Combining learning and reasoning: new challenges for knowledge graphs

Frank van Harmelen

Vrije Universiteit Amsterdam

Let's first celebrate our successes!

Tencent 腾讯

UniProt

USGS

Google

Bing

Alibaba.com

Baidu 百度

PubMed

facebook

ANTONI VAN LEEUWENHOEK FOUNDATION

MAASTRO

DEUTSCHE NATIONAL BIBLIOTHEK

The New York Times

BBC

europeana

GETTY

NXP

REUTERS

KUNGL. BIBLIOTEKET

National Library of Sweden

RENAULT

EPA United States Environmental Protection Agency

European Commission

IOS Press

Walmart

SIEMENS

BEST BUY

Deloitte.

ELSEVIER

SPRINGER NATURE

amazon.com

accenture

Just some of the use cases

- **Google** = meaningful **search**
- **NXP** = data **integration**
- **BBC** = content **re-use**
- **Amazon** = product search
- **data.gov** = data-**publishing**

Yahoo, Bing

Oracle DB, IBM DB2

Reuters,
New York Times, Guardian

Best-Buy, Sears, Kmart,,
Volkswagen, Renault
GoodRelations ontology,
schema.org

Knowledge Graphs in 4 principles

1. **Give all things a name**

2. **Make a graph of relations between the things**

This makes a ***Giant Graph***

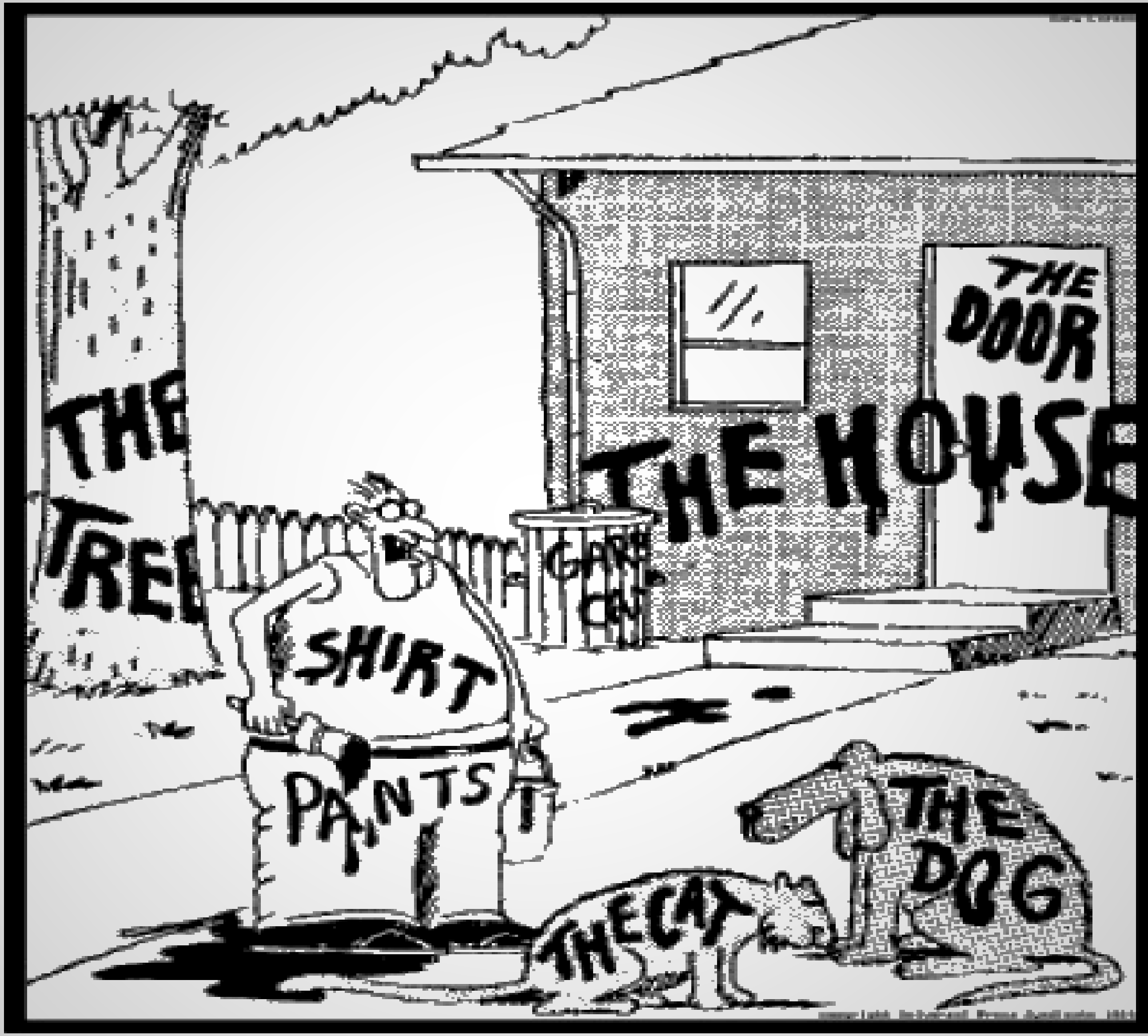
3. **Make sure all names are URIs**

This makes a ***Giant Global Graph***

4. **Add semantics (= predictable inferenc**e)

This makes a ***Giant Global Knowledge Graph***

P1. Give all things a name

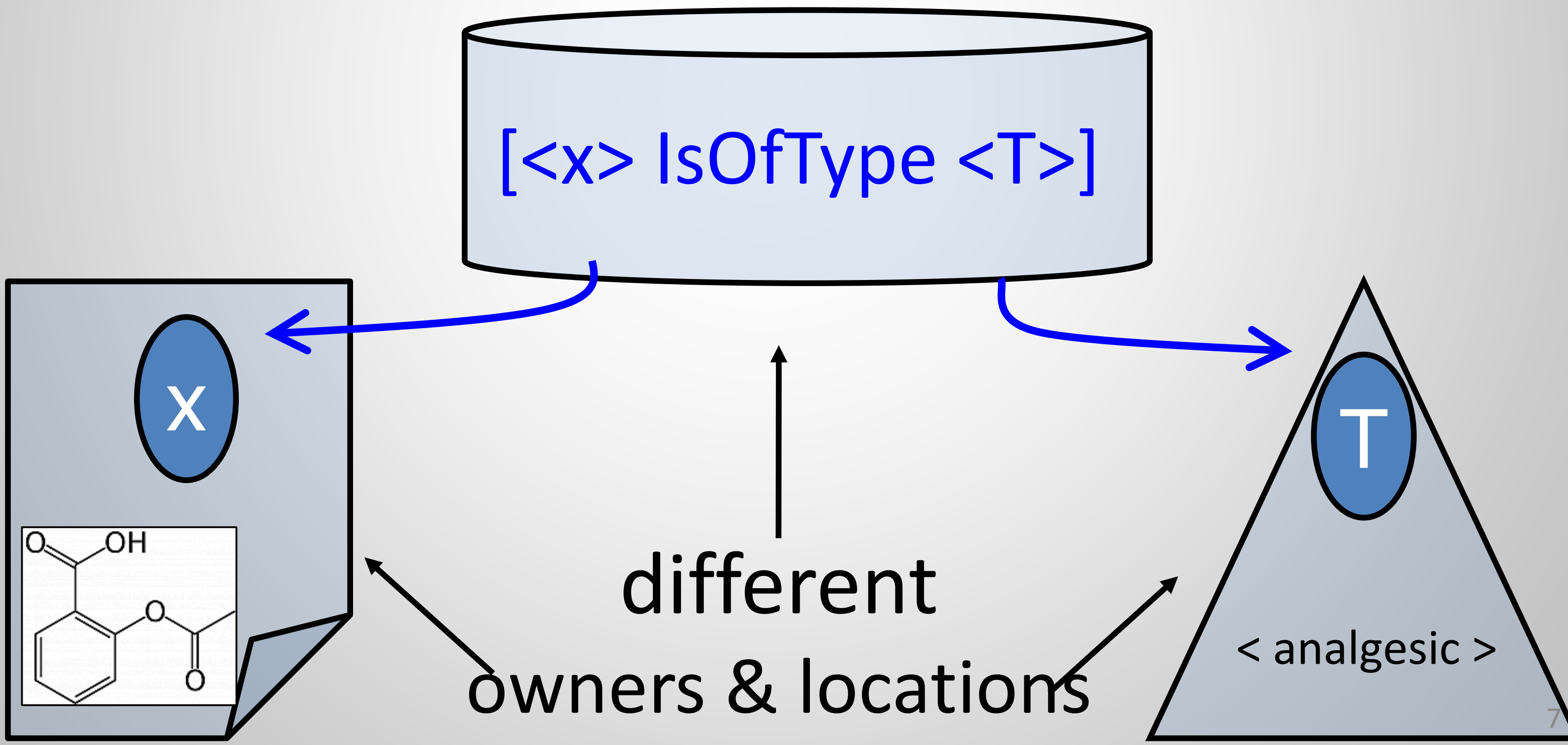


"Now! *That* should clear up a few things around here!"

