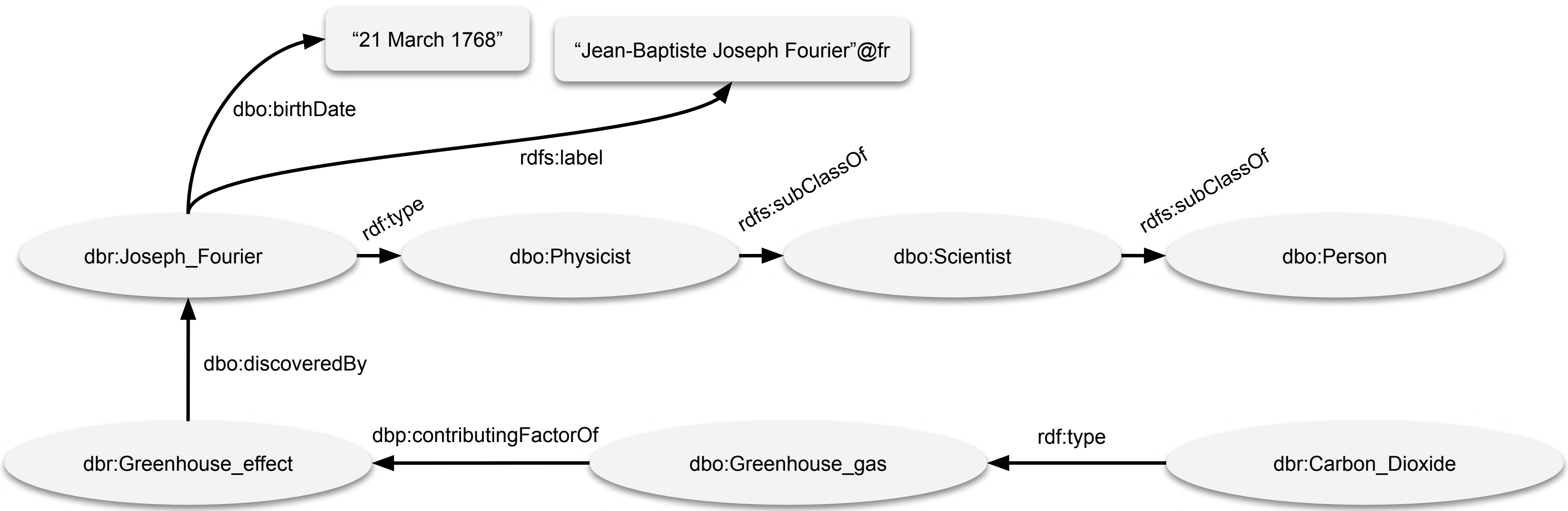
```
:CarbonDioxide rdf:type :GreenhouseGas.
:GreenhouseGas, rdf:type, :Gas.
:Gas, rdf:type, :FundamentalStateOfMatter.
```

  - Selection strategies:
    - Depth first search
    - Breadth first search
    - Random walk
    - RDF Graph Kernels



Petar Ristoski and Heiko Paulheim RDF2Vec: RDF graph embeddings for data mining, ISWC 2016

Graph Walks RDF2Vec



Generated Sequences of depth = 3:

- dbr:Carbon_Dioxide→ rdf:type→dbo:Greenhouse_gas → dbp:contributingFactorOf → dbr:Greenhouse_effect → dbo:discoveredBy → dbr:Joseph_Fourier

