Understanding Neural Networks Through Background Knowledge

Pascal Hitzler
Data Semantics Laboratory (DaSe Lab)
Data Science and Security Cluster (DSSC)
Wright State University
http://www.pascal-hitzler.de

Semantic Web

- **A research field about methods for:**

Data and Information sharing, discovery, integration, and reuse.

Key paradigms:

- **Representation of information via knowledge graphs in standardized formats (e.g., W3C’s RDF).**
- **Typing of the knowledge graphs together with a type logic a.k.a. ontology or schema, represented in standardized/sharable formats (e.g., W3C’s OWL)**

Semantic Web

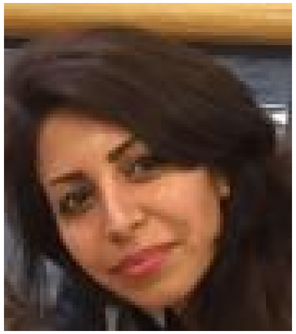
Two major examples of semantic web technologies at work:

- **Google knowledge graph**
  You see a glimpse of it in the boxes to the right of your search results.

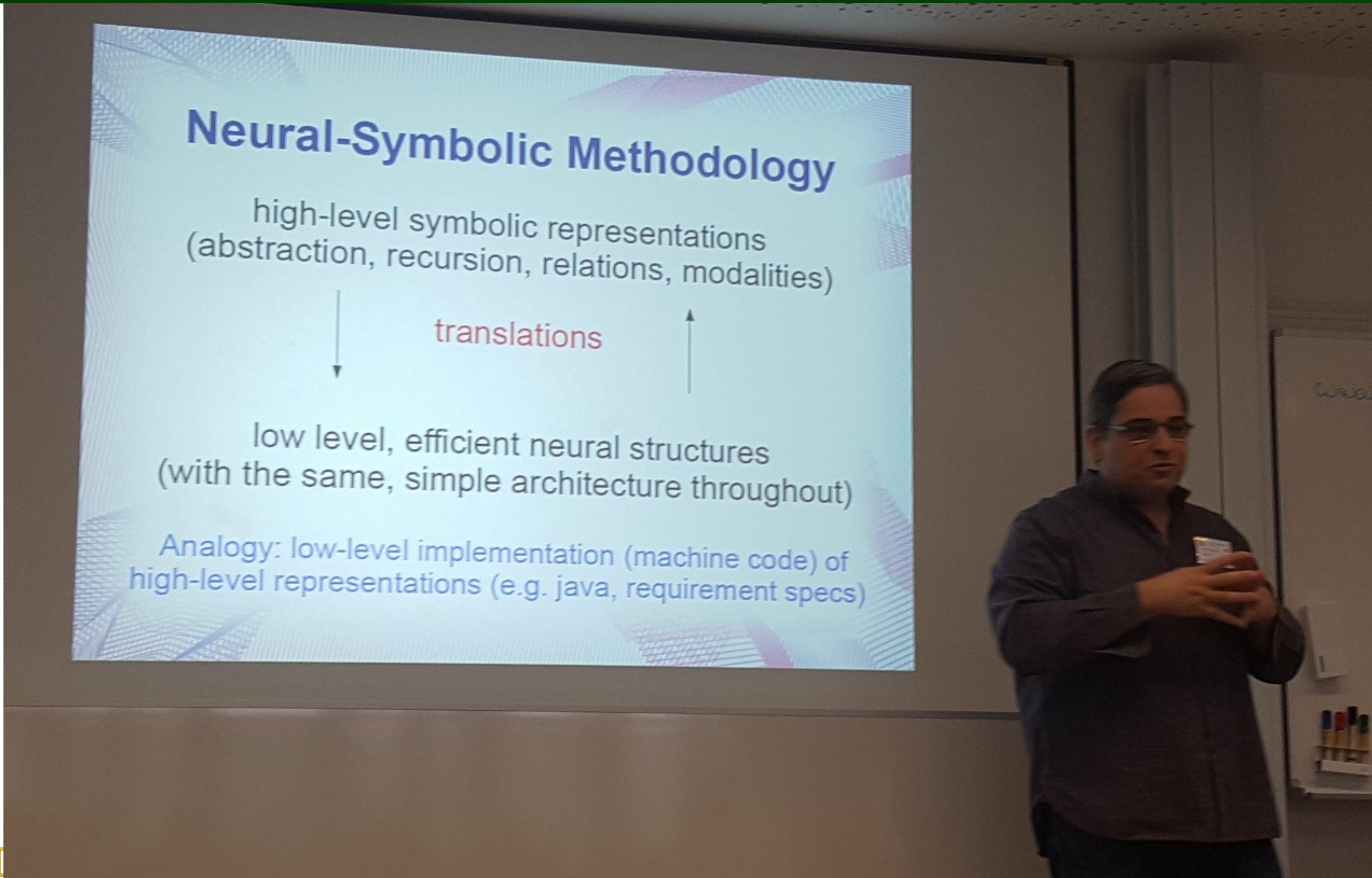
- **Schema.org**
  Joint effort by major search engine providers.
  Schema/ontology for annotating Web page content, so that search engines can provide better results.
  In the meantime, schema.org annotations are ubiquitous on the Web.