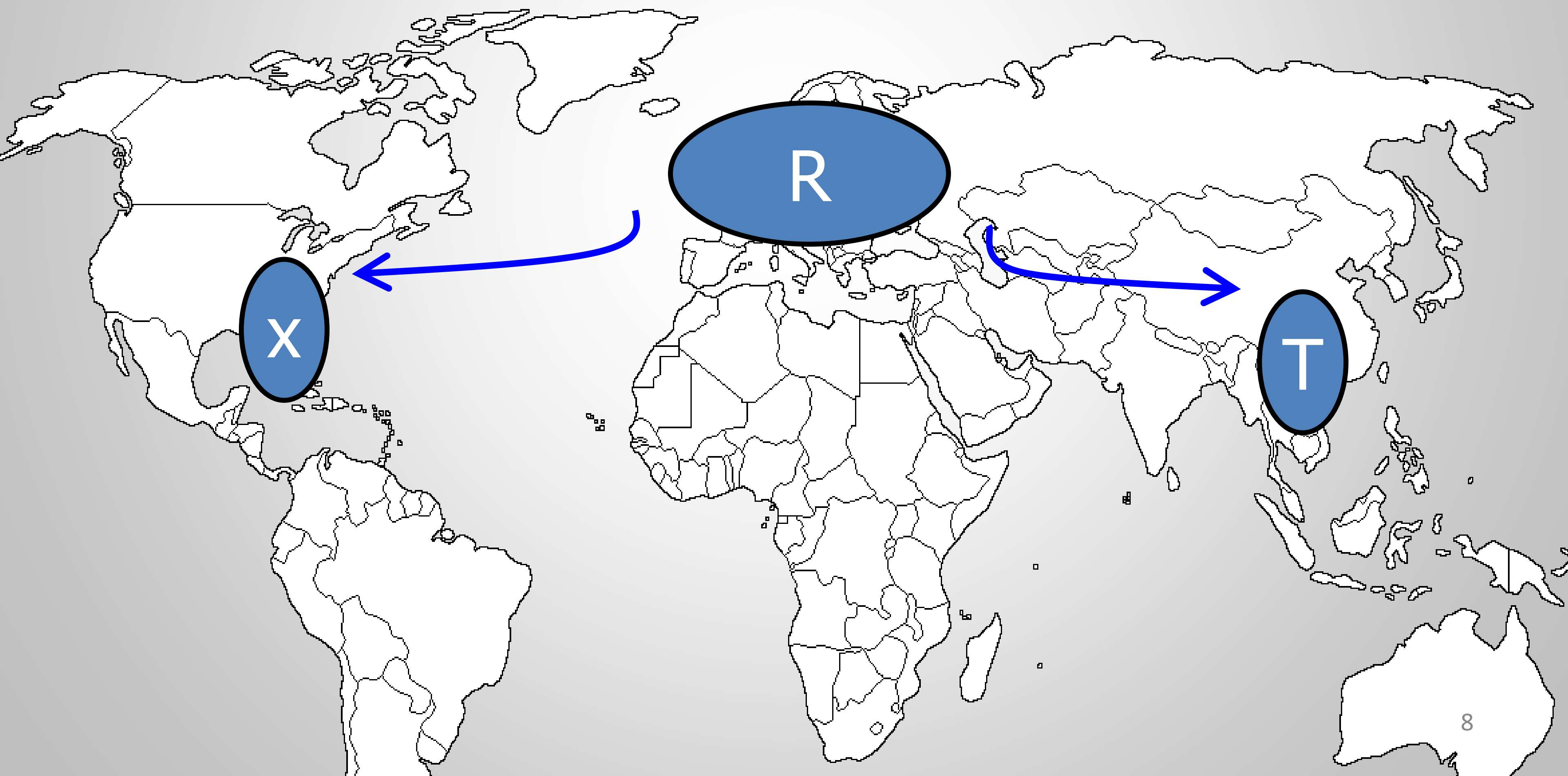
Data URIs

P3. The names are addresses on the Web

On the Web,
anybody can say anything about anything



P3. The names are addresses on the Web



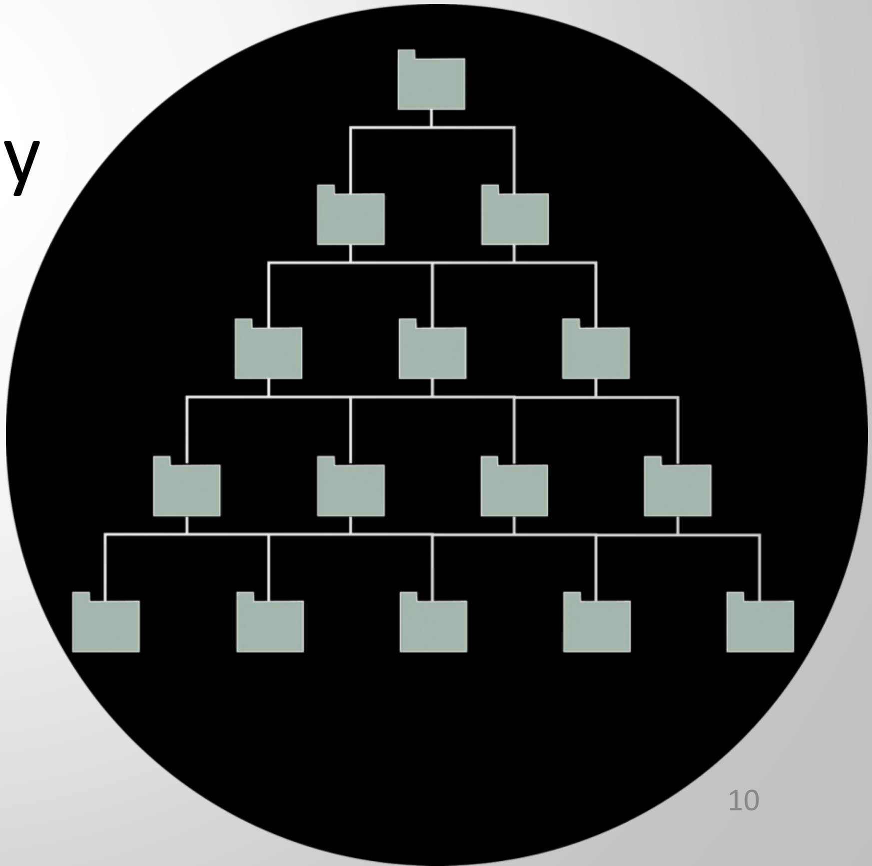
P1+P2+P3 = Giant Global Graph

RDF

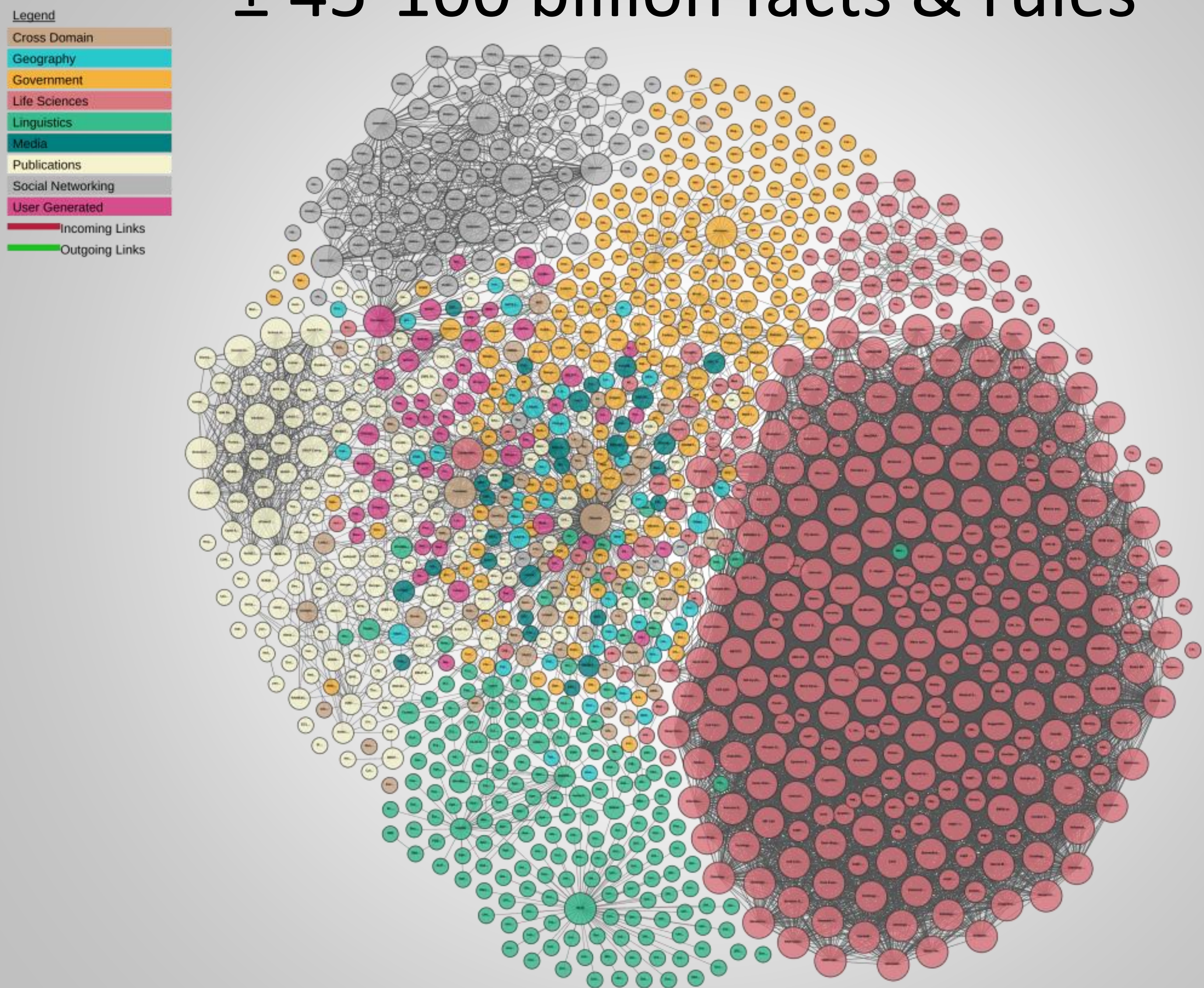
P4. explicit & formal semantics

- assign types to things
- assign types to relations
- organise types in a hierarchy
- empose constraints on possible interpretations

