UCLA Computer Science

STAR AI
RESEARCH LAB
UCLA

AI can learn from data. But can it learn to reason?

Guy Van den Broeck

IBM Neuro-Symbolic AI Workshop - Jan 27 2023

Outline

1. The paradox of learning to reason from data

   ~~deep learning~~

2. Learning with symbolic knowledge

   logical reasoning + deep learning

Outline

1. **The paradox of learning to reason from data**

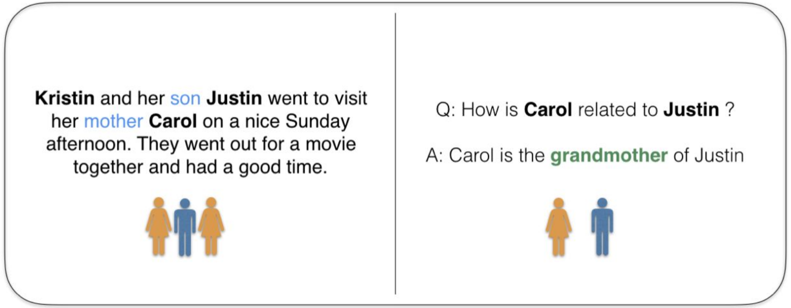
   ~~deep learning~~

2. Learning with symbolic knowledge

   logical reasoning + deep learning

Can Language Models Perform Logical Reasoning?

Language Models achieve high performance on various “reasoning” benchmarks in NLP.



Reasoning Example from the CLUTRR dataset

It is unclear whether they solve the tasks following the rules of logical deduction.

<u>Language Models</u>:
input → ? → Carol is the grandmother of Justin.

<u>Logical Reasoning</u>:
input → Justin in Kristin’s son; Carol is Kristin’s mother; → Carol is Justin’s mother’s mother; if X is Y’s mother’s mother then X is Y’s grandmother → Carol is the grandmother of Justin.

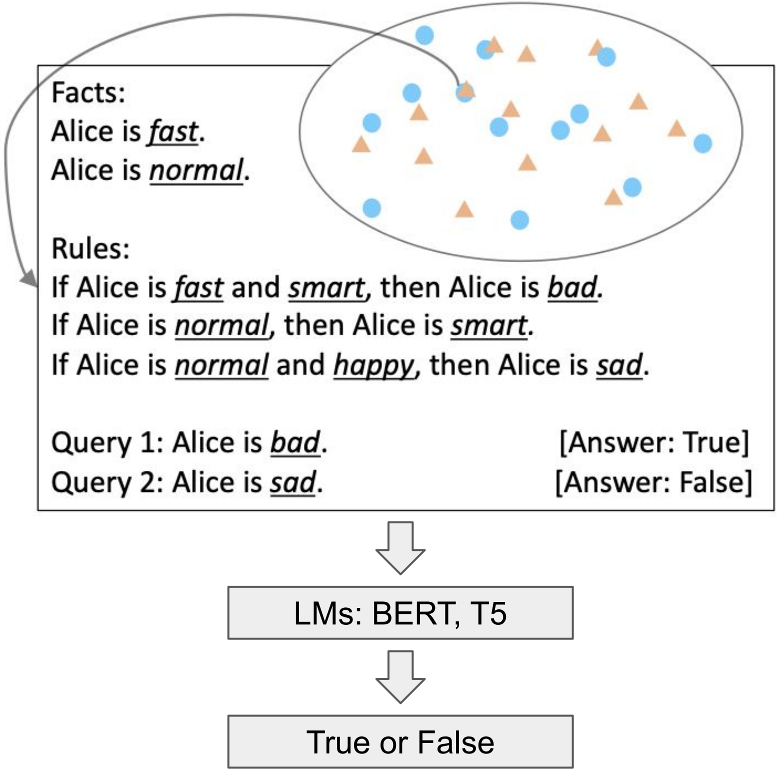
Problem Setting: SimpleLogic

Rules: If witty, then diplomatic. If careless and condemned and attractive, then blushing. If dishonest and inquisitive and average, then shy. If average, then stormy. If popular, then blushing. If talented, then hurt. If popular and attractive, then thoughtless. If blushing and shy and stormy, then inquisitive. If adorable, then popular. If cooperative and wrong and stormy, then thoughtless. If popular, then sensible. If cooperative, then wrong. If shy and cooperative, then witty. If polite and shy and thoughtless, then talented. If polite, then condemned. If polite and wrong, then inquisitive. If dishonest and inquisitive, then talented. If blushing and dishonest, then careless. If inquisitive and dishonest, then troubled. If blushing and stormy, then shy. If diplomatic and talented, then careless. If wrong and beautiful, then popular. If ugly and shy and beautiful, then stormy. If shy and inquisitive and attractive, then diplomatic. If witty and beautiful and frightened, then adorable. If diplomatic and cooperative, then sensible. If thoughtless and inquisitive, then diplomatic. If careless and dishonest and troubled, then cooperative. If hurt and witty and troubled, then dishonest. If scared and diplomatic and troubled, then average. If ugly and wrong and careless, then average. If dishonest and scared, then polite. If talented, then dishonest. If condemned, then wrong. If wrong and troubled and blushing, then scared. If attractive and condemned, then frightened. If hurt and condemned and shy, then witty. If cooperative, then attractive. If careless, then polite. If adorable and wrong and careless, then diplomatic. Facts: Alice sensible Alice condemned Alice thoughtless Alice polite Alice scared Alice average
Query: Alice is shy ?

