Explaining hidden neuron activations using Semantic Web methods

Pascal Hitzler
Data Semantics Laboratory (DaSe Lab)
Kansas State University

http://www.daselab.org

Neuro-symbolic AI

Publications on neuro-symbolic AI in major conferences (research papers only):

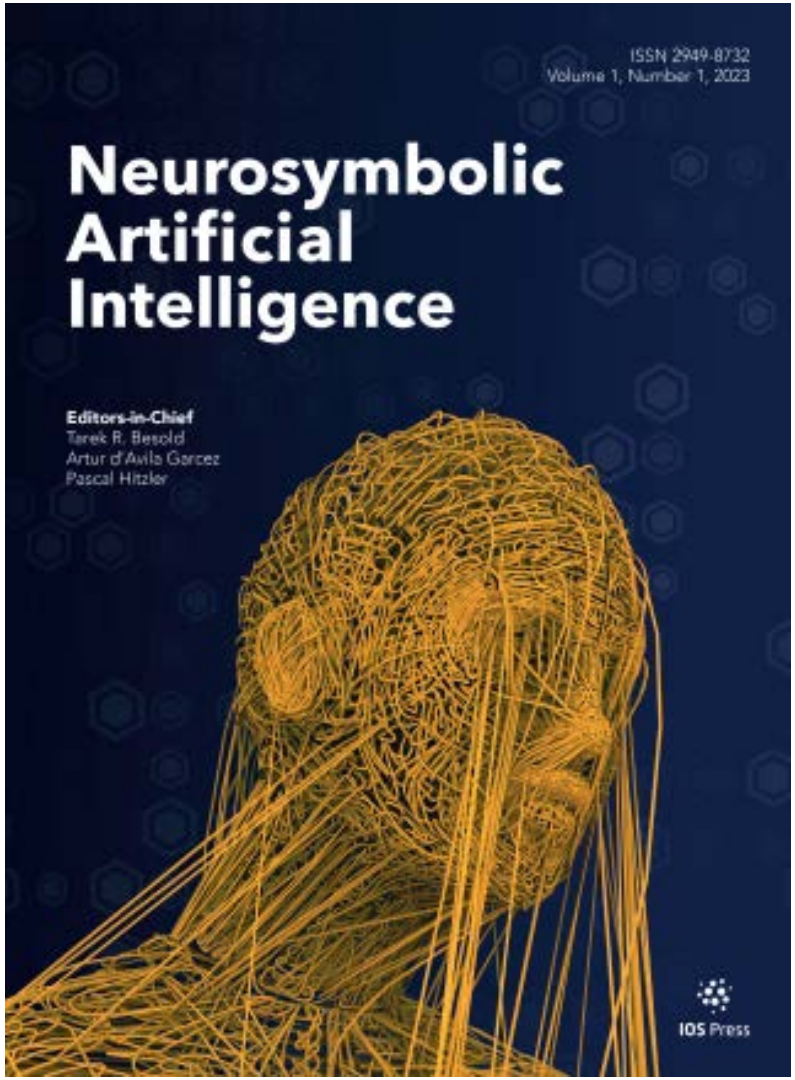
conference	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	total
ICML	0	0	0	0	0	1	3	2	5	6	17
NeurIPS	0	0	0	0	0	0	0	4	2	4	10
AAAI	0	0	0	0	0	1	0	1	1	1	4
IJCAI	1	0	0	0	0	0	2	2	0	2	7
ICLR	N/A	N/A	0	0	0	0	1	1	1	3	6
total	1	0	0	0	0	2	6	10	9	16	44

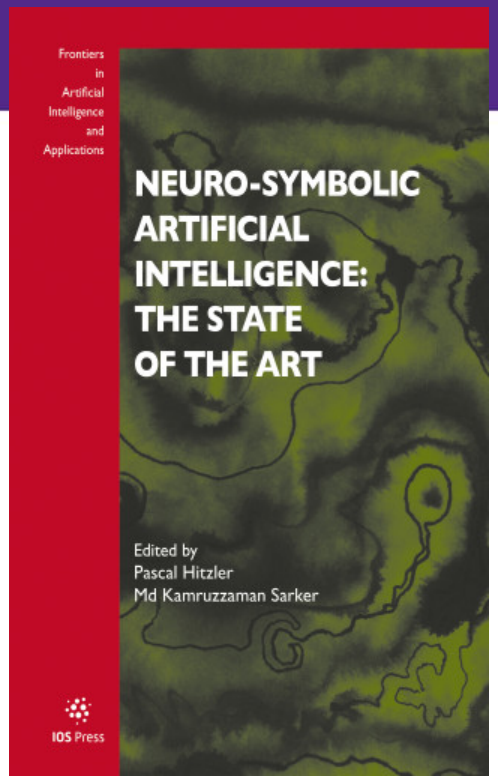
See

Md Kamruzzaman Sarker, Lu Zhou, Aaron Eberhart, Pascal Hitzler

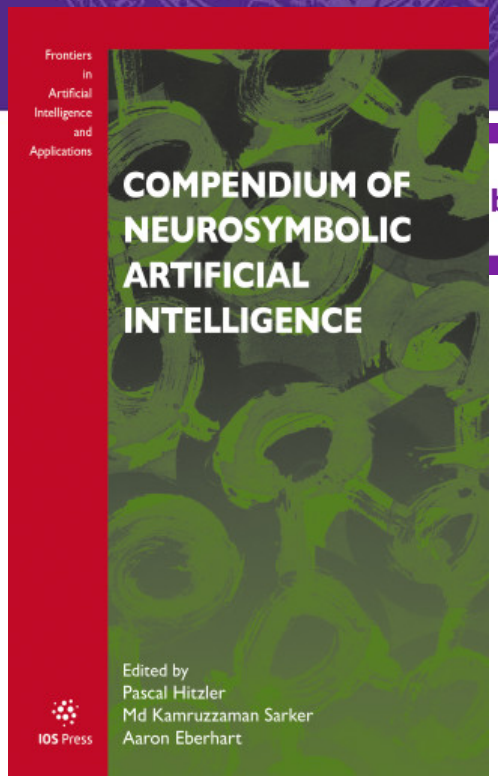
Neuro-Symbolic Artificial Integration: Current Trends

AI Communications 34 (3), 197-209, 2022.