RDF OWL



± 45-100 billion facts & rules

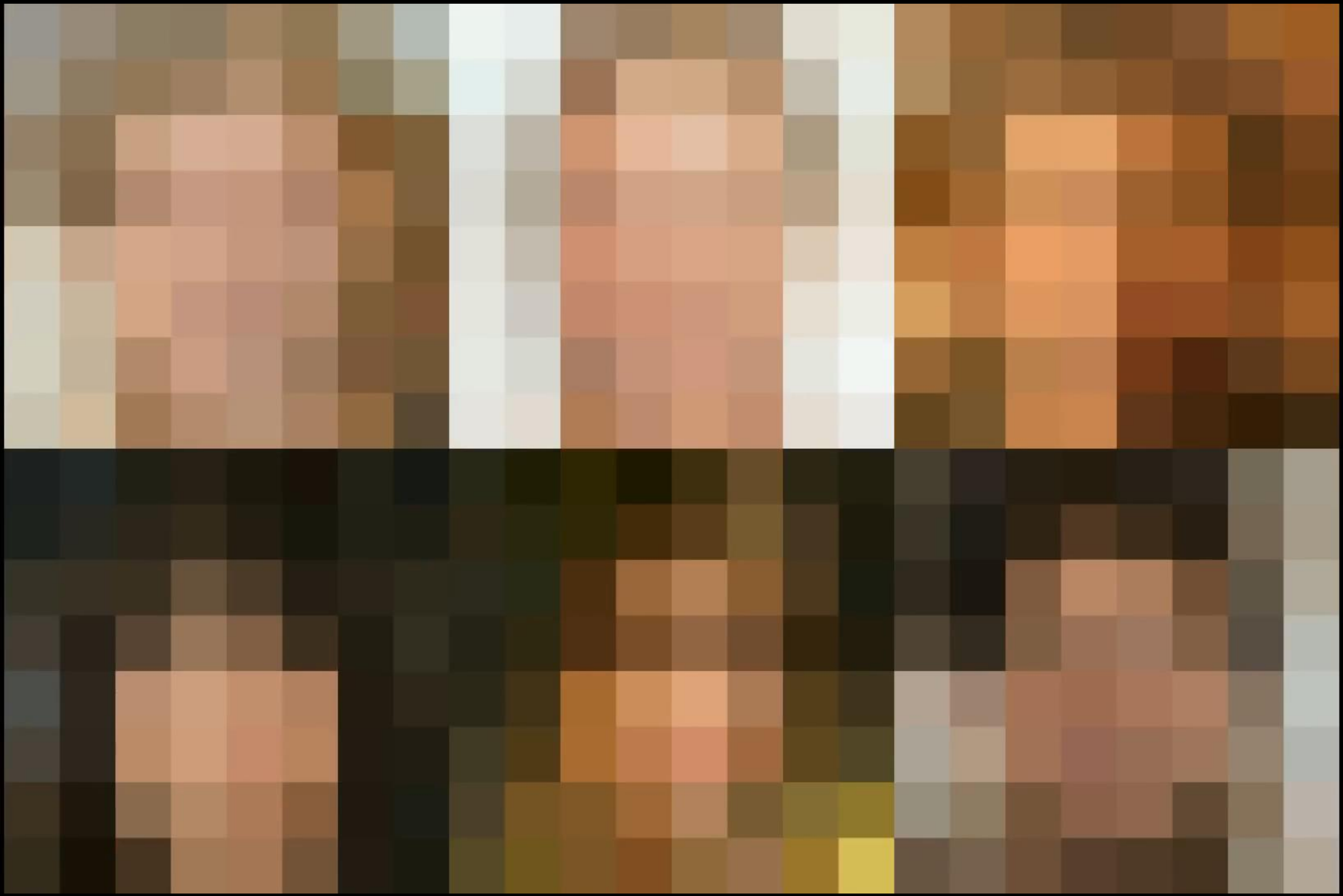


But the eyes of the world are elsewhere…

Supervised learning

Today

Unsupervised learning



8x8


DEEPART.io
DEEPART.io

Reinforcement learning

Robot Motor Skill Coordination with EM-based Reinforcement Learning

Petar Kormushev, Sylvain Calinon, and Darwin G. Caldwell

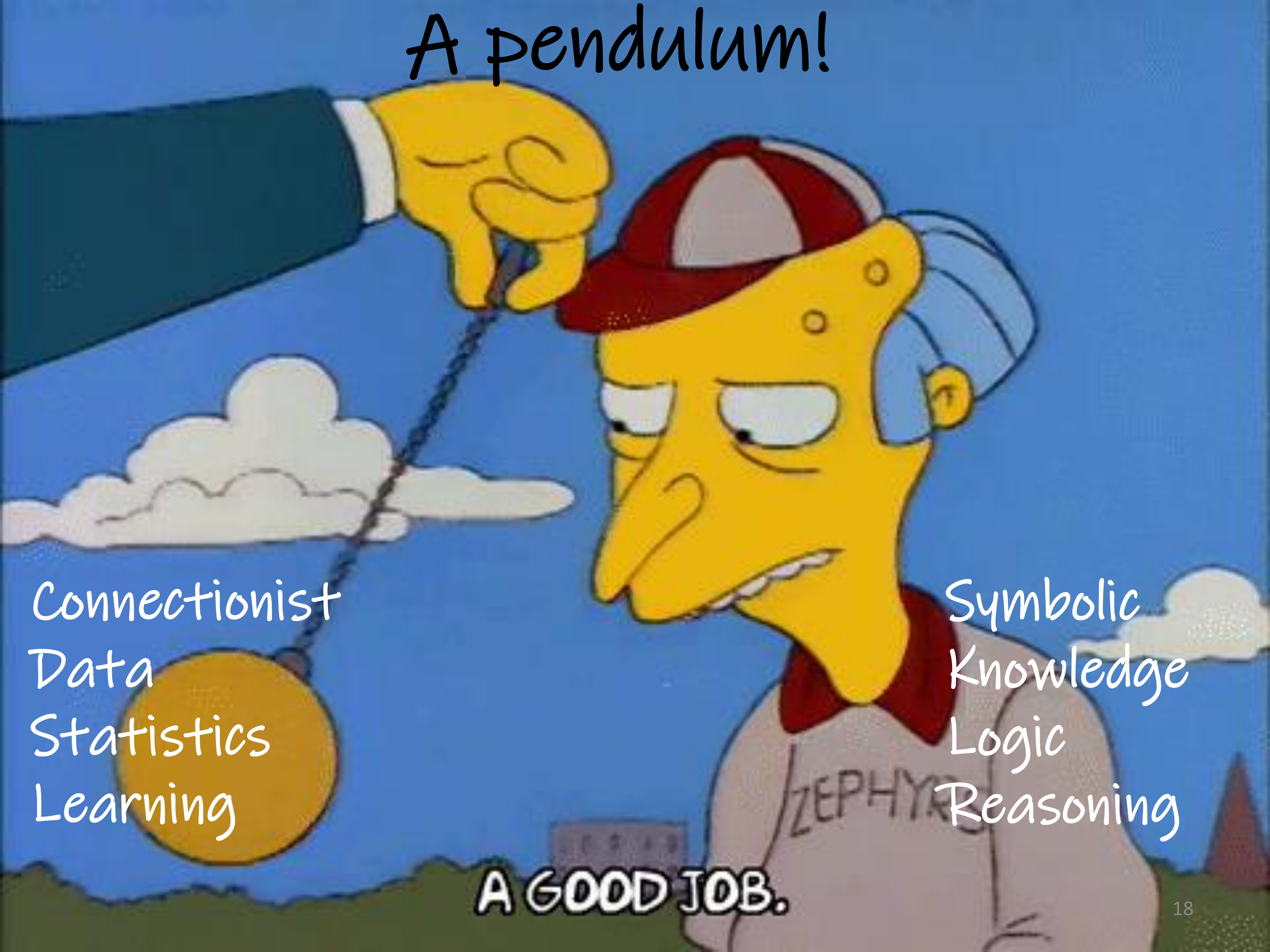
Italian Institute of Technology

A stairway?

Connectionist
Data
Statistics
Learning

Symbolic
Knowledge
Logic
Reasoning

A pendulum!