Propositional rule extraction from trained neural networks under background knowledge

(work with Maryam Labaf)



Neural and symbolic



Extraction

In this case: extracting propositional rules.

General idea:

- **Input value 1 interpreted as “true”, value 0 as “false”**
- **Outputs interpreted as true or false according to a threshold**
- **I.e. network function maps binary vectors.**

Garcez et al, 2001: By weight analysis (layer by layer) under differentiable activation functions. Possible in principle but intricate and, arguably, the resulting rule sets are usually rather difficult to understand.

Lehmann, Bader, Hitzler, 2010: Black-box approach (looking at inputs and outputs only).

Lehmann, Bader, Hitzler, 2010

For every monotonic function

$$f : \{0,1\}^n \rightarrow \{0,1\}^k$$

there is a unique reduced set of positive propositional rules which capture exactly the function f.

Reduced means: no redundancies, and as small as possible.

Problem: Rule sets can get large and messy, i.e. still very difficult to understand.

Adding Background Knowledge

Can we lift the result just given to include background knowledge?

Given:

- **A (reduced) propositional logic program P (extracted from an ANN as above).**
- **Set I of prop. variables representing ANN inputs.**
- **Set O of prop. variables representing ANN outputs.**
- **A background knowledge base K (a propositional logic program).**

We then seek a logic program P’ (simpler than P)
s.t. for all subsets i in I and each o in O we have

$$P \wedge i \models o \qquad \text{iff} \qquad P' \wedge K \wedge i \models o.$$

Adding Background Knowledge

It turns out that

- P' is no longer unique in general (even under reduction).
- P' may not even exist (unless I is restricted to the left-hand side of rules in K).

- But with suitable K you can get P' which are simpler than P. Typical case:

P: $p_1 \wedge q \rightarrow o$ K: $p_1 \rightarrow r$ P': $r \wedge q \rightarrow o$