2022, 17 chapters



2023, 30 chapters

Neurosymbolic AI community slack
currently over 800 members
email hitzler@ksu.edu to get an invite

Problem setting:
why we need strong explainabilty for deep learning systems

The black box problem

There have been enormous strides recently in methods and applications of Deep learning.

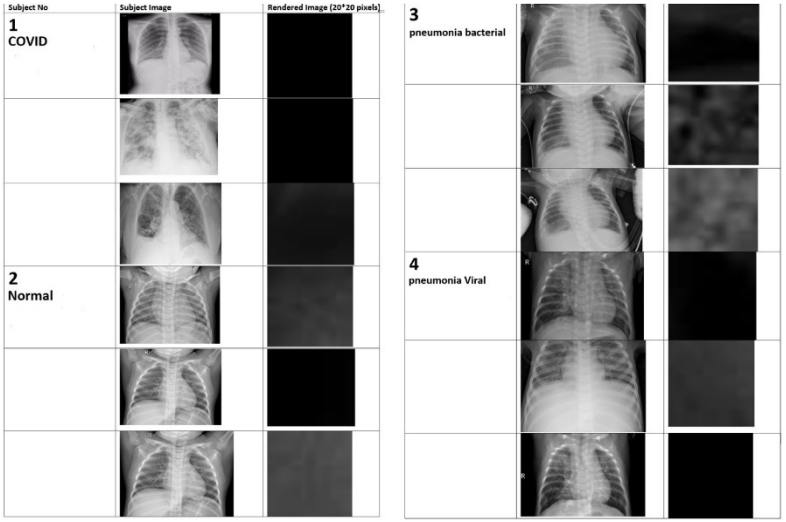
However

- **Deep Learning system are black boxes**
- **Evaluation is only done statistically**

This is insufficient for many application areas, and problematic for most.

The black box problem

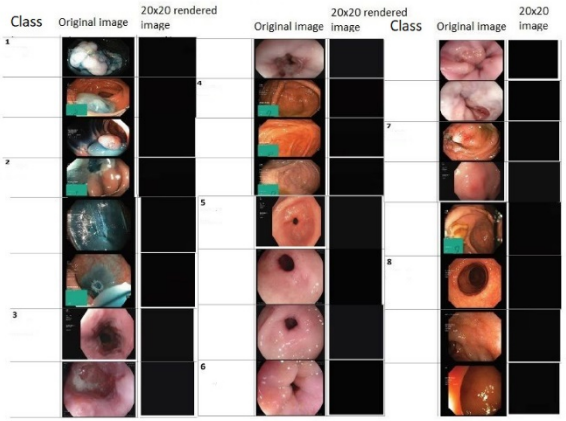
COVID-19 detection



CNN classification accuracy:

Original images – 67%
Blank background images – 62%
Mere chance accuracy – 25%

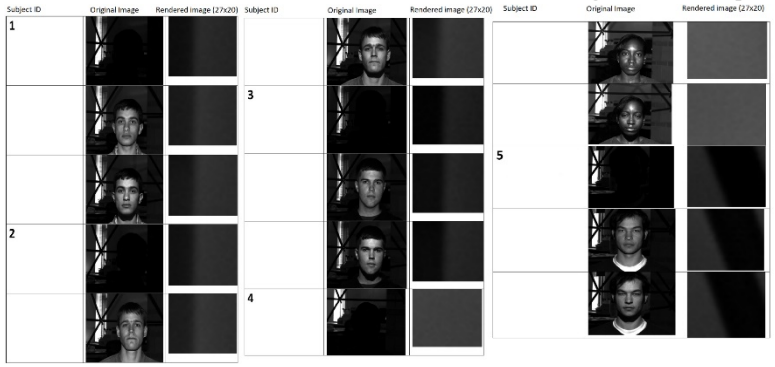
Gastrointestinal disease detection (Kvasir dataset)



CNN classification accuracy:

Original images – 77%
Blank background images – 41%
Mere chance accuracy – 12%

Face recognition (Yale B)



CNN classification accuracy:

Original images – 99%
Blank background images – 87%
Mere chance accuracy – 4%

Dhar, S., Shamir, L., 2021, *Visual Informatics*, 5(3), 92-101 – thanks to Lior Shamir for the slides input

Galaxy image annotation

Classification to spiral galaxies and elliptical galaxies

When the test set and training set are from the same part of the sky, the CNN shows a different Universe than when the training and test images come from different parts of the sky.