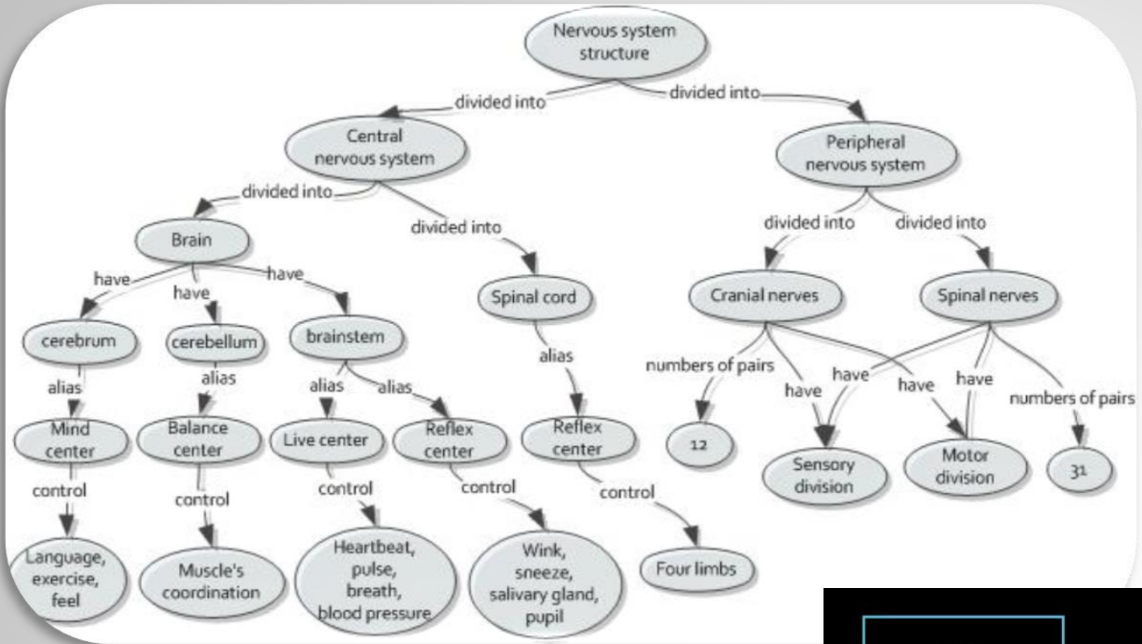


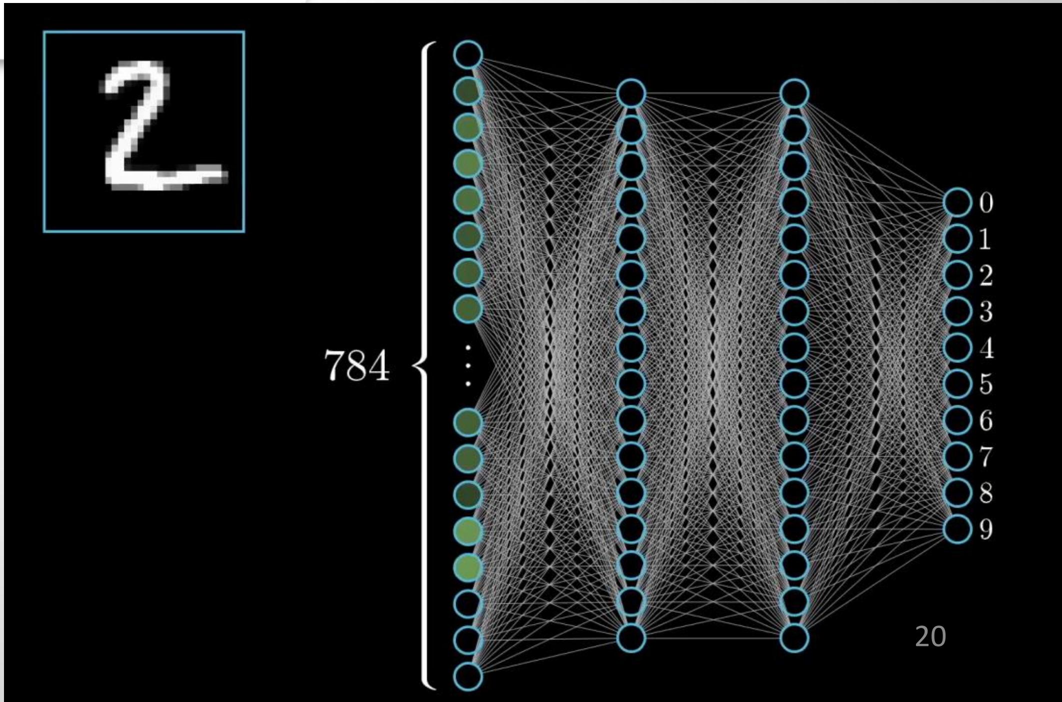
Connectionist
Data
Statistics
Learning

Symbolic
Knowledge
Logic
Reasoning

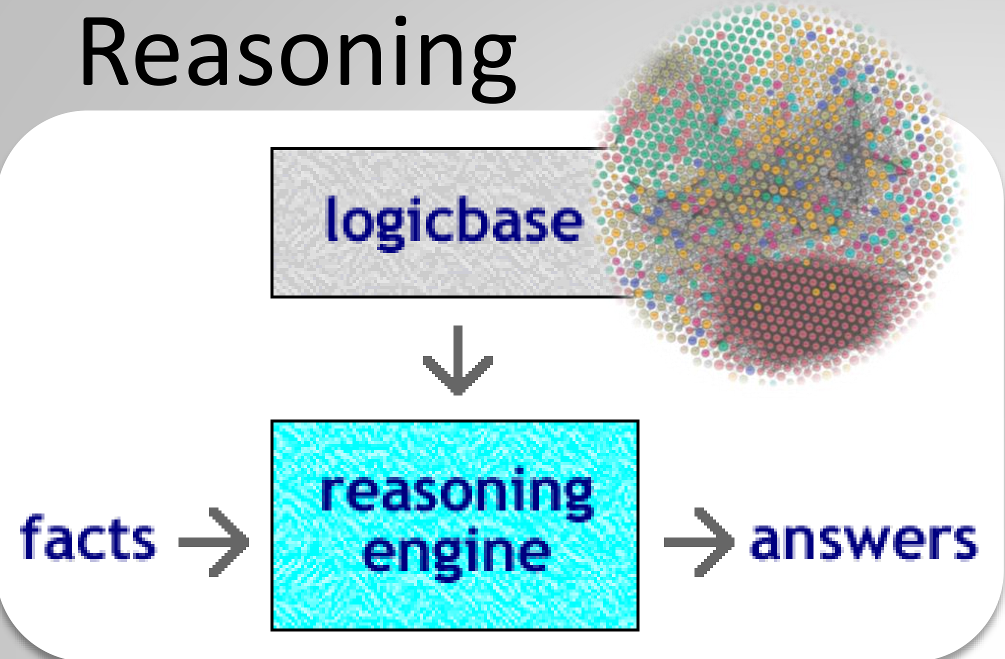
So let's compare

Representation

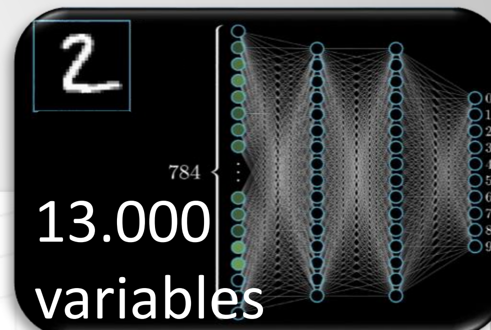


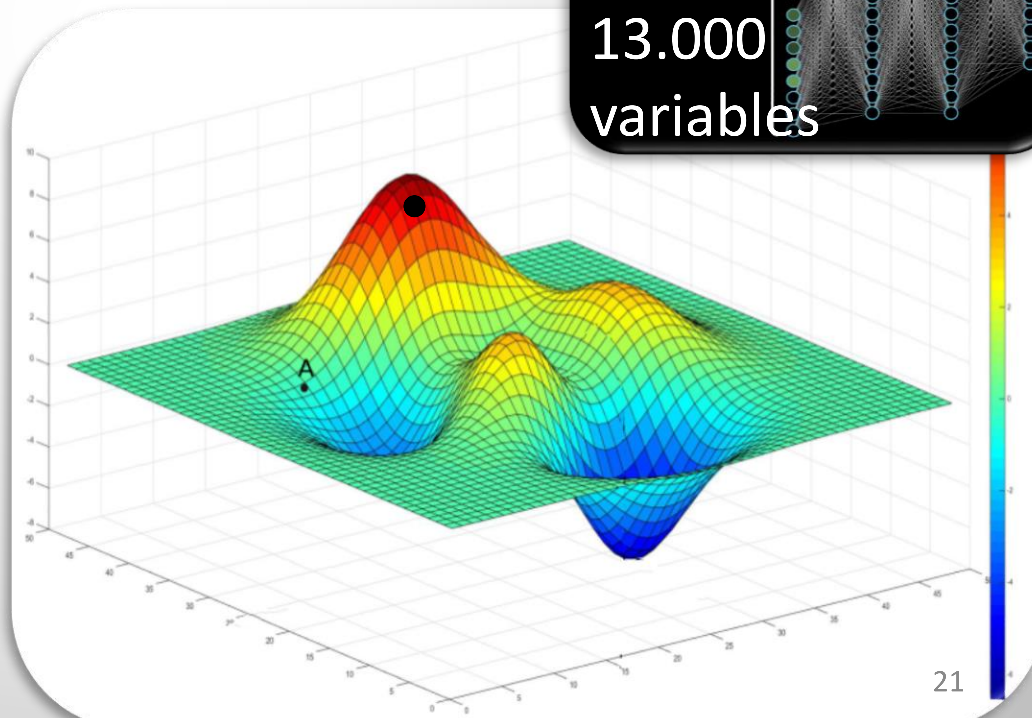


Reasoning



Model-based

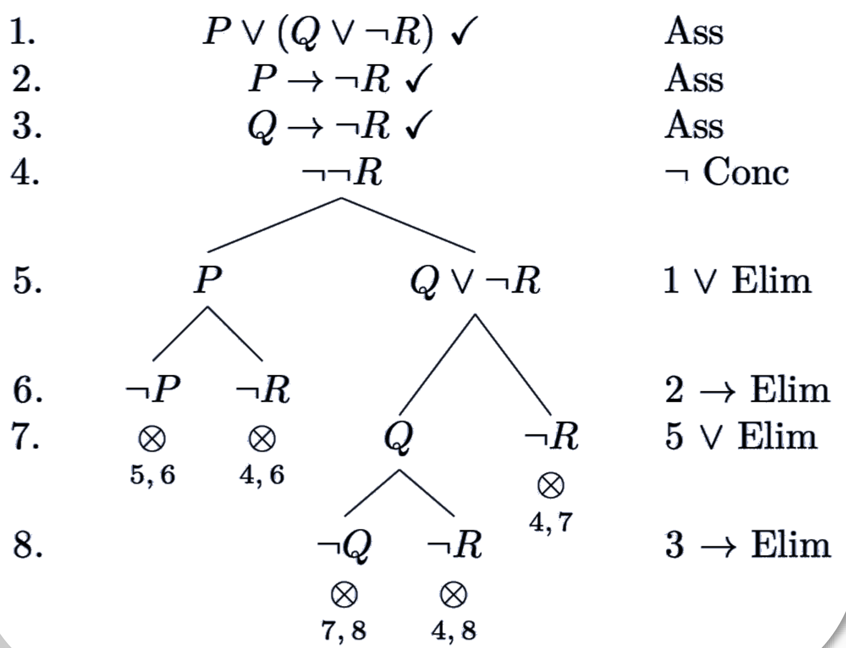




Function-based

Math

$$\{P \vee (Q \vee \neg R), P \to \neg R, Q \to \neg R\} \vdash \neg R$$



$$\widetilde{\mathcal{L}}^A(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}^{(i)}) = \frac{1}{L} \sum_{l=1}^{L} \log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}, \mathbf{z}^{(i,l)}) - \log q_{\boldsymbol{\phi}}(\mathbf{z}^{(i,l)} | \mathbf{x}^{(i)})$$

$$\text{where} \quad \mathbf{z}^{(i,l)} = g_{\boldsymbol{\phi}}(\boldsymbol{\epsilon}^{(i,l)}, \mathbf{x}^{(i)}) \quad \text{and} \quad \boldsymbol{\epsilon}^{(l)} \sim p(\boldsymbol{\epsilon})$$

Strengths & Weaknesses

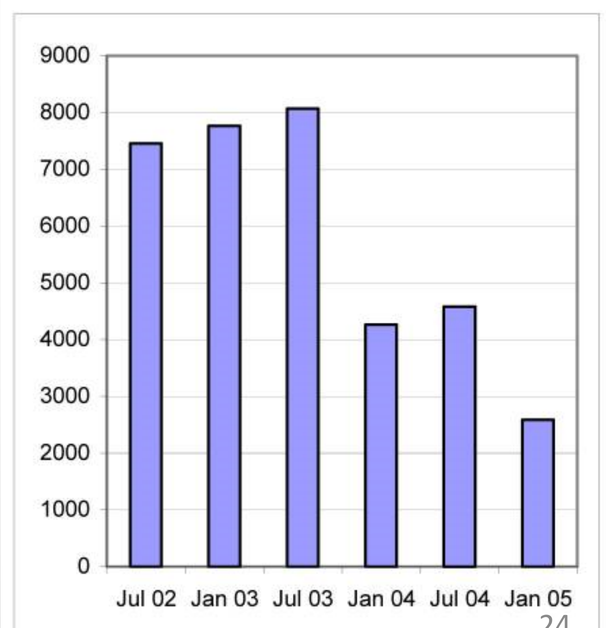
	Symbolic	Connectionist
Construction	Human effort	Data hunger
Scaleable	+/-	+/-
Explainable	+	-
Generalisable	Performance cliff	Performance cliff

Strengths & Weaknesses

	Symbolic	Connectionist
Construction	**Human effort**	Data hunger
Scaleable	+/-	+/-
Explainable	+	-
Generalisable	Performance cliff	Performance cliff

SNOMED CT
The global language of healthcare

40 years of effort,
10.000 updates every years



Strengths & Weaknesses

	Symbolic	Connectionist
Construction	Human effort	**Data hunger**
Scalable	+/-	+/-
Explainable	+	-
Generalisable	Performance cliff	Performance cliff

10M training samples

4.8M training games

Strengths & Weaknesses

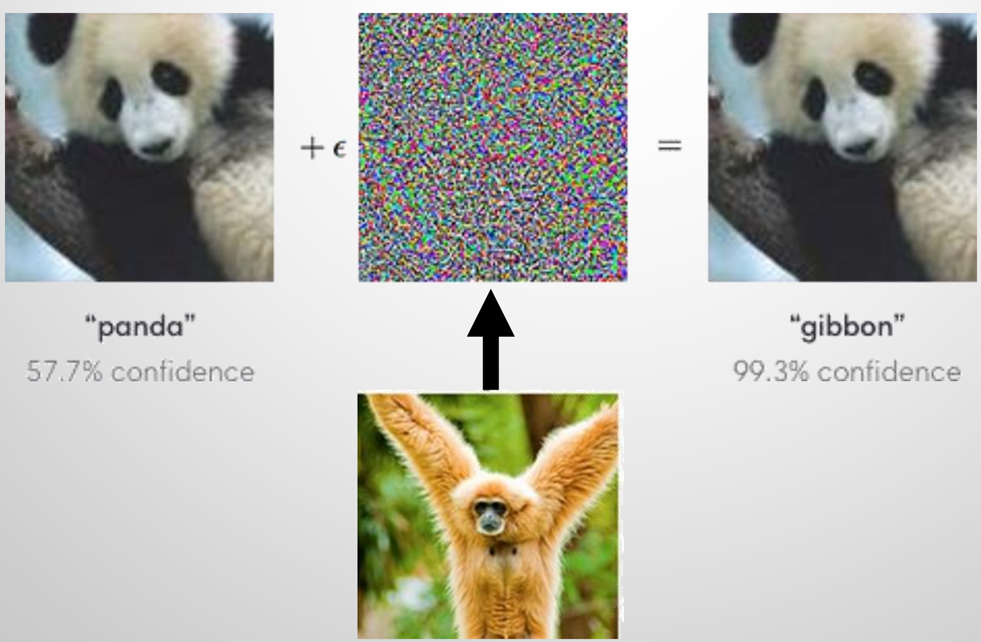
	Symbolic	**Connectionist**
Construction	Human effort	Data hunger
Scaleable	**+/-**	**+/-**
Explainable	+	-
Generalisable	Performance cliff	Performance cliff

worse with **more** data

worse with **less** data

Strengths & Weaknesses

	Symbolic	Connectionist
Construction	Human effort	Data hunger
Scaleable	+/-	+/-
Explainable	+	-
Generalisable	Performance cliff	Performance cliff

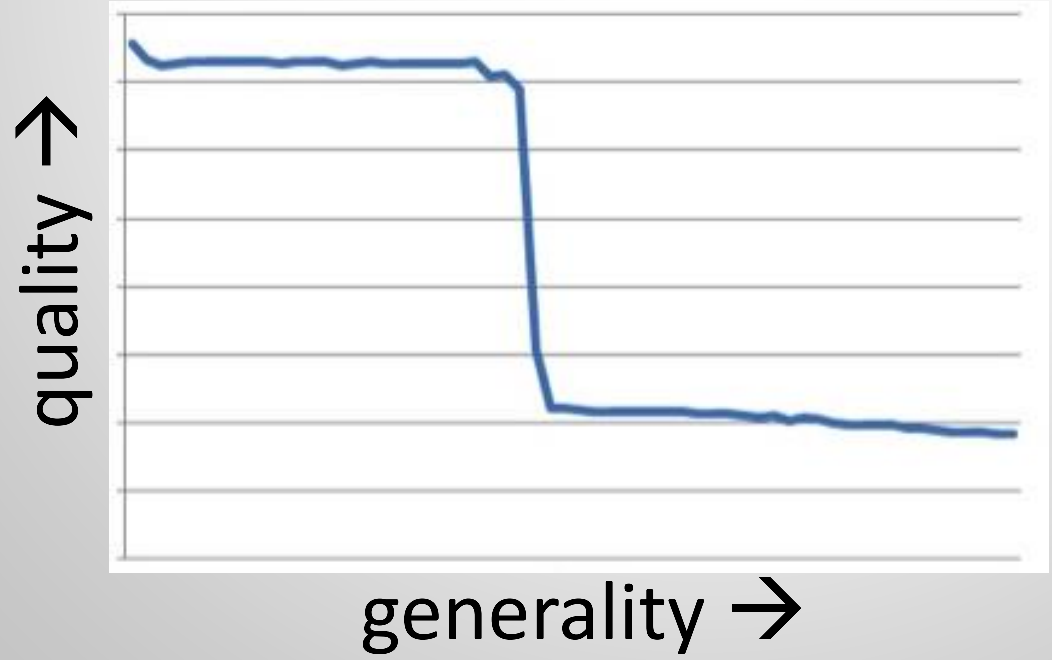


Strengths & Weaknesses

	Symbolic	Connectionist
Construction	Human effort	Data hunger
Scaleable	+/-	+/-
Explainable	+	-
Generalisable	Performance cliff	Performance cliff

Strengths & Weaknesses

	Symbolic	Connectionist
Construction	Human effort	Data hunger
Scaleable	+/-	+/-
Explainable	+	-
Generalisable	**Performance cliff**	Performance cliff



Strengths & Weaknesses

	Symbolic	Connectionist
Construction	Human effort	Data hunger
Scaleable	+/-	+/-
Explainable	+	-
Generalisable	Performance cliff	**Performance cliff**

Class: 793
Label: n04209133 **(shower cap)**
Certainty: 99.7%

Can we get them to collaborate?