Libraries for KG Embedding

PyTorch BigGraph

https://github.com/facebookresearch/PyTorch-BigGraph

AmpliGraph

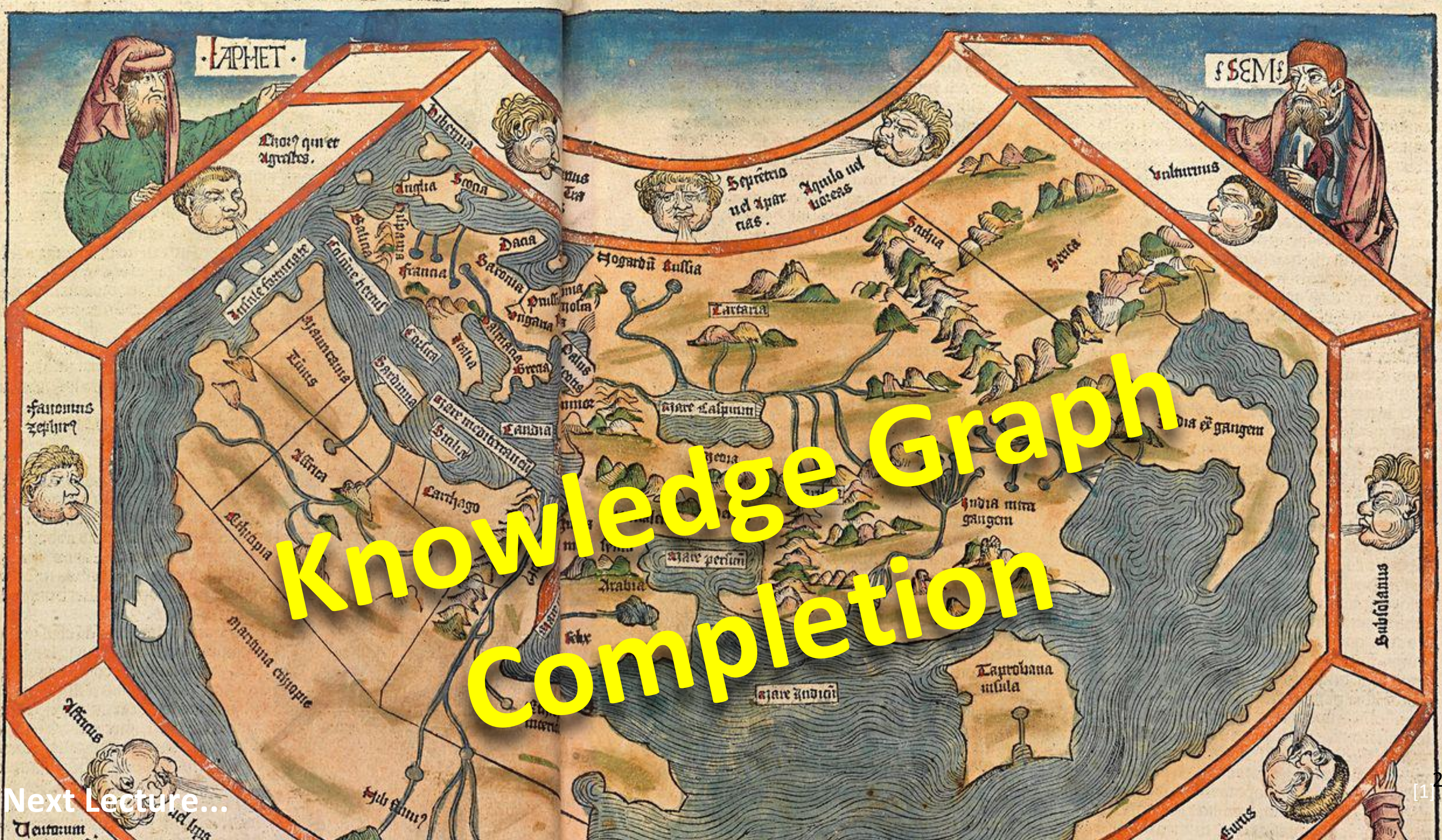
https://github.com/Accenture/AmpliGraph

PyKeen

https://github.com/SmartDataAnalytics/PyKEEN

OpenKE

http://openke.thunlp.org/



Knowledge Graph Completion

Next Lecture...

[1]

Picture References:

- [1] John P. McCrae, The Linked Open Data Cloud, [CC-BY-4.0] https://lod-cloud.net/
- [2] Hartman Schedel, Nuremberg Chronicle, 1493, https://commons.wikimedia.org/wiki/File:Nuremberg_chronicles_-_f_13b.png