SDSS

Training set and test set from the same part of the sky

	Elliptical	Spiral
Elliptical	2891	109
Spiral	85	2915

Training set and test set from different parts of the sky

	Elliptical	Spiral
Elliptical	2704	296
Spiral	31	2969

Pan-STARRS

Training set and test set from the same part of the sky

	Elliptical	Spiral
Elliptical	7850	150
Spiral	756	7244

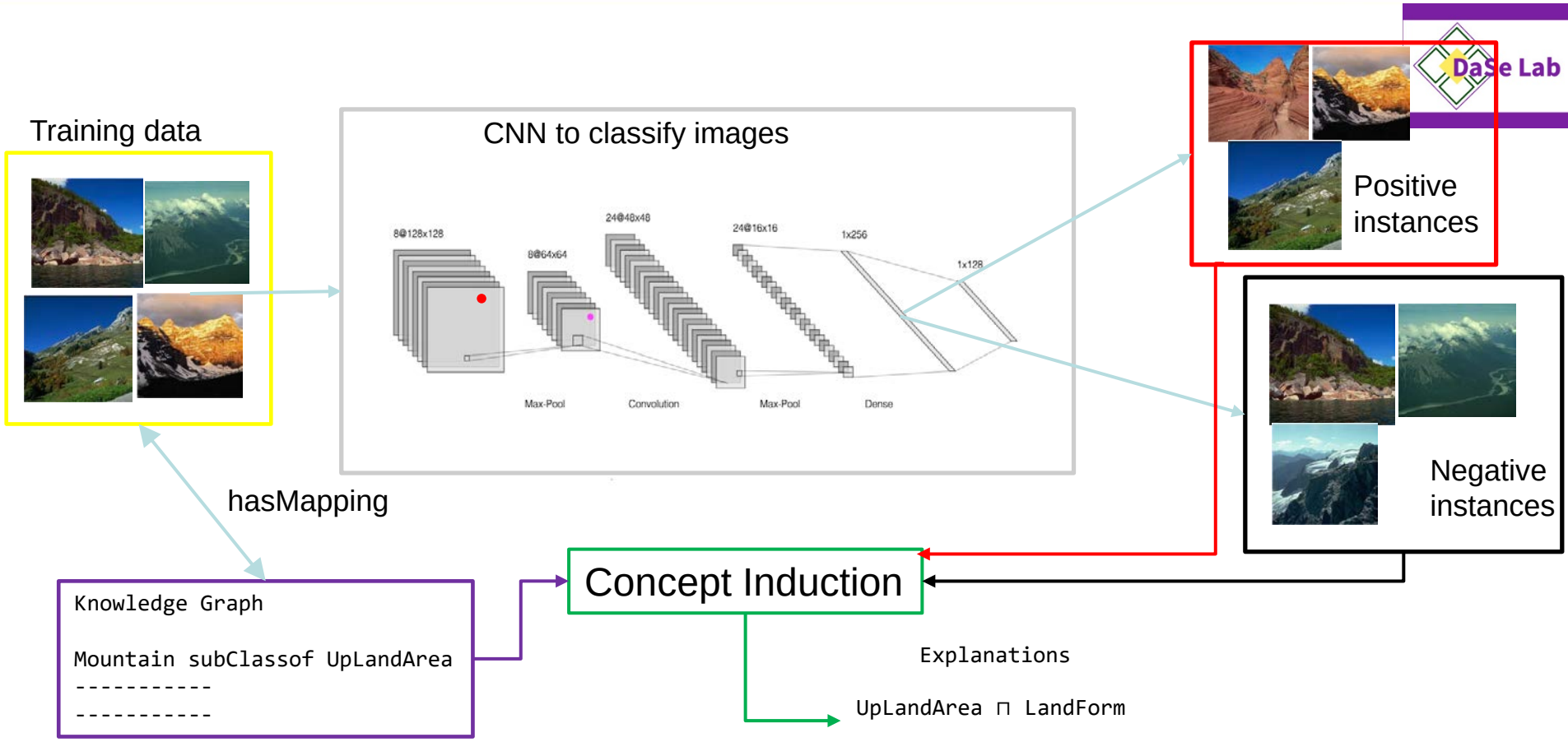
Training set and test set from the same part of the sky

	Elliptical	Spiral
Elliptical	7699	301
Spiral	450	7550

Dhar, S., Shamir, L., 2022, *Astronomy and Computing*, 38, 100545

Approach: Concept Induction for Hidden Layer Analysis

Idea



New results based on: Abhilekha Dalal, Md Kamruzzaman Sarker, Adrita Barua, Eugene Vasserman, Pascal Hitzler https://arxiv.org/abs/2308.03999.

Concept Induction

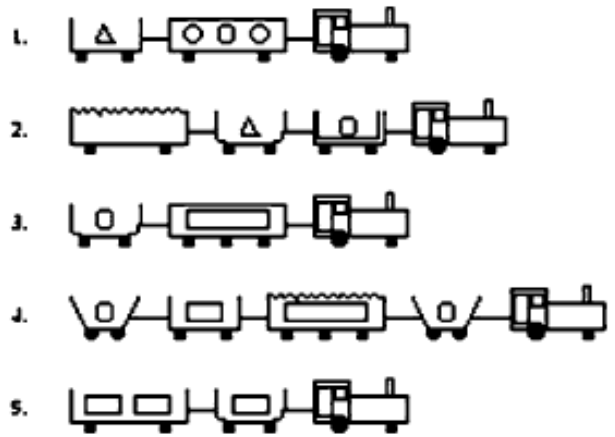
Some slides adapted from Joshua Schwartz, with permission.

Concept Induction

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

Positive examples:

negative examples:

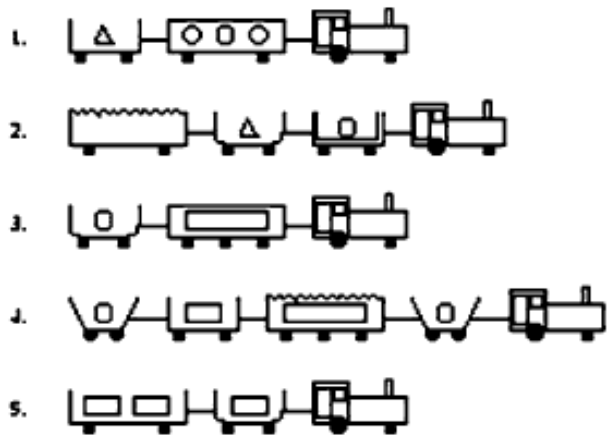


Task: find a class description (logical formula) which separates positive and negative examples.