Plan:

loose coupling

+

compositional patterns

$$\boxed{\text{sym}} \rightarrow (\text{KR}) \rightarrow \boxed{\text{sym}}$$

$$\boxed{\text{data}} \rightarrow (\text{ML}) \rightarrow \boxed{\text{data}}$$

Simplest unification:
Learning on symbols

- Inductive Logic Programming
- Probabilistic Soft Logic
- Markov Logic Networks

Ignored by too many of us

All of these need Knowledge Engineering

From data to symbols

data → ML → sym

- Ontology learning from text
  - Learning class/instance distictions from text
    - Knowledge graph learning from text
- Conceptual spaces

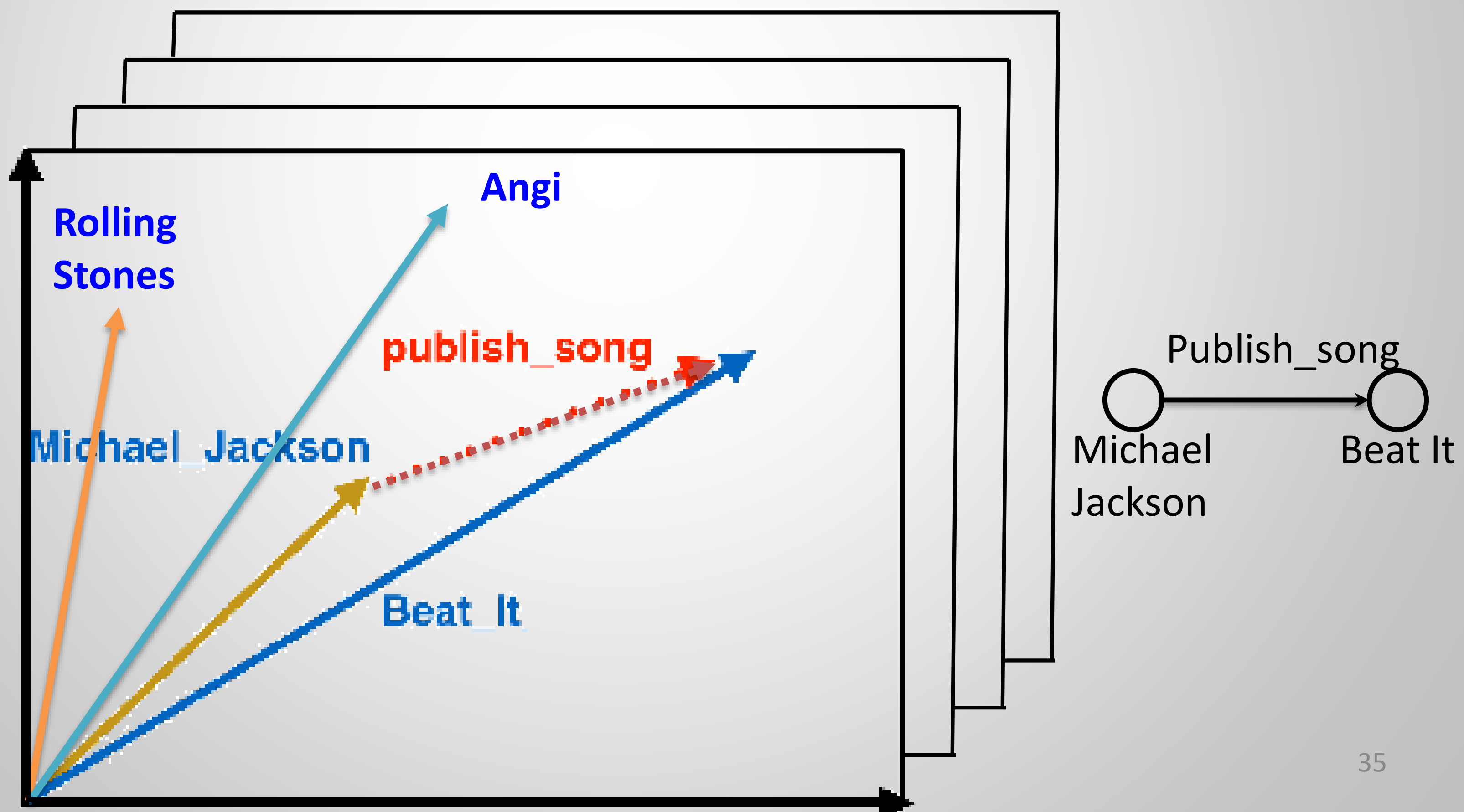
Also covers many “classical” learning algorithms:

- Decision tree learning
- Rule learning

All of these need Knowledge Engineering

From symbols to data and back again

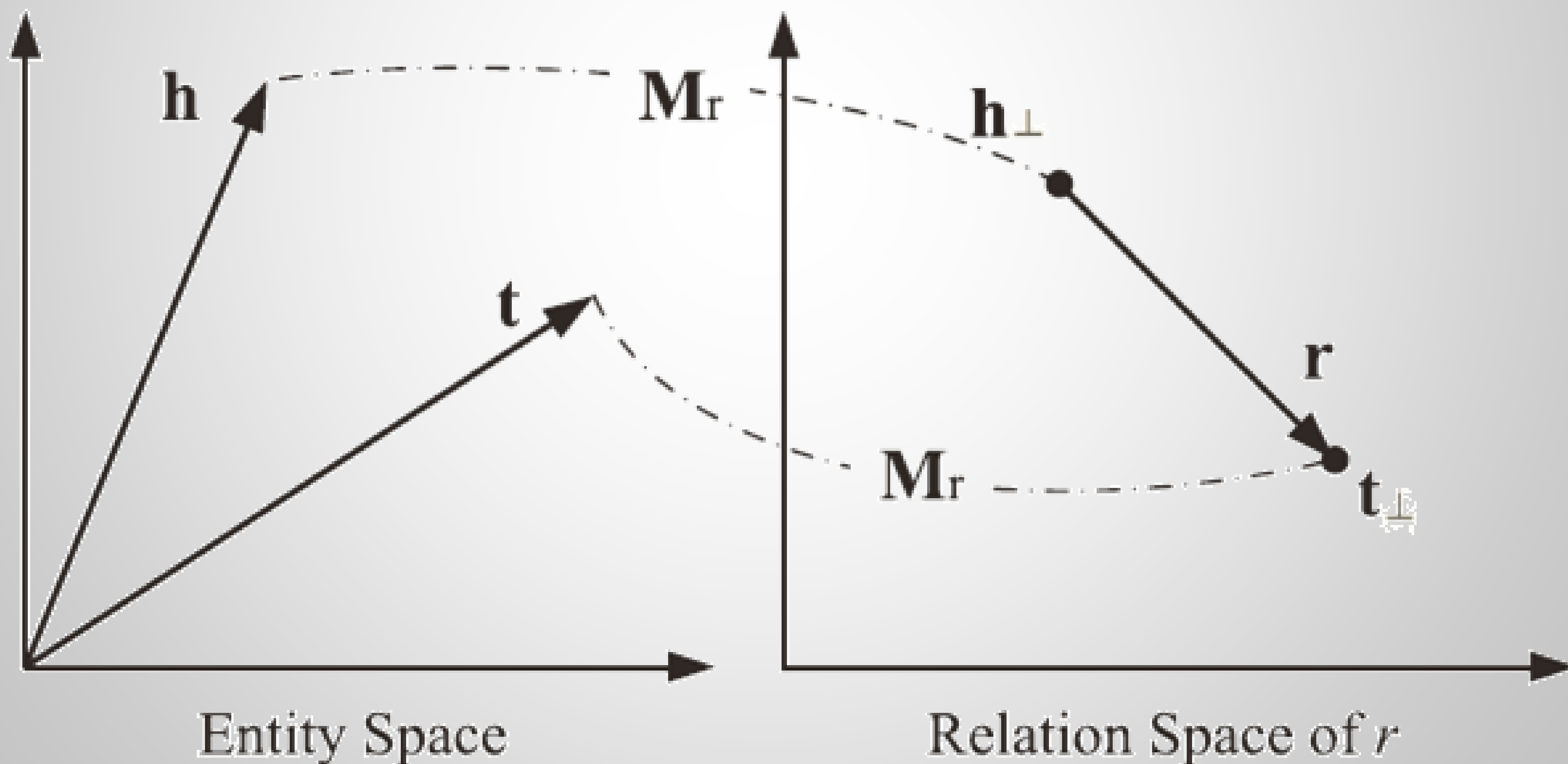
Knowledge Graph completion



From symbols to data and back again

sym → ML → data → ML → sym

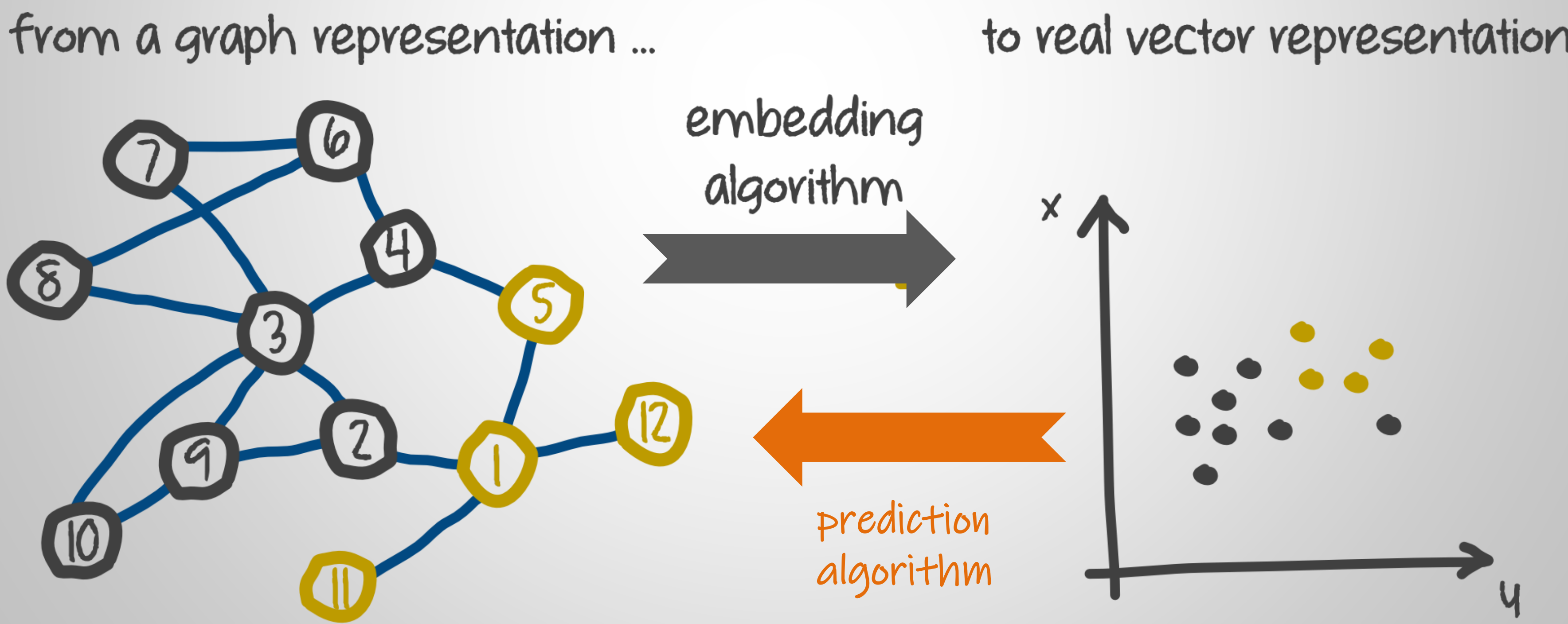
Knowledge Graph completion



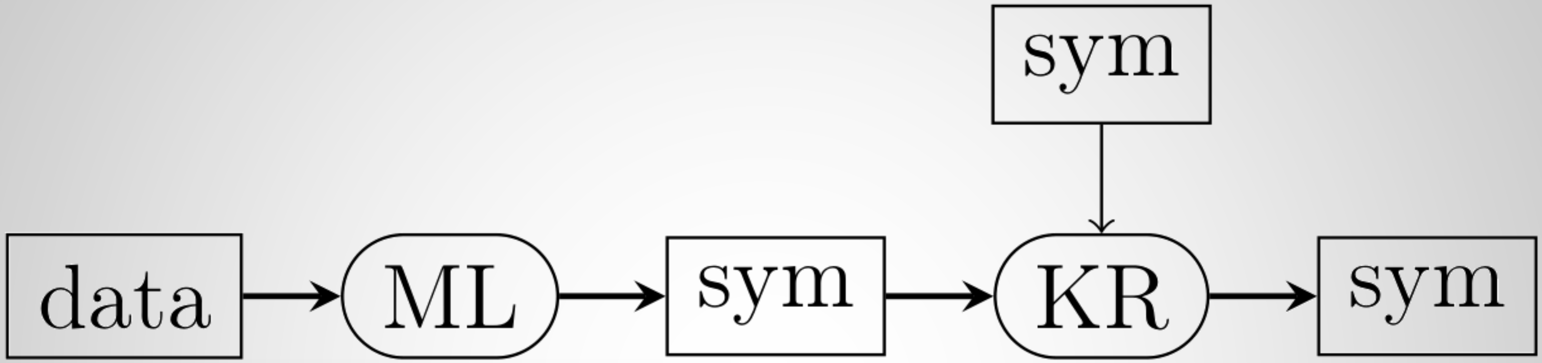
From symbols to data and back again

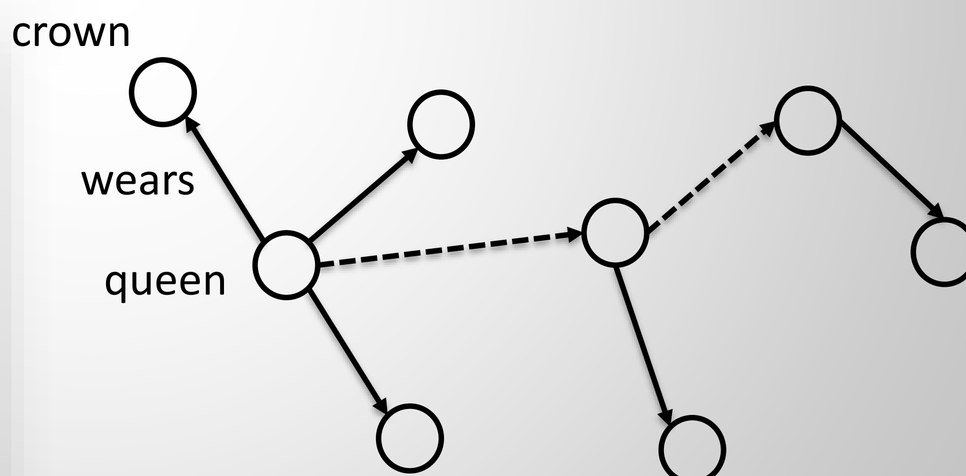
sym → ML → data → ML → sym

Knowledge Graph completion

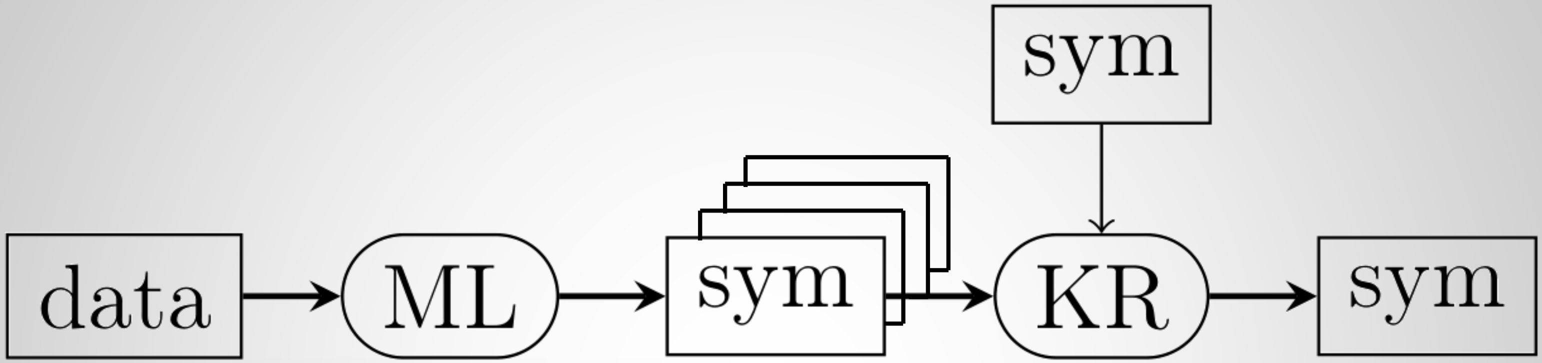


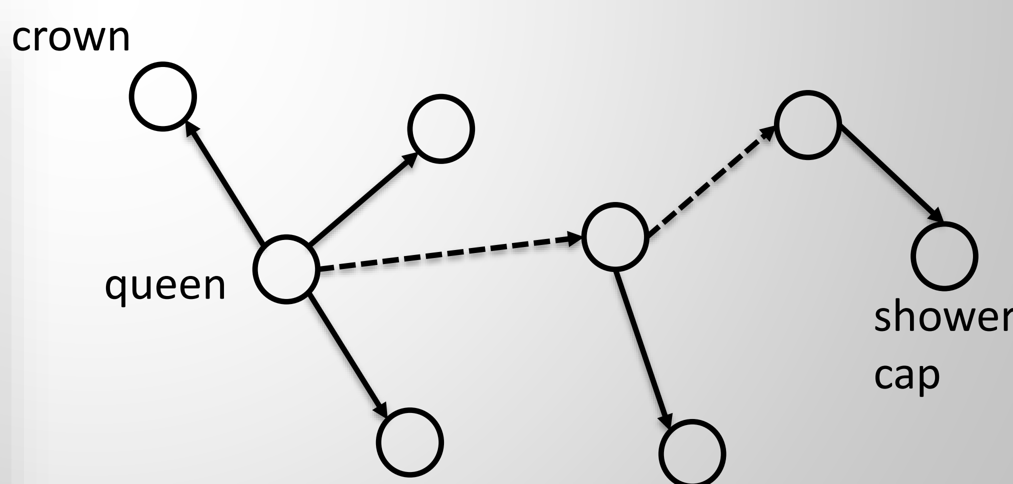
Generating an explanation



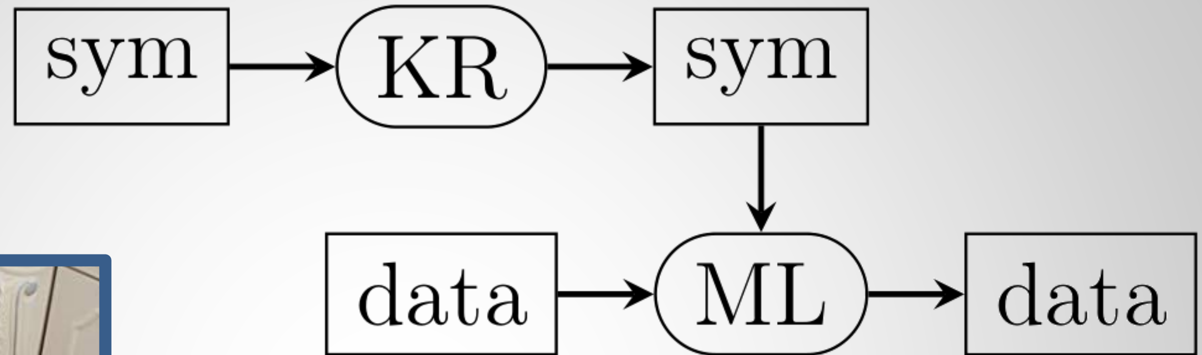


Ranking hypotheses
(≈ explaining why not)