$p_2 \wedge q \rightarrow o$ $p_2 \rightarrow r$

Adding Background Knowledge

P: $p_1 \wedge q \to o$

$p_2 \wedge q \to o$

K: $p_1 \to r$

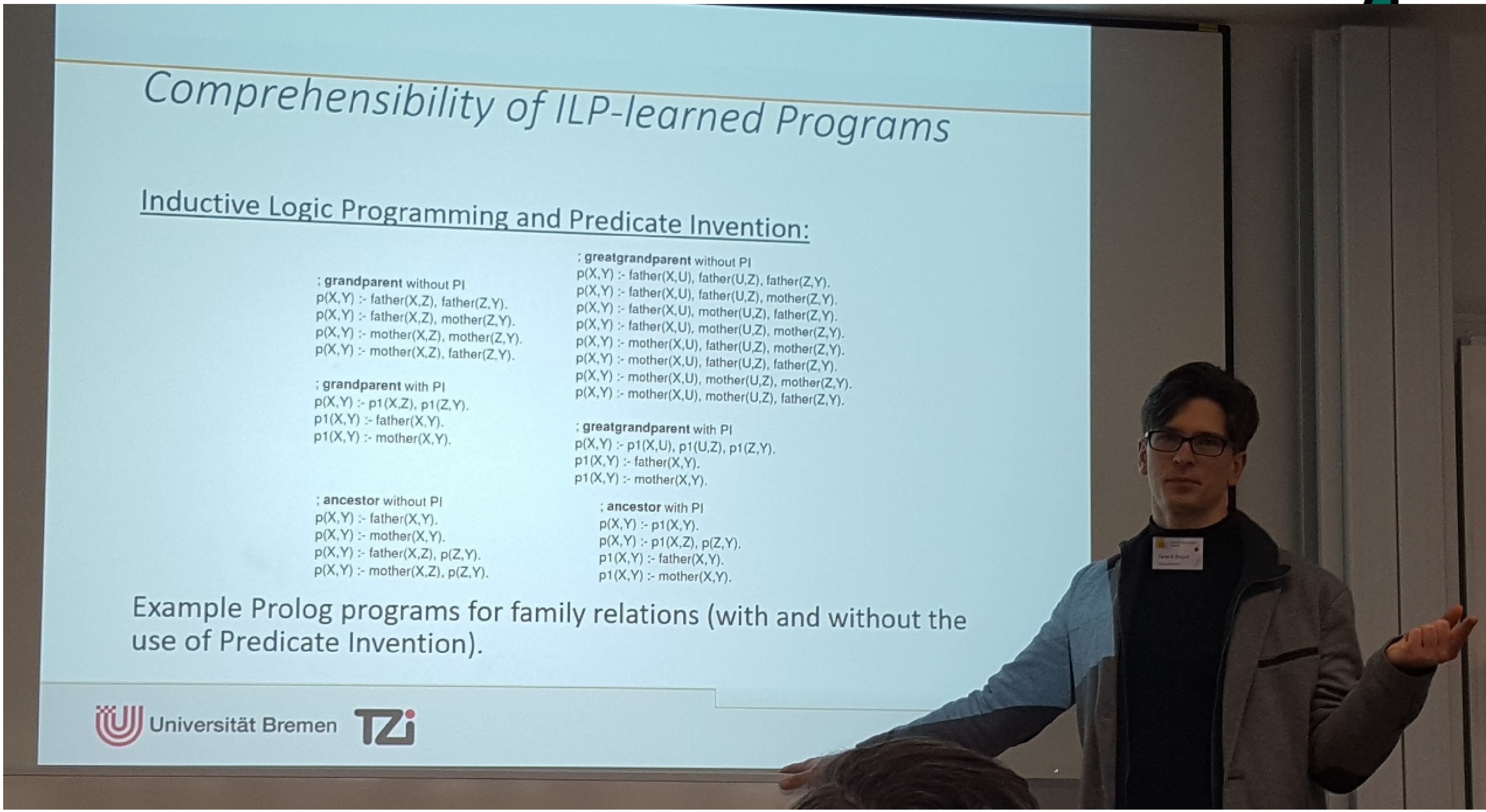
$p_2 \to r$

P': $r \wedge q \to o$

Note that K essentially groups input variables. Once could think of r being a "more general concept" than either p1 and p2.

Of course, we have only discussed the propositional case so far, but in order to obtain strong explanations for the input-output behavior of ANNs we need to go beyond propositional.

Comprehensibility!



Description Logic extraction from trained neural networks under background knowledge

(work with Md Kamruzzaman Sarker, Derek Doran, Ning Xie, Mike Raymer)




DL Extraction from ANNs

- Explain input-output behavior of trained (deep) NNs.

- Idea:
  - Use background knowledge in the form of linked data and ontologies to help explain.
  - Link inputs and outputs to background knowledge.
  - Use a symbolic learning system (e.g., DL-Learner) to generate an explanatory theory.

- We're just starting on this, experiments (below) just came out.