DL-Learner

Positive examples:

negative examples:



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

In FOL: $\{x \mid \exists y(\mathrm{hasCar}(x, y) \wedge \mathrm{Closed}(y) \wedge \mathrm{Short}(y))\}$

Theory and system: [Lehmann & Hitzler 2010], DL-Learner

```
car(car_11).  car(car_12).  car(car_13).
car(car_14).
car(car_21).  car(car_22).  car(car_23).
car(car_31).  car(car_32).  car(car_33).
car(car_41).  car(car_42).  car(car_43).
car(car_44).
car(car_51).  car(car_52).  car(car_53).
car(car_61).  car(car_62).
car(car_71).  car(car_72).  car(car_73).
car(car_81).  car(car_82).
car(car_91).  car(car_92).  car(car_93).
car(car_94).
car(car_101).  car(car_102).


train(east1).  train(east2).  train(east3).
train(east4).  train(east5).
train(west6).  train(west7).  train(west8).
train(west9).  train(west10).
```

```
// eastbound train 1

has_car(east1,car_11).
has_car(east1,car_12).
has_car(east1,car_13).
has_car(east1,car_14).

short(car_12).
closed(car_12).
long(car_11).
long(car_13).
short(car_14).
open_car(car_11).
open_car(car_13).
open_car(car_14).
shape(car_11,rectangle).
shape(car_12,rectangle).
shape(car_13,rectangle).
shape(car_14,rectangle).
load(car_11,rectangle).
load_count(car_11,three).
load(car_12,triangle).
load_count(car_12,one).
load(car_13,hexagon).
load_count(car_13,one).
load(car_14,circle).
load(car_14,one).
wheels(car_11,two).
wheels(car_12,two).
wheels(car_13,three).
wheels(car_14,two).
```

Somewhat more formally…

generating complex description logic class expressions S from a given description logic knowledge base (or ontology) $\mathcal{O}$ and sets P and N of instances, understood as positive and negative examples, such that $\mathcal{O} \models S(a)$ for all $a \in P$, and $\mathcal{O} \not\models S(b)$ for all $b \in N$

Algorithmically – Refinement Operator

```
Start with simple formula E (e.g., ⊤)
Loop: Expand E minimally in all possible ways to
            E1,...,En
      Check accuracy for E1 through En regarding P and N
      Replace E by highest-scoring Ei
Exit loop if perfect solution found or other stopping
criteria met

Return E
```

In reality, a list of formulas is returned, ranked by accuracy.

Accuracy can be f-measure, precision, recall, etc.

Checking accuracy needs deductive reasoning, i.e., is expensive.

[Lehmann & Hitzler, Machine Learning, 2010], DL-Learner system

Algorithmically – heuristic

- **Restrict allowed syntax expansions (e.g., conjunctions only)**
- **Restrict complexity of logic in background knowledge (e.g., class hierarchy only)**

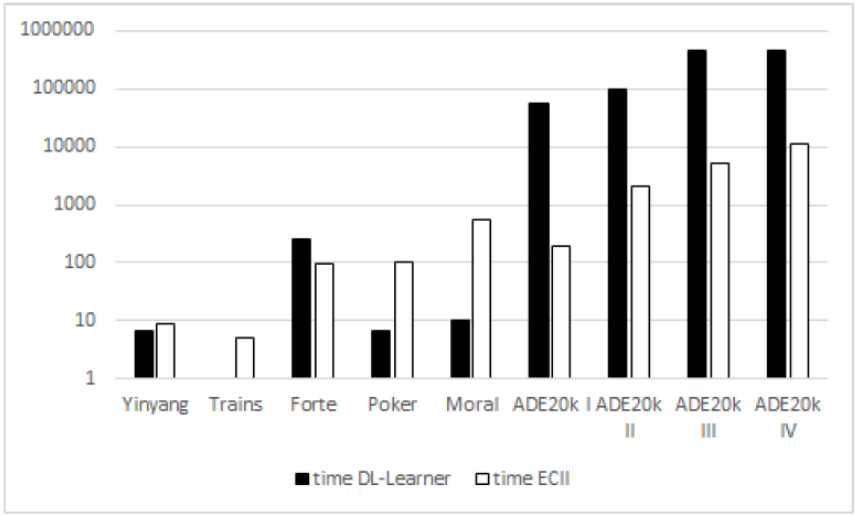


Figure 1: Runtime comparison between DL-Learner and ECII. The vertical scale is logarithmic in hundredths of seconds, and note that DL-Learner runtime has been capped at 4,500 seconds for ADE20k III and IV. For ADE20k I it was capped at each run at 600 seconds.