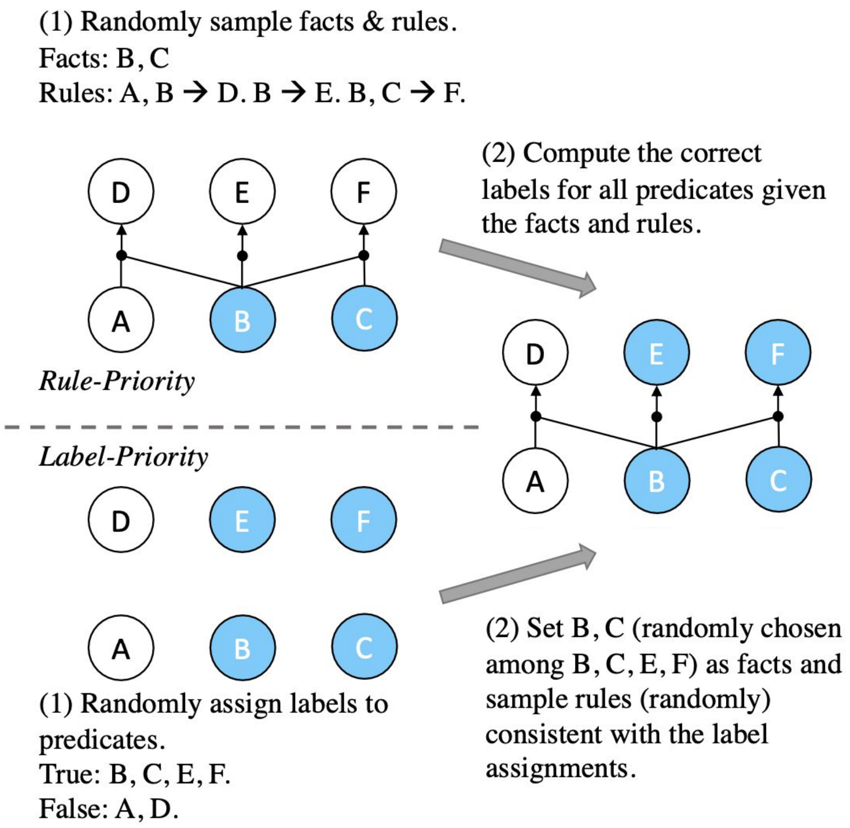
Problem Setting: SimpleLogic

The easiest of reasoning problems:

1. **Propositional logic** fragment
   a. bounded vocabulary & number of rules
   b. bounded reasoning depth (≤ 6)
   c. finite space (≈ 10^360)
2. **No language variance**: templated language
3. **Self-contained**
   No prior knowledge
4. **Purely symbolic** predicates
   No shortcuts from word meaning
5. **Tractable** logic (definite clauses)
   Can always be solved efficiently



Training a BERT model on SimpleLogic



Test accuracy for different reasoning depths

Test	0	1	2	3	4	5	6
RP	99.9	99.8	99.7	99.3	98.3	97.5	95.5

Test	0	1	2	3	4	5	6
LP	100.0	100.0	99.9	99.9	99.7	99.7	99.0

Honghua Zhang, Liunian Harold Li, Tao Meng, Kai-Wei Chang and Guy Van den Broeck. On the Paradox of Learning to Reason from Data, 2022

Has BERT learned to reason from data?

1. Easiest of reasoning problems (no variance, self-contained, purely symbolic, tractable)
2. RP/LP data covers the whole problem space
3. The learned model has almost 100% test accuracy
4. There exist BERT parameters that compute the ground-truth reasoning function:

> <u>Theorem 1:</u> *For a BERT model with* n *layers and 12 attention heads, by construction, there exists a set of parameters such that the model can correctly solve any reasoning problem in SimpleLogic that requires at most* n − 2 *steps of reasoning.*

Surely, under these conditions,
BERT has learned the ground-truth reasoning function!