DL Extraction from ANNs

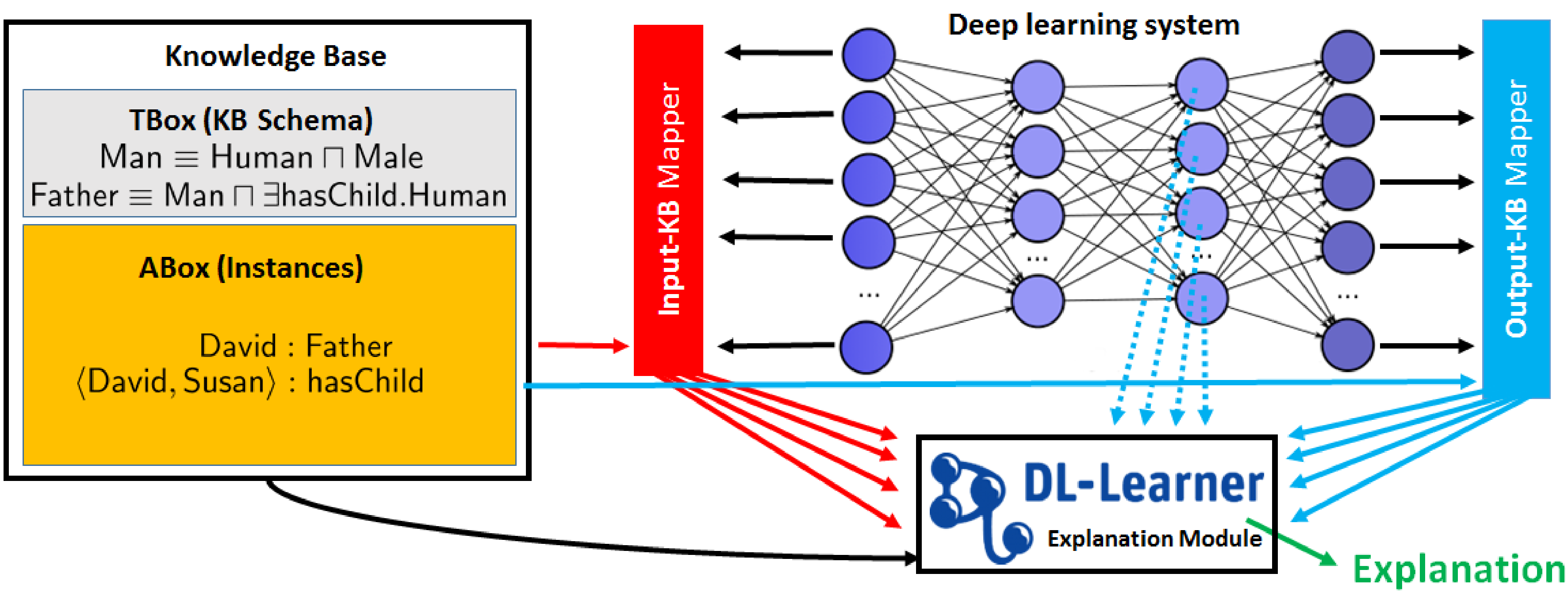
Possible data sources:

- **Linked data / semantic web data**
  - **I.e. structured data on the web, organized in so-called RDF graphs.**
- **Cross-domain ontologies (e.g., SUMO, Proton)**
- **Wikidata**
- **schema.org**

Essentially, all content already readily and publicly available in structured form.

If further domain knowledge is needed: use state-of-the-art approaches for knowledge graph generation in order to obtain structured data from suitable text corpora.