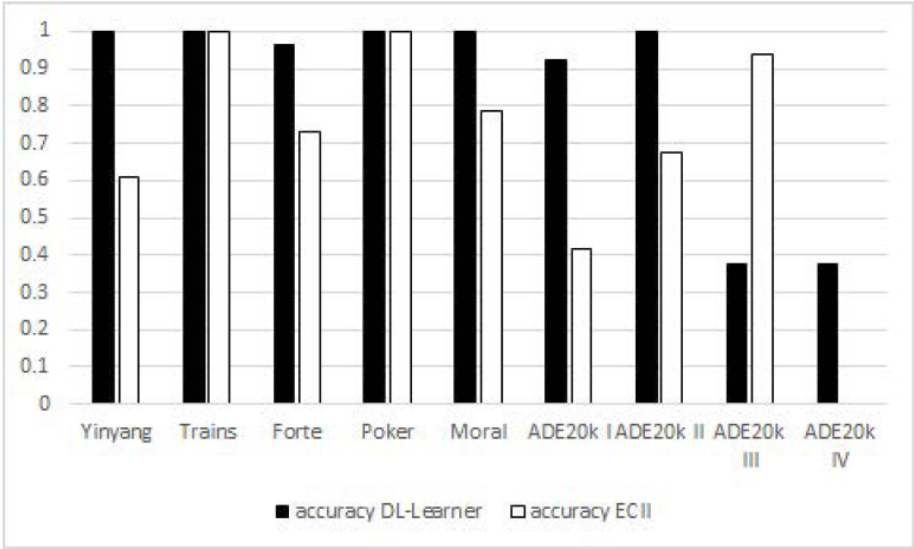


Figure 2: Accuracy (α_3) comparison between DL-Learner and ECII. For ADE20k IV it was not possible to compute an accuracy score within 3 hours for ECII as the input ontology was too large.

[Sarker & Hitzler, AAAI, 2019]: ECII system

Background Knowledge

Background knowledge

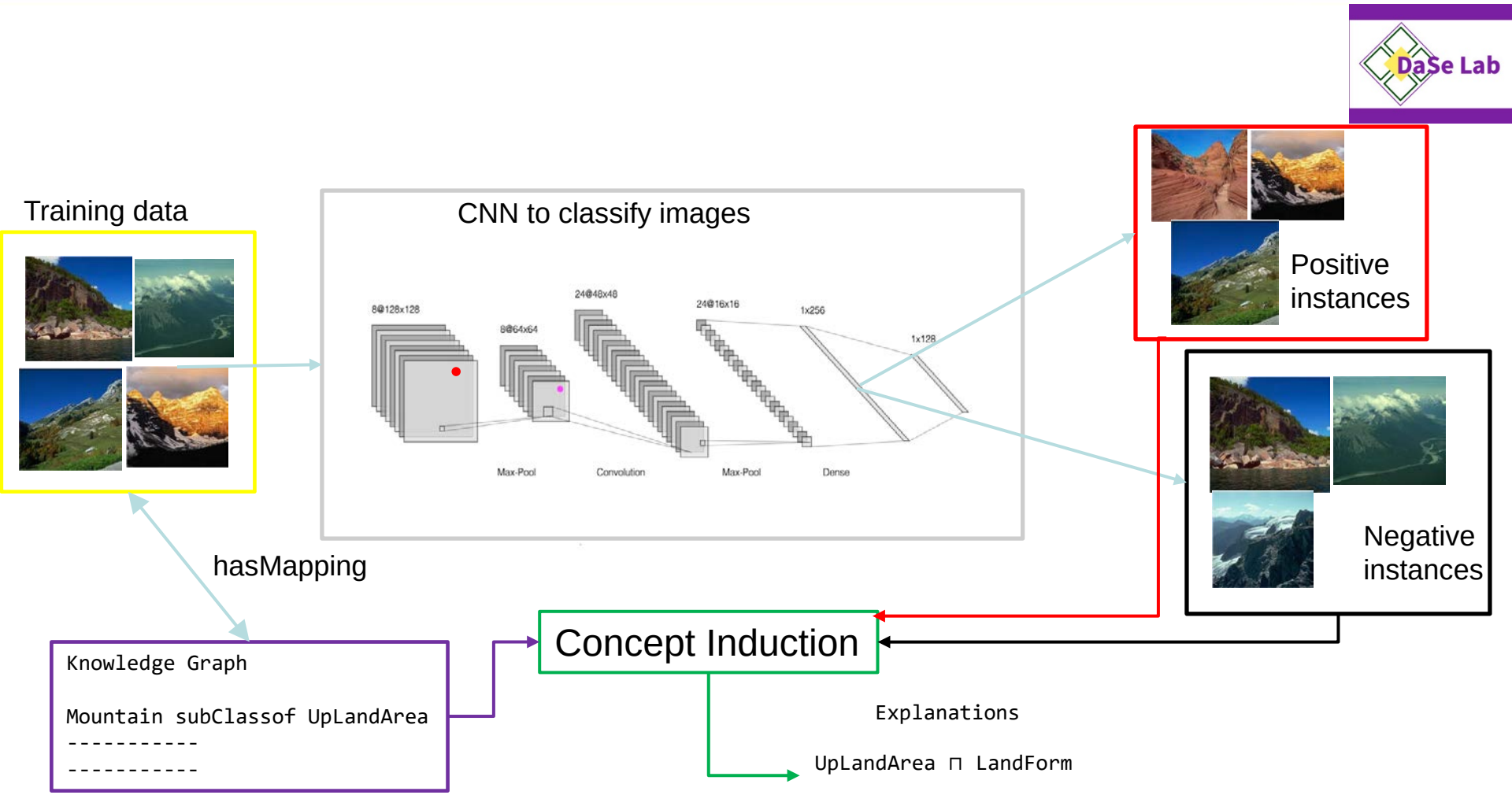
- Based on Wikipedia category hierarchy
  - which is not a hierarchy because it has loops, caused by crowd-sourcing

- Heuristically curated by removing loops
- Resulting class hierarchy has approx. 2M concepts
- Broad coverage (all things in Wikipedia)
- Can easily refer to it from instances by mapping to Wikipedia pages and looking up the page categories.