Honghua Zhang, Liunian Harold Li, Tao Meng, Kai-Wei Chang and Guy Van den Broeck. On the Paradox of Learning to Reason from Data, 2022

The Paradox of Learning to Reason from Data

Train	Test	0	1	2	3	4	5	6
RP	RP	99.9	99.8	99.7	99.3	98.3	97.5	95.5
	LP	99.8	99.8	99.3	96.0	90.4	75.0	57.3
LP	RP	97.3	66.9	53.0	54.2	59.5	65.6	69.2
	LP	100.0	100.0	99.9	99.9	99.7	99.7	99.0

The BERT model trained on one distribution fails to generalize to the other distribution within the same problem space.

1. If BERT **has learned** to reason,
   it should not exhibit such generalization failure.

2. If BERT **has not learned** to reason,
   it is baffling how it achieves near-perfect in-distribution test accuracy.

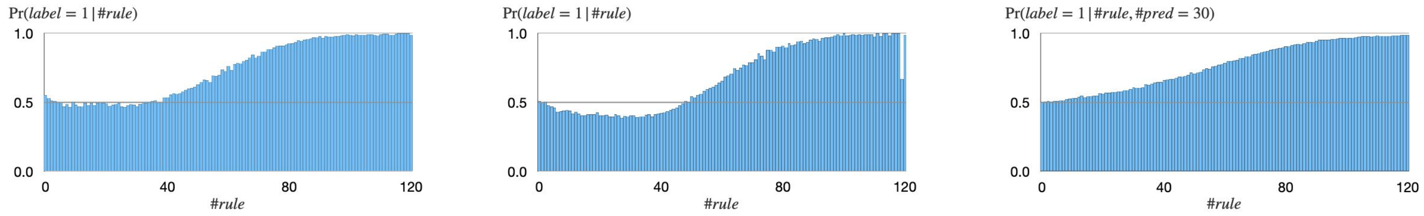
Honghua Zhang, Liunian Harold Li, Tao Meng, Kai-Wei Chang and Guy Van den Broeck. On the Paradox of Learning to Reason from Data, 2022

Why? Statistical Features

Monotonicity of entailment:
Any rules can be freely added to the hypothesis of any proven fact.

The more rules given, the more likely a predicate will be proved.

Pr(label = True | Rule # = x) should increase (roughly) monotonically with x



(a) Statistics for examples generated by Rule-Priority (RP). (b) Statistics for examples generated by Label-Priority (LP). (c) Statistics for examples generated by uniform sampling;

BERT leverages statistical features to make predictions

RP_b downsamples from RP such that Pr(label = True | rule# = x) = 0.5 for all x

Train	Test	0	1	2	3	4	5	6
	RP	99.9	99.8	99.7	99.3	98.3	97.5	95.5
RP	RP_b	99.0	99.3	98.5	97.5	96.7	93.5	88.3

1. Accuracy drop from RP to RP_b indicates that
   the model is using rule# as a statistical feature to make predictions.

2. Potentially countless statistical features

3. Such features are **inherent to the reasoning problem**, cannot make data "clean"

Honghua Zhang, Liunian Harold Li, Tao Meng, Kai-Wei Chang and Guy Van den Broeck. On the Paradox of Learning to Reason from Data, 2022

First Conclusion

Experiments unveil the fundamental difference between

1. learning to reason, and
2. learning to achieve high performance on benchmarks using statistical features.

Be careful deploying AI in applications where this difference matters.

Honghua Zhang, Liunian Harold Li, Tao Meng, Kai-Wei Chang and Guy Van den Broeck. On the Paradox of Learning to Reason from Data, 2022