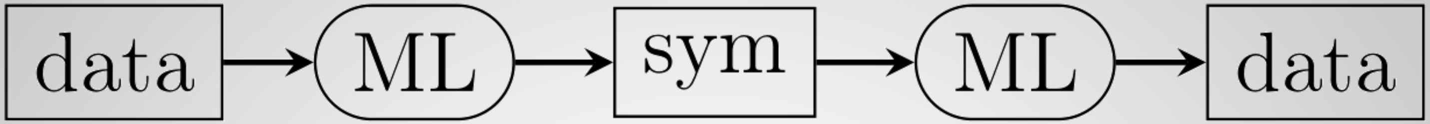
Symbolic prior



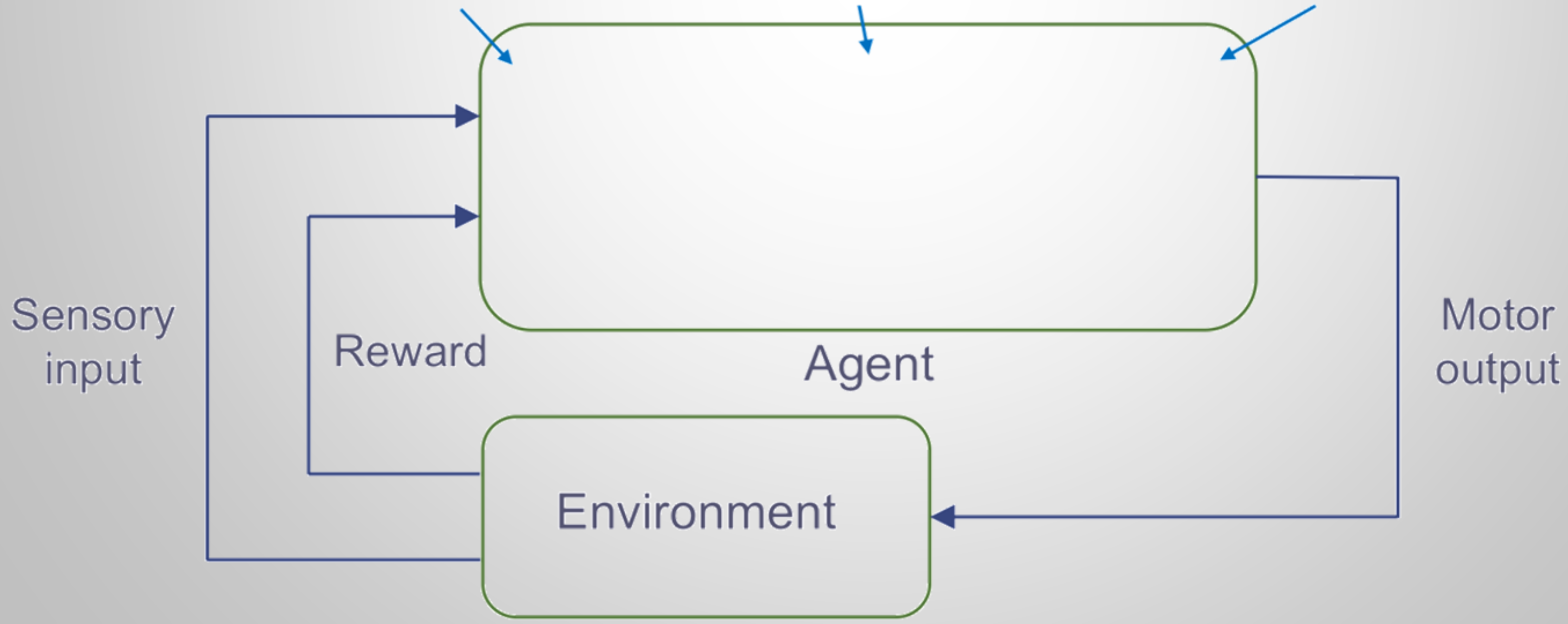
$$\forall x, y \; \mathsf{chair}(x) \land \mathsf{partOf}(y, x) \rightarrow \mathsf{cushion}(y) \lor \mathsf{armRest}(y)$$

P(cushion|chair) >> P(flower|chair)

Learning intermediate abstractions



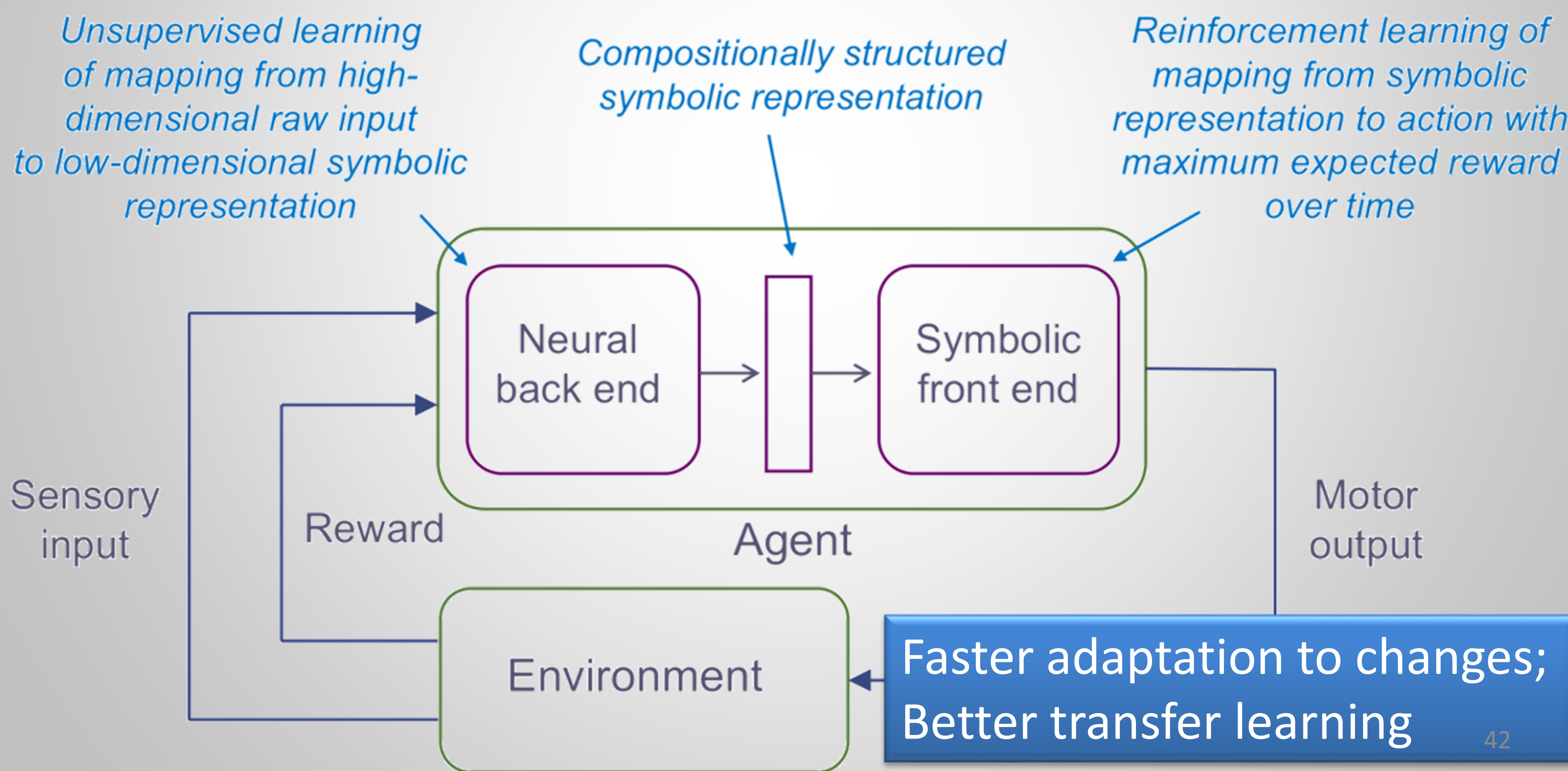
Example: Reinforcement learning for spatial navigation



Learning intermediate abstractions

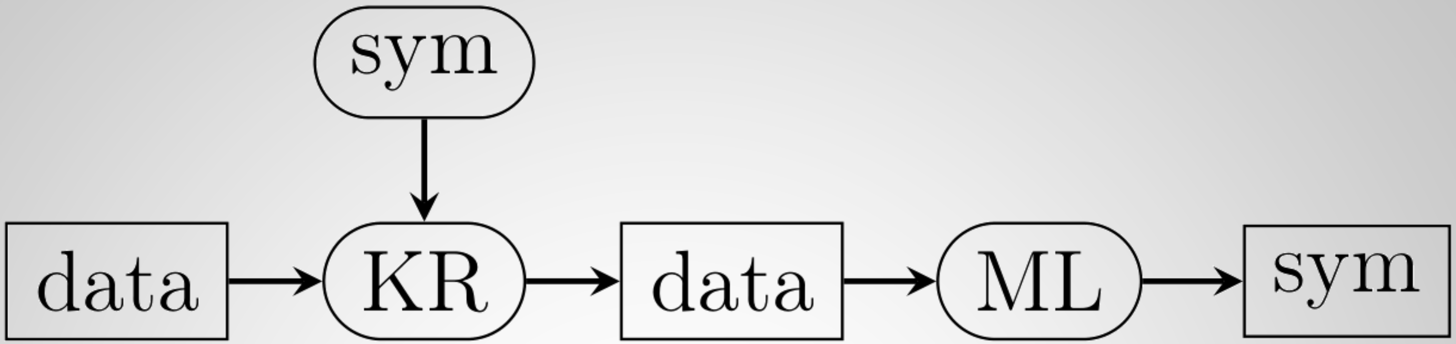
data → ML → sym → ML → data

Example: Reinforcement learning for spatial navigation



Faster adaptation to changes;
Better transfer learning

Learning intermediate abstractions

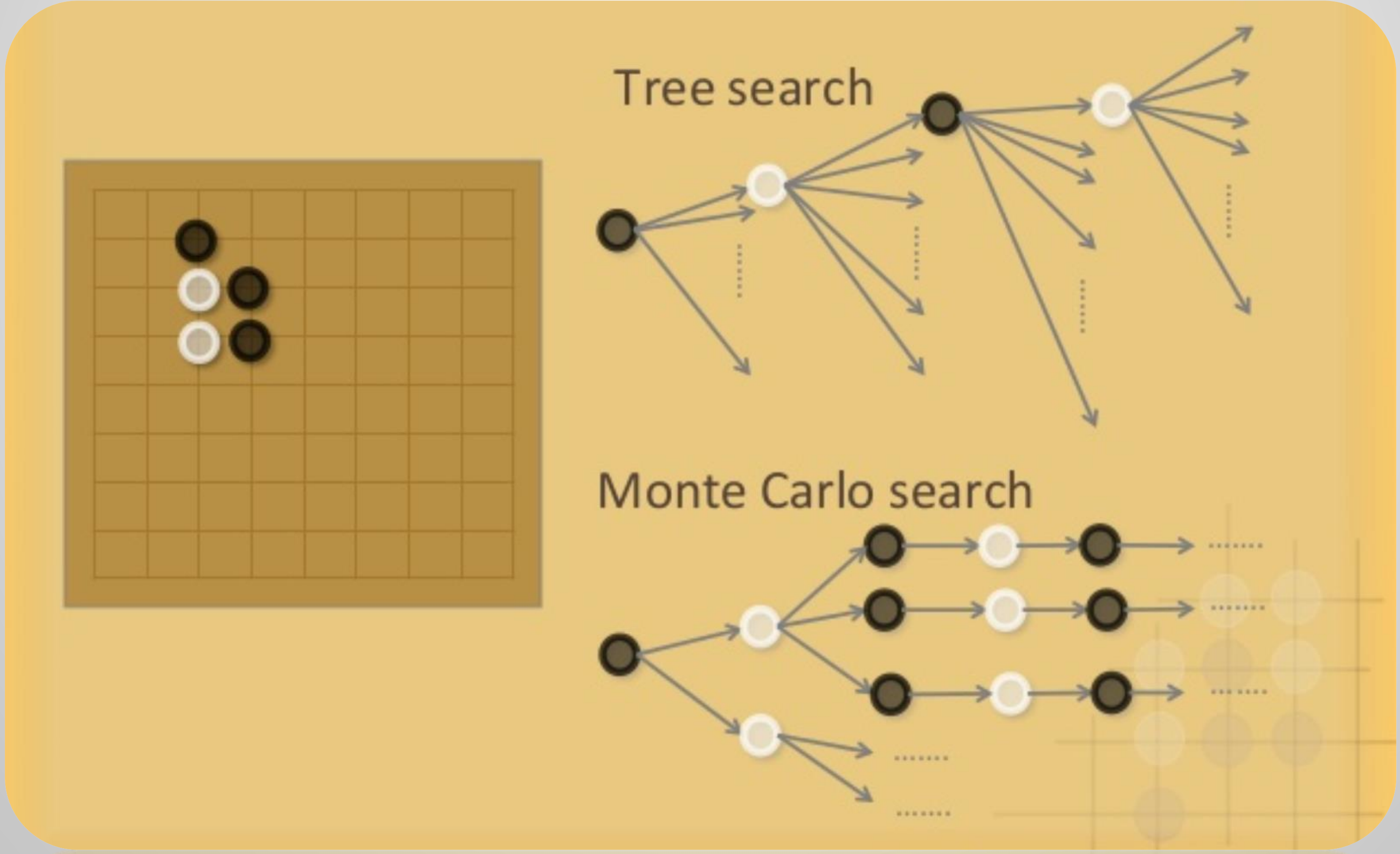


Example:

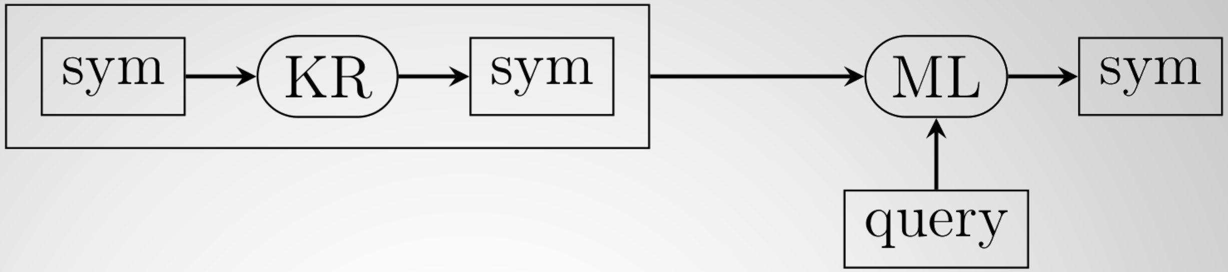
Try to learn early symptoms of CRC from GP data: (symptoms, life style, diagnoses, drugs)

1. Raw data → no meaningful signal
2. Raw data + KG = abstracted data → significant results

Learning intermediate abstractions



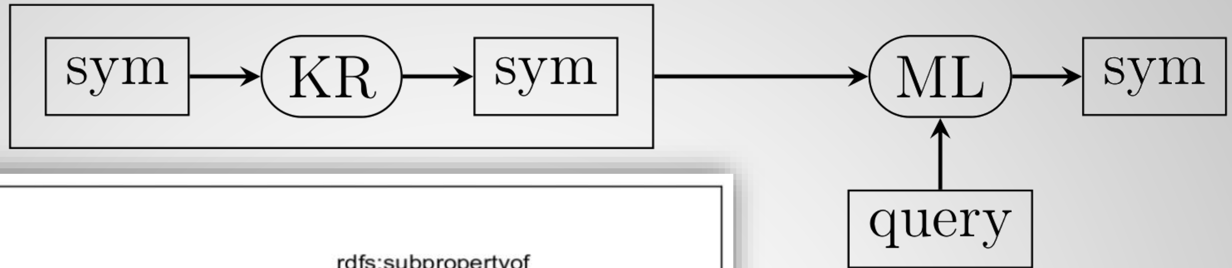
Learning ("imitating?") to reason

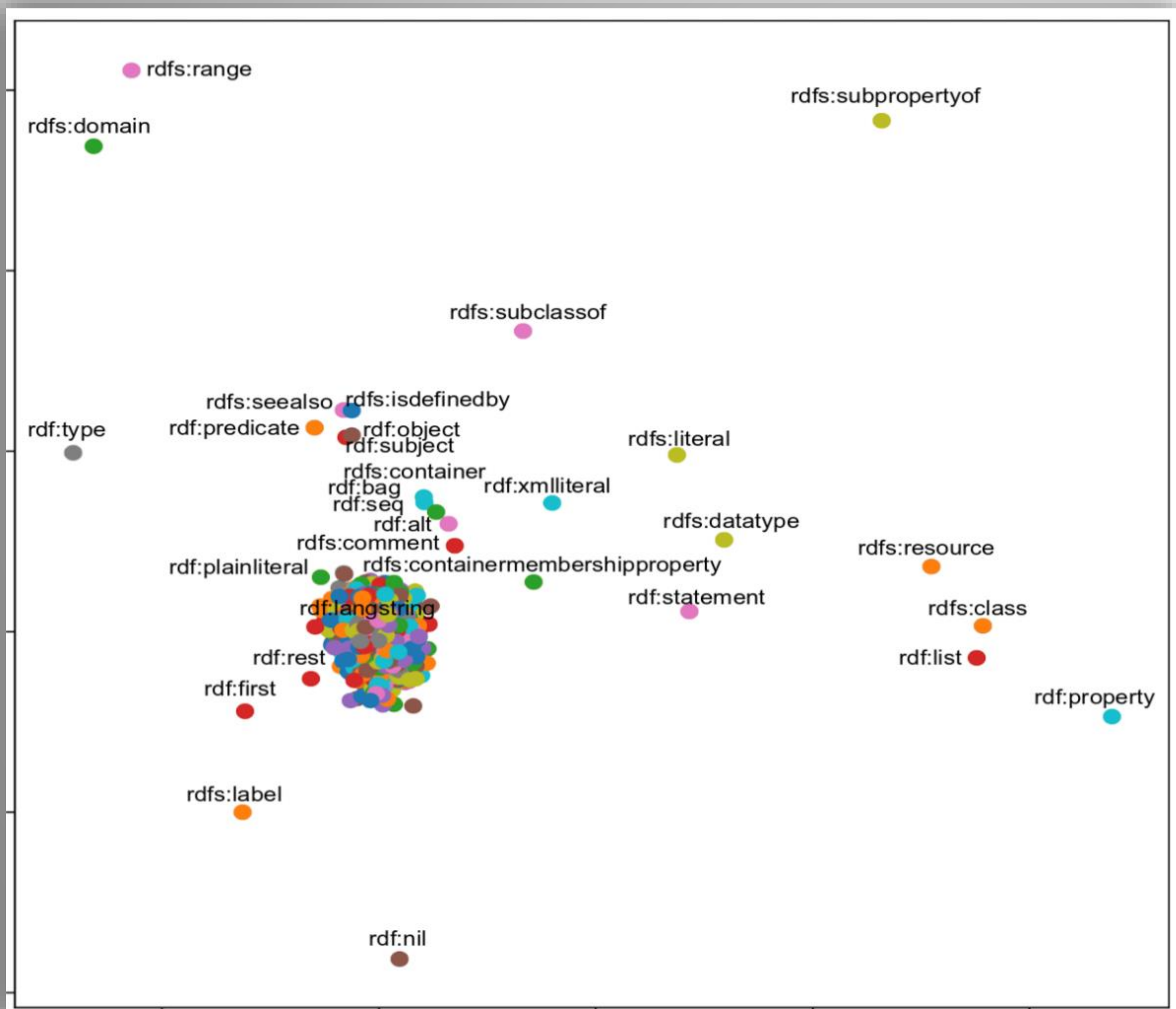


Test Dataset	Hop 0			Hop 1			Hop 2			Hop 3			Hop 4		
	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
Linked Data[a]	0	0	0	80	99	88	89	97	93	77	98	86	-	-	-
Linked Data[b]	2	0	0	82	91	86	89	98	93	79	100	88	-	-	-
OWL-Centric [c]	19	5	9	31	75	42	78	80	78	48	47	44	4	34	6
Synthetic	32	46	33	31	87	38	66	55	44	25	45	32	29	46	33

[a] LemonUby Ontology
[b] Agrovoc Ontology
[c] Completely Different Domain

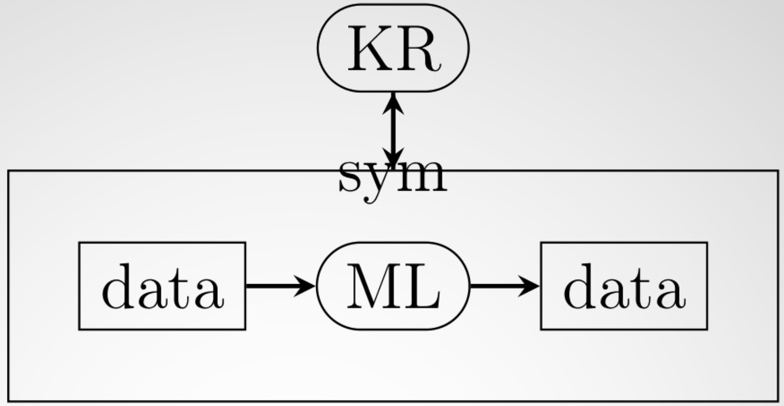
Learning ("imitating?") to reason



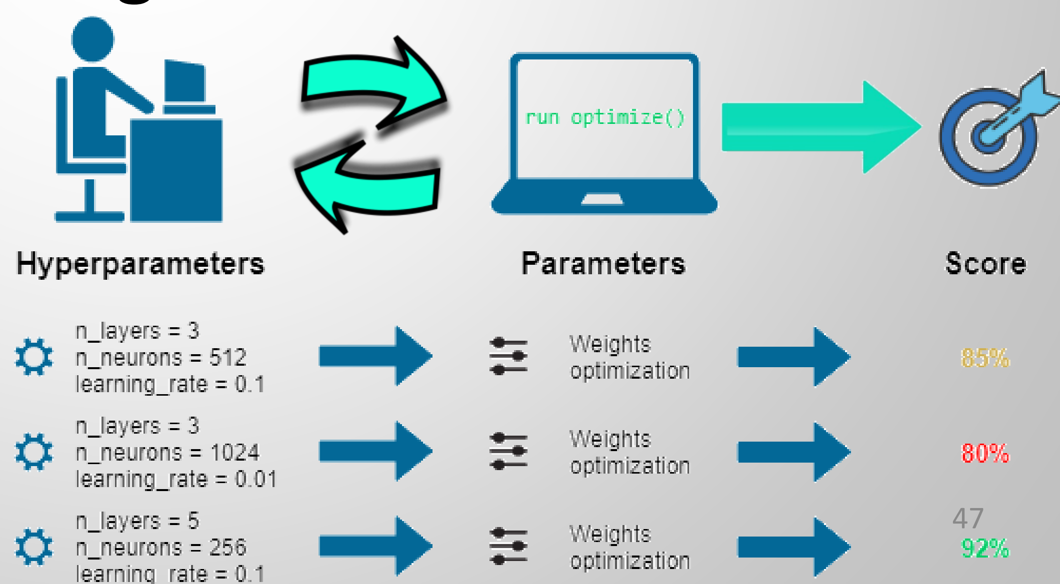


Reasoning transfers to other KG's!

Knowledge-based auto-ML



- Algorithmic configuration
- Hyperparameter tuning
- Selection of training examples



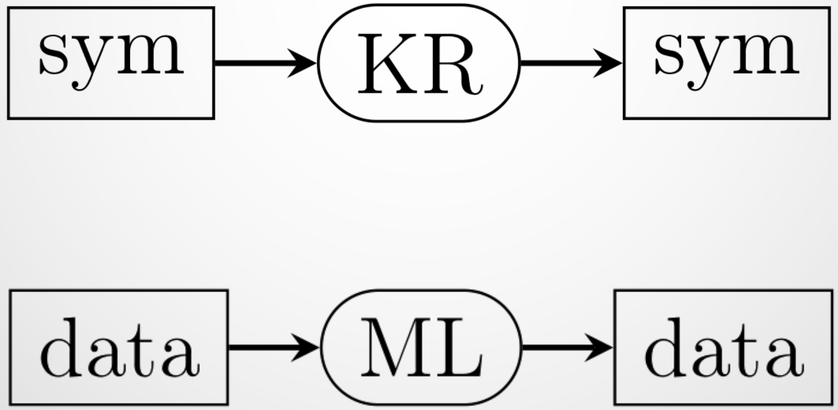
Concluding remarks

Plan:

loose coupling

\+

compositional patterns



1. **The future of AI** depends on combining

Thinking fast	and	thinking slow, or
Perception	and	cognition, or
Learning	and	reasoning

2. Lots of useful results by "**loose coupling**"

3. Develop a theory of **reusable components** and **compositional patterns**

4. In the style of **knowledge engineering** & software engineering

1. **The future of AI** depends on combining
   Thinking fast and thinking slow
   Perception and cognition
   Learning and reasoning

2. Lots of useful results by “**loose coupling**”

3. Develop a science of
   reusable components and
   compositional patterns

4. In the style of
   knowledge engineering
   software engineering
   “X” engineering