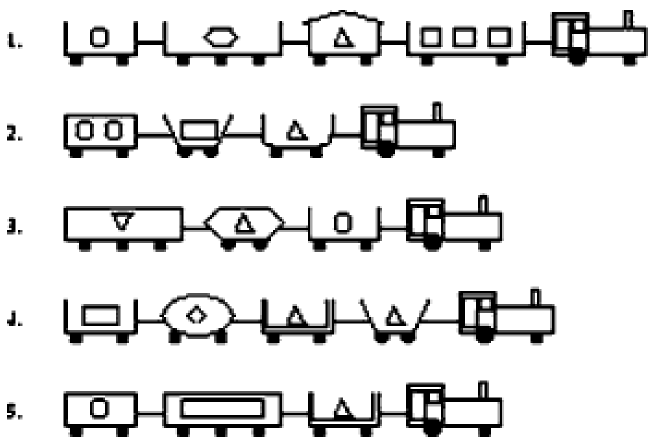
DL Extraction from ANNs

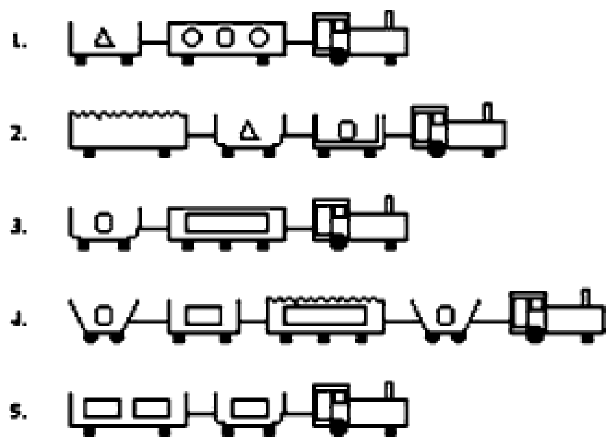


DL-Learner

Approach similar to inductive logic programming, but using Description Logics (the logic underlying OWL).

Positive examples: negative examples:

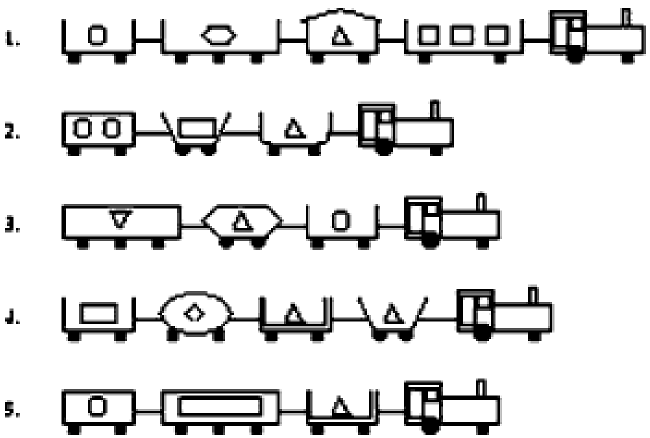




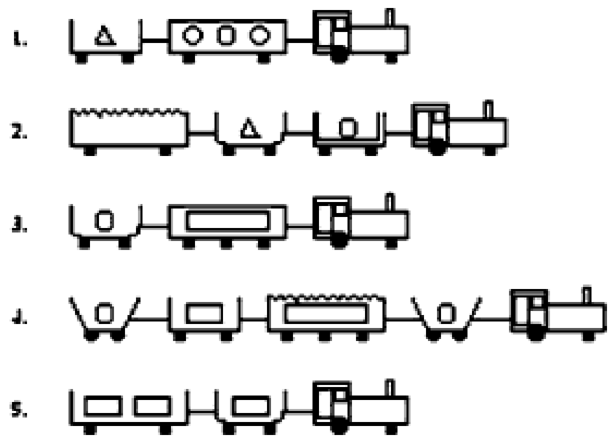
Task: find a class description (logical formula) which separates positive and negative examples.

DL-Learner

Positive examples:



negative examples:



DL-Learner result: $\exists \mathtt{hasCar}.(\mathtt{Closed} \sqcap \mathtt{Short})$

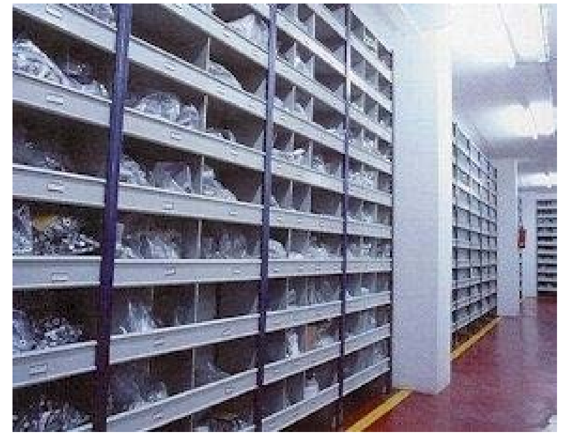
In FOL:

$$\{x \mid \exists y(\text{hasCar}(x,y) \land \text{Closed}(y) \land \text{Short}(y))\}$$