[Sarker et al., KGSWC2020]

Concrete Setting

Idea


DaSe Lab
Training data
CNN to classify images
8@128x128
8@64x64
24@48x48
24@16x16
1x256
1x128
Max-Pool
Convolution
Max-Pool
Dense
Positive instances
Negative instances
hasMapping
Knowledge Graph
Mountain subClassof UpLandArea

Concept Induction
Explanations
UpLandArea ⊓ LandForm

Scenario

- Scene recognition (from images)
- MIT ADE20k dataset
  http://groups.csail.mit.edu/vision/datasets/ADE20K/
- 10 overlapping scenes selected for our study
- Resnet50V2 trained (best of those we tried)
  - Training accuracy 87.6%
  - Validation accuracy 86.5%

Images annotations

The ADE20k images 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 # ""
```

We ignore everything but the types of object on each image.

Mapping to Background Knowledge

- **String matching (Levenshtein with edit distance 0) from object types to Wikipedia categories**

contains(img1,road1)
contains(img1, window1)
contains(img1, door1)
contains(img1, wheel1)
contains(img1, sidewalk1)
contains(img1, truck1)
contains(img1, box1)
contains(img1, building1)

Label Hypothesis Generation and Confirmation

Trained CNN

- **Scene classification on ADE20k**
- **Resnet50V2; 64 hidden nodes in the dense layer**

	precision	recall	f1-score	support
bathroom	0.90	0.78	0.84	134
bedroom	0.89	0.88	0.88	277
building_facade	0.68	0.60	0.64	45
conference_room	0.77	0.91	0.83	33
dining_room	0.75	0.84	0.79	82
highway	0.96	0.88	0.92	59
kitchen	0.84	0.87	0.86	130
living_room	0.76	0.74	0.75	139
skyscraper	0.90	0.88	0.89	64
street	0.92	0.96	0.94	407
accuracy			0.87	1370
macro avg	0.84	0.83	0.83	1370
weighted avg	0.87	0.87	0.87	1370

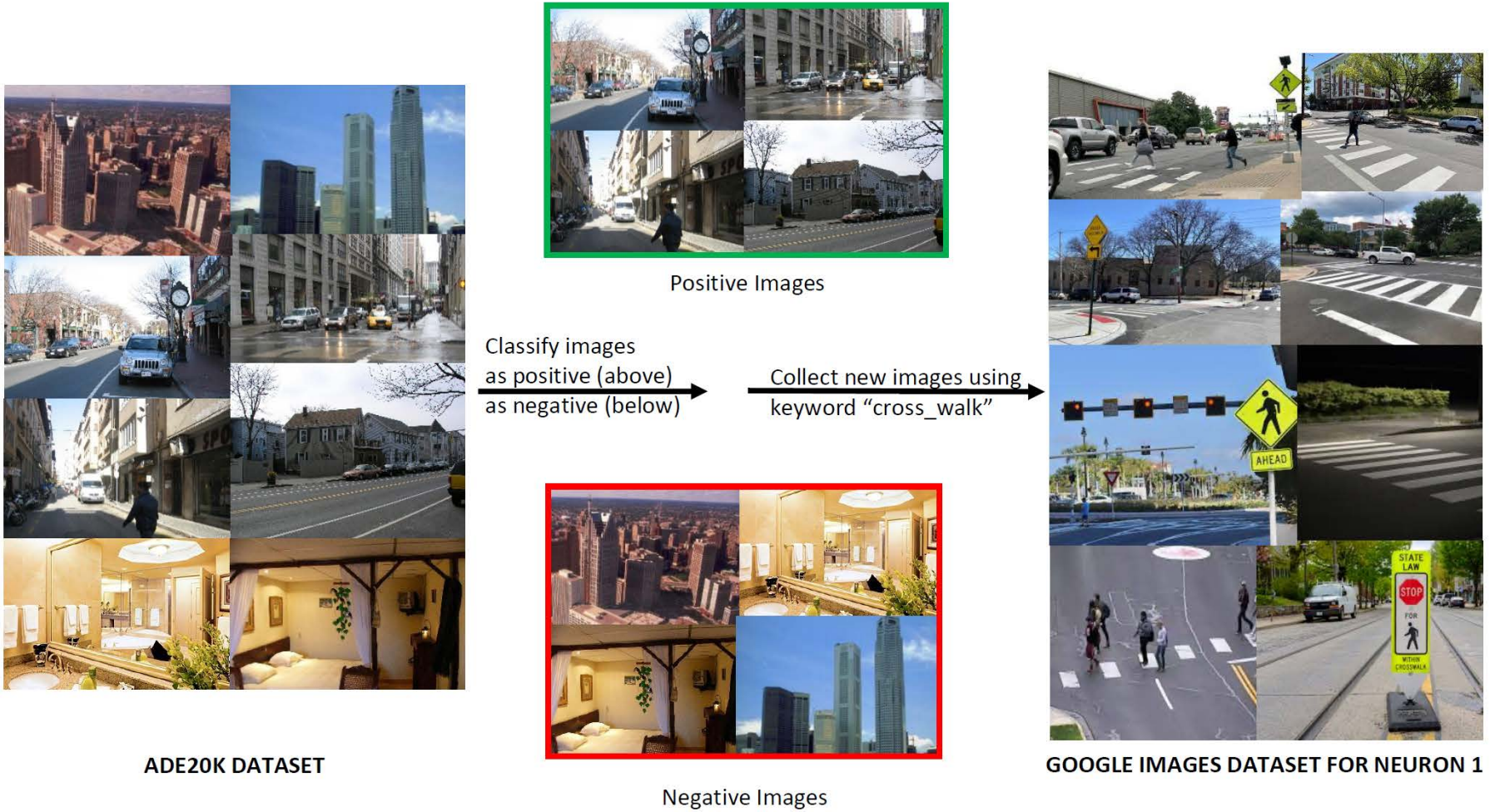


Figure 1: Example of images that were used for generating and confirming the label hypothesis for neuron 1

workflow: label hypothesis generation and confirmation of label hypothesis with new images from Google images

Neuron #	Obtained Label(s)	Images	Coverage	Target %	Non-Target %
0	**building**	**164**	**0.997**	**89.024**	**72.328**
1	**cross_walk**	**186**	**0.994**	**88.710**	**28.923**
3	**night_table**	**157**	**0.987**	**90.446**	**56.714**
6	dishcloth, toaster	106	0.999	16.038	39.078
7	toothbrush, Pipage	112	0.991	75.893	59.436
8	**shower_stall, cistern**	**136**	**0.995**	**100.000**	**53.186**
11	river_water	157	0.995	31.847	22.309
12	baseboard, dish_rag	108	0.993	75.926	48.248
14	rocking_horse, rocker	86	0.985	54.651	47.816
16	**mountain, bushes**	**108**	**0.995**	**87.037**	**24.969**
17	stem	133	0.993	30.827	31.800
18	**slope**	**139**	**0.983**	**92.086**	**69.919**
19	**wardrobe, air_conditioning**	**110**	**0.999**	**89.091**	**65.034**
20	fire_hydrant	158	0.990	5.696	13.233
22	**skyscraper**	**156**	**0.992**	**99.359**	**54.893**
23	fire_escape	162	0.996	61.111	18.311
25	spatula, nuts	126	0.999	2.381	0.883
26	skyscraper, river	112	0.995	77.679	35.489
27	manhole, left_arm	85	0.996	35.294	26.640
28	flooring, fluorescent_tube	115	1.000	38.261	33.198
29	**lid, soap_dispenser**	**131**	**0.998**	**99.237**	**78.571**
30	**teapot, saucepan**	**108**	**0.998**	**81.481**	**47.984**
31	fire_escape	162	0.961	77.160	63.147
33	tanklid, slipper	81	0.987	41.975	30.214
34	left_foot, mouth	110	0.994	20.909	49.216

Neuron #	Obtained Label(s)	Images	Coverage	Target %	Non-Target %
35	utensils_canister, body	111	0.999	7.207	11.223
36	**tap, crapper**	**92**	**0.997**	**89.130**	**70.606**
37	cistern, doorcase	101	0.999	21.782	24.147