Proof of Concept Experiment

Positive:

Negative:



Images

Come from the MIT ADE20k dataset
http://groups.csail.mit.edu/vision/datasets/ADE20K/
They come with annotations of objects in the picture:

```
001 # 0 # 0 # sky # sky # ""
002 # 0 # 0 # road, route # road # ""
005 # 0 # 0 # sidewalk, pavement # sidewalk # ""
006 # 0 # 0 # building, edifice # building # ""
007 # 0 # 0 # truck, motortruck # truck # ""
008 # 0 # 0 # hovel, hut, hutch, shack, shanty # hut # ""
009 # 0 # 0 # pallet # pallet # ""
011 # 0 # 0 # box # boxes # ""
001 # 1 # 0 # door # door # ""
002 # 1 # 0 # window # window # ""
009 # 1 # 0 # wheel # wheel # ""
```

Mapping to SUMO

Simple approach: for each known object in image, create an individual for the ontology which is in the appropriate SUMO class:

contains road1

contains window1

contains door1

contains wheel1

contains sidewalk1

contains truck1

contains box1

contains building1

DL-Learner input

Positive:

- img1: road, window, door, wheel, sidewalk, truck, box, building
- img2: tree, road, window, timber, building, lumber
- img3: hand, sidewalk, clock, steps, door, face, building, window, road

Negative:

- img4: shelf, ceiling, floor
- img5: box, floor, wall, ceiling, product
- img6: ceiling, wall, shelf, floor, product

DL-Learner results include:

$$\exists \text{contains}.\text{Transitway}$$

$$\exists \text{contains}.\text{LandArea}$$

Proof of Concept Experiment

Positive:

Negative:

$$\exists \mathrm{contains}.\mathrm{Transitway}$$

$$\exists \mathrm{contains}.\mathrm{LandArea}$$

Experiment 2

Positive (selection):

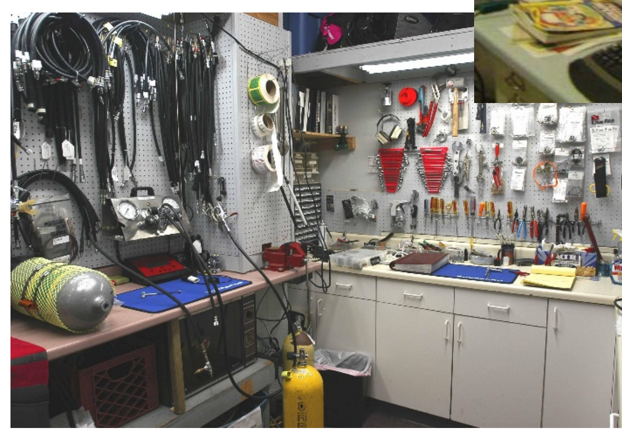
Negative (selection):

$$\exists \text{contains}.(\text{DurableGood} \sqcap \neg \text{ForestProduct})$$

Experiment 3

Positive:

Negative:



$$\forall \text{contains}.(\neg \text{Furniture} \sqcap \neg \text{IndustrialSupply})$$

Experiment 4

Positive (selection):

Negative (selection):

$\exists \text{contains.SentientAgent}$

Experiment 5

Positive:

Negative (selection):

$\exists$contains.BodyOfWater

Next steps

- Utilize more sophisticated ontology.
- Utilize more sophisticated mappings.

- Explain hidden neurons.

- Tune DL-Learner better to the specific task.

Explaining hidden neurons

Collaborators Derek Doran and Ning Xie (Web and Complex Systems Lab)

They explore how to determine groups of hidden neurons which often fire together and thus may indicate the "detection" of certain features.