38	letter_box, go_cart	125	0.999	28.000	31.314
39	side_rail	148	0.980	35.811	34.687
40	sculpture, side_rail	119	0.995	25.210	21.224
41	**open_fireplace, coffee_table**	**122**	**0.992**	**88.525**	**16.381**
42	pillar, stretcher	117	0.998	52.137	42.169
43	**central_reservation**	**157**	**0.986**	**95.541**	**84.973**
44	saucepan, dishrack	120	0.997	69.167	36.157
46	Casserole	157	0.999	45.223	36.394
48	**road**	**167**	**0.984**	**100.000**	**73.932**
49	**footboard, chain**	**126**	**0.982**	**88.889**	**66.702**
50	night_table	157	0.972	65.605	62.735
51	**road, car**	**84**	**0.999**	**98.810**	**48.571**
53	pylon, posters	104	0.985	11.538	17.332
54	**skyscraper**	**156**	**0.987**	**98.718**	**70.432**
56	**flusher, soap_dish**	**212**	**0.997**	**90.094**	**63.552**
57	**shower_stall, screen_door**	**133**	**0.999**	**98.496**	**31.747**
58	plank, casserole	80	0.998	3.750	3.925
59	manhole, left_arm	85	0.994	35.294	21.589
60	paper_towels, jar	87	0.999	0.000	1.246
61	ornament, saucepan	102	0.995	43.137	17.274
62	sideboard	100	0.991	21.000	29.734
63	**edifice, skyscraper**	**178**	**0.999**	**92.135**	**48.761**

Evaluation

Approach

- Each row of the table is a hypothesis, e.g. “neuron 1 activates more strongly on cross_walk images (retrieved from Google images using keyword “cross_walk”) than on other images.”

- Null hypothesis: There is no difference in activations.

- There is no reason to assume a normal distribution,
- hence using Mann-Whitney U test for assessment.

Evaluation results

Neuron #	Label(s)	Images	# Activations (%)		Mean		Median		z-score	p-value
			targ	non-t	targ	non-t	targ	non-t		
0	building	42	80.95	73.40	2.08	1.81	2.00	1.50	-1.28	0.0995
1	cross_walk	47	91.49	28.94	4.17	0.67	4.13	0.00	-8.92	<.00001
3	night_table	40	100.00	55.71	2.52	1.05	2.50	0.35	-6.84	<.00001
8	shower_stall, cistern	35	100.00	54.40	5.26	1.35	5.34	0.32	-8.30	<.00001
16	mountain, bushes	27	100.00	25.42	2.33	0.67	2.17	0.00	-6.72	<.00001
18	slope	35	91.43	68.85	1.59	1.37	1.44	1.00	-2.03	0.0209
19	wardrobe, air_conditioning	28	89.29	65.81	2.30	1.28	2.30	0.84	-4.00	<.00001
22	skyscraper	39	97.44	56.16	3.97	1.28	4.42	0.33	-7.74	<.00001
29	lid, soap_dispenser	33	100.00	80.47	4.38	2.14	4.15	1.74	-5.92	<.00001
30	teapot, saucepan	27	85.19	49.93	2.52	1.05	2.23	0.00	-4.28	<.00001
36	tap, crapper	23	91.30	70.78	3.24	1.75	2.82	1.29	-3.59	<.00001
41	open_fireplace, coffee_table	31	80.65	15.11	2.03	0.14	2.12	0.00	-7.15	<.00001
43	central_reservation	40	97.50	85.42	7.43	3.71	8.08	3.60	-5.94	<.00001
48	road	42	100.00	74.46	6.15	2.68	6.65	2.30	-7.78	<.00001
49	footboard, chain	32	84.38	66.41	2.63	1.67	2.30	1.17	-2.58	0.0049
51	road, car	21	100.00	47.65	5.32	1.52	5.62	0.00	-6.03	<.00001
54	skyscraper	39	100.00	71.78	4.14	1.61	4.08	1.12	-7.60	<.00001
56	flusher, soap_dish	53	92.45	64.29	3.47	1.48	3.08	0.86	-6.47	<.00001
57	shower_stall, screen_door	34	97.06	32.31	2.60	0.61	2.53	0.00	-7.55	<.00001
63	edifice, skyscraper	45	88.89	48.38	2.41	0.83	2.36	0.00	-6.73	<.00001

Table 3: Evaluation details as discussed in Section 4. Images: number of images used for evaluation. # Activations: (targ(et)): Percentage of target images activating the neuron (i.e., activation at least 80% of this neuron's activation maximum); (non-t): Same for all other images used in the evaluation. Mean/Median (targ(et)/non-t(arget)): mean/median activation value for target and non-target images.

Discussion

-target images not activating neuron 1

Non-target images activating neuron 1

Figure 2: Examples of some Google images used: target images ("cross_walk") that did not activate the neuron; non-target images from labels like "central_reservation," "road and car," and "fire_hydrant" that activated the neuron.

Note: "bushes, bush" is the third-highest concept induction output (coverage 0.993; 48.052% of target images activating the neuron)

Going forward

We would really want to have labels with high target activation and low non-target activation.

- **make use of more concept induction results**
- **better background knowledge**
- **optimize parameters (like thresholds)**
- **investigate neuron ensembles (●)**

Label(s)	Images	# Activations (%)	
		targ	non-t
building	42	80.95	73.40
cross_walk	47	91.49	28.94
night_table	40	100.00	55.71
shower_stall, cistern	35	100.00	54.40
mountain, bushes	27	100.00	25.42
slope	35	91.43	68.85
wardrobe, air_conditioning	28	89.29	65.81
skyscraper	39	97.44	56.16
lid, soap_dispenser	33	100.00	80.47
teapot, saucepan	27	85.19	49.93
tap, crapper	23	91.30	70.78
open_fireplace, coffee_table	31	80.65	15.11
central_reservation	40	97.50	85.42
road	42	100.00	74.46
footboard, chain	32	84.38	66.41
road, car	21	100.00	47.65
skyscraper	39	100.00	71.78
flusher, soap_dish	53	92.45	64.29
shower_stall, screen_door	34	97.06	32.31
edifice, skyscraper	45	88.89	48.38

Concluding

- It works!
- But it needs to be refined.

Thanks!

References

Md Kamruzzaman Saker, Lu Zhou, Aaron Eberhart, Pascal Hitzler, Neuro-Symbolic Artificial Intelligence: Current Trends. AI Communications 34 (3), 197-209, 2022.

Pascal Hitzler, Md Kamruzzaman Sarker (eds.), Neuro-Symbolic Artificial Intelligence - The State of the Art. Frontiers in Artificial Intelligence and Applications Vol. 342, IOS Press, Amsterdam, 2022.

Pascal Hitzler, Md Kamruzzaman Sarker, Aaron Eberhart (eds.), Compendium of Neurosymbolic Artificial Intelligence. Frontiers in Artificial Intelligence and Applications Vol. 369, IOS Press, Amsterdam, 2023.

Jens Lehmann, Pascal Hitzler, Concept Learning in Description Logics Using Refinement Operators. Machine Learning 78 (1-2), 203-250, 2010.

References

Md Kamruzzaman Sarker, Pascal Hitzler, Efficient Concept Induction for Description Logics. In:The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019. AAAI Press 2019 , pp. 3036-3043.

Md Kamruzzaman Sarker, Joshua Schwartz, Pascal Hitzler, Lu Zhou, Srikanth Nadella, Brandon Minnery, Ion Juvina, Michael L. Raymer, William R. Aue, Wikipedia Knowledge Graph for Explainable AI. In: Boris Villazón-Terrazas, Fernando Ortiz-Rodríguezm Sanju M. Tiwari, Shishir K. Shandilya (eds.), Knowledge Graphs and Semantic Web. Second Iberoamerican Conference and First Indo-American Conference, KGSWC 2020, Mérida, Mexico, November 26-27, 2020, Proceedings. Communications in Computer and Information Science, vol. 1232, Springer, Heidelberg, 2020, pp. 72-87.

References

Dhar, S., Shamir, L., 2021, Visual Informatics, 5(3), 92-101

Dhar, S., Shamir, L., 2022, Astronomy and Computing, 38, 100545

Cara Widmer, Md Kamruzzaman Sarker, Srikanth Nadella, Joshua Fiechter, Ion Juvina, Brandon Minnery, Pascal Hitzler, Joshua Schwartz, Michael Raymer, Towards Human-Compatible XAI: Explaining Data Differentials with Concept Induction over Background Knowledge https://arxiv.org/abs/2209.13710

New results based on: Abhilekha Dalal, Md Kamruzzaman Sarker, Adrita Barua, Eugene Vasserman, Pascal Hitzler, https://arxiv.org/abs/2308.03999.

Thanks!