We plan to apply the above mentioned DL-Learner approach also to these groups of hidden neurons, in order to determine which features they detect.

Thanks!

References

- Md. Kamruzzaman Sarker, David Carral, Adila A. Krisnadhi, Pascal Hitzler, Modeling OWL with Rules: The ROWL Protege Plugin. In: Takahiro Kawamura, Heiko Paulheim (eds.), Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016. CEUR Workshop Proceedings 1690, CEUR-WS.org 2016.
- Md Kamruzzaman Sarker, Adila A. Krisnadhi, David Carral, Pascal Hitzler, Rule-based OWL Modeling with ROWLTab Protege Plugin. In: Proceedings ESWC 2017. To appear.
- Hitzler, Krötzsch, Rudolph, Foundations of Semantic Web Technologies, CRC/Chapman & Hall, 2010
- Adila Krisnadhi, Ontology Pattern-Based Data Integration. Dissertation, Department of Computer Science and Engineering, Wright State University, 2015.

References

- Pascal Hitzler, Adila Krisnadhi, On the Roles of Logical Axiomatizations for Ontologies. In: Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web. IOS Press/AKA Verlag, 2016/2017.
- Md. Kamruzzaman Sarker, Adila A. Krisnadhi, Pascal Hitzler, OWLAx: A Protege Plugin to Support Ontology Axiomatization through Diagramming In: Takahiro Kawamura, Heiko Paulheim (eds.), Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016. CEUR Workshop Proceedings 1690, CEUR-WS.org 2016.
- Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web. IOS Press/AKA Verlag, 2016.

References

- P. Hitzler, S. Hölldobler and A. K. Seda. Logic Programs and Connectionist Networks. Journal of Applied Logic, 2(3), 2004, 245-272.
- S. Bader and P. Hitzler, Dimensions of neural-symbolic integration – a structured survey. In: S. Artemov et al. (eds). We Will Show Them: Essays in Honour of Dov Gabbay, Volume 1. College Publications, London, 2005, pp. 167-194.
- J. Lehmann, S. Bader and P. Hitzler, Extracting reduced logic programs from artificial neural networks, In: Proceedings of the IJCAI-05 Workshop on Neural-Symbolic Learning and Reasoning, NeSy'05, Edinburgh, UK, August 2005.
- S. Bader, P. Hitzler, and S. Hölldobler, The Integration of Connectionism and First-Order Knowledge Representation and Reasoning as a Challenge for Artificial Intelligence, Journal of Information 9 (1), 2006. Invited paper.

References

- B. Hammer, P. Hitzler (eds.). Perspectives of Neural-Symbolic Integration. *Studies in Computational Intelligence*, Vol. 77. Springer, 2007, ISBN 978-3-540-73952-1.
- S. Bader, P. Hitzler, S. Hölldobler. Connectionist Model Generation: A First-Order Approach. Neurocomputing 71, 2008, 2420-2432.
- Artur d'Avila Garcez, Tarek R. Besold, Luc de Raedt, Peter Földiak, Pascal Hitzler, Thomas Icard, Kai-Uwe Kühnberger, Luis C. Lamb, Risto Miikkulainen, Daniel L. Silver, Neural-Symbolic Learning and Reasoning: Contributions and Challenges. In: Andrew McCallum, Evgeniy Gabrilovich, Ramanathan Guha, Kevin Murphy (eds.), Proceedings of the AAAI 2015 Spring Symposium on Knowledge Representation and Reasoning: Integrating Symbolic and Neural Approaches. Technical Rport SS-15-03, AAAI Press, Palo Alto, 2015.

References

- Jens Lehmann, Pascal Hitzler, Concept Learning in Description Logics Using Refinement Operators. Machine Learning 78 (1-2), 203-250, 2010.
- Cogan Shimizu, Pascal Hitzler, Matthew Horridge, Rendering OWL in Description Logic Syntax. In: ESWC 2017 poster and demo proceedings.
- Adila Krisnadhi, Pascal Hitzler, Modeling With Ontology Design Patterns: Chess Games As a Worked Example. In: Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnadhi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web Vol. 25, IOS Press/AKA Verlagpp. 3-22.