Outline

1. The paradox of learning to reason from data

   ~~deep learning~~

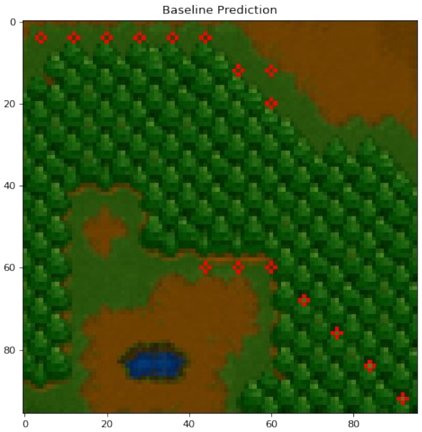
2. **Learning with symbolic knowledge**

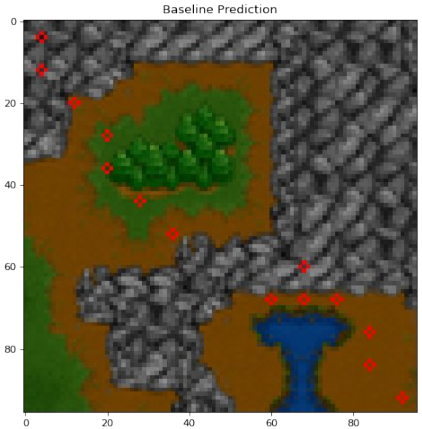
   logical reasoning + deep learning

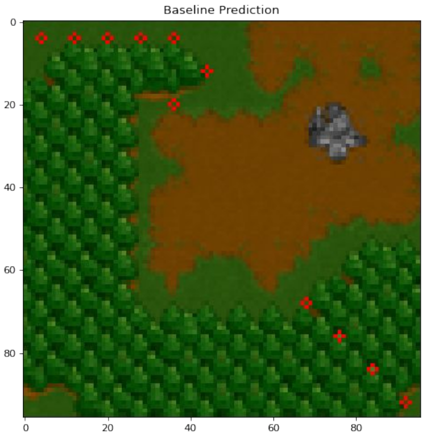
Warcraft Shortest Path

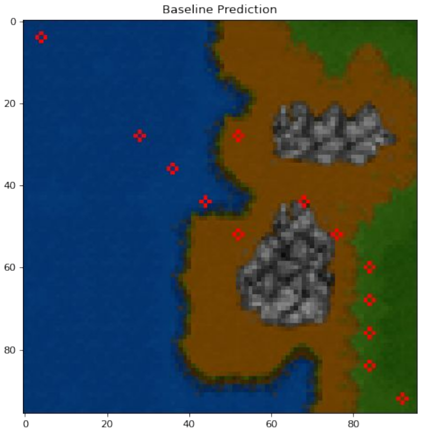
// for a 12×12 *grid,* 2^{144} *states but only* 10^{10} *valid ones!*

[Differentiation of Blackbox Combinatorial Solvers, Marin Vlastelica, Anselm Paulus, Vít Musil, Georg Martius, Michal Rolínek, 2019]


Baseline Prediction


Baseline Prediction


Baseline Prediction


Baseline Prediction

Warcraft min-cost simple-path prediction results

Test accuracy %	Coherent	Incoherent	Constraint
ResNet-18	44.8	**97.7**	56.9

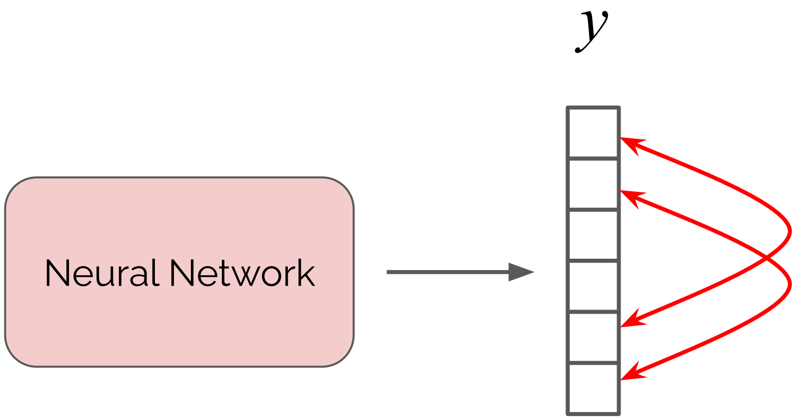
Is prediction the shortest path?
This is the real task!

Are individual edge predictions correct?

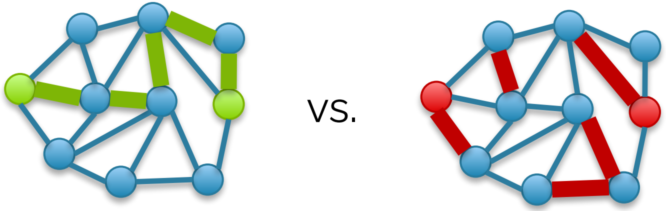
Is output a path?

Kareem Ahmed, Eric Wang, Kai-Wei Chang and Guy Van den Broeck. Neuro-Symbolic Entropy Regularization, 2021.

Declarative Knowledge of the Output



How is the output structured?
Are all possible outputs valid?



How are the outputs related to each other?

Learning this from data is inefficient
Much easier to express this declaratively

pylon

PyTorch Code

```
for i in range(train_iters):
    ...
    py = model(x)
    ...
    loss = CrossEntropy(py,...)
```

1 Specify knowledge as a predicate

```
def check(y):
    ...
    return isValid
```

pylon

PyTorch Code

```
for i in range(train_iters):
    ...
    py = model(x)
    ...
    loss = CrossEntropy(py,...)

    loss += constraint_loss(check)(py)
```

1 Specify knowledge as a predicate

```
def check(y):
    ...
    return isValid
```

2 Add as loss to training

```
loss += constraint_loss(check)
```

pylon

PyTorch Code

```
for i in range(train_iters):
    ...
    py = model(x)
    ...
    loss = CrossEntropy(py,...)

    loss += constraint_loss(check)(py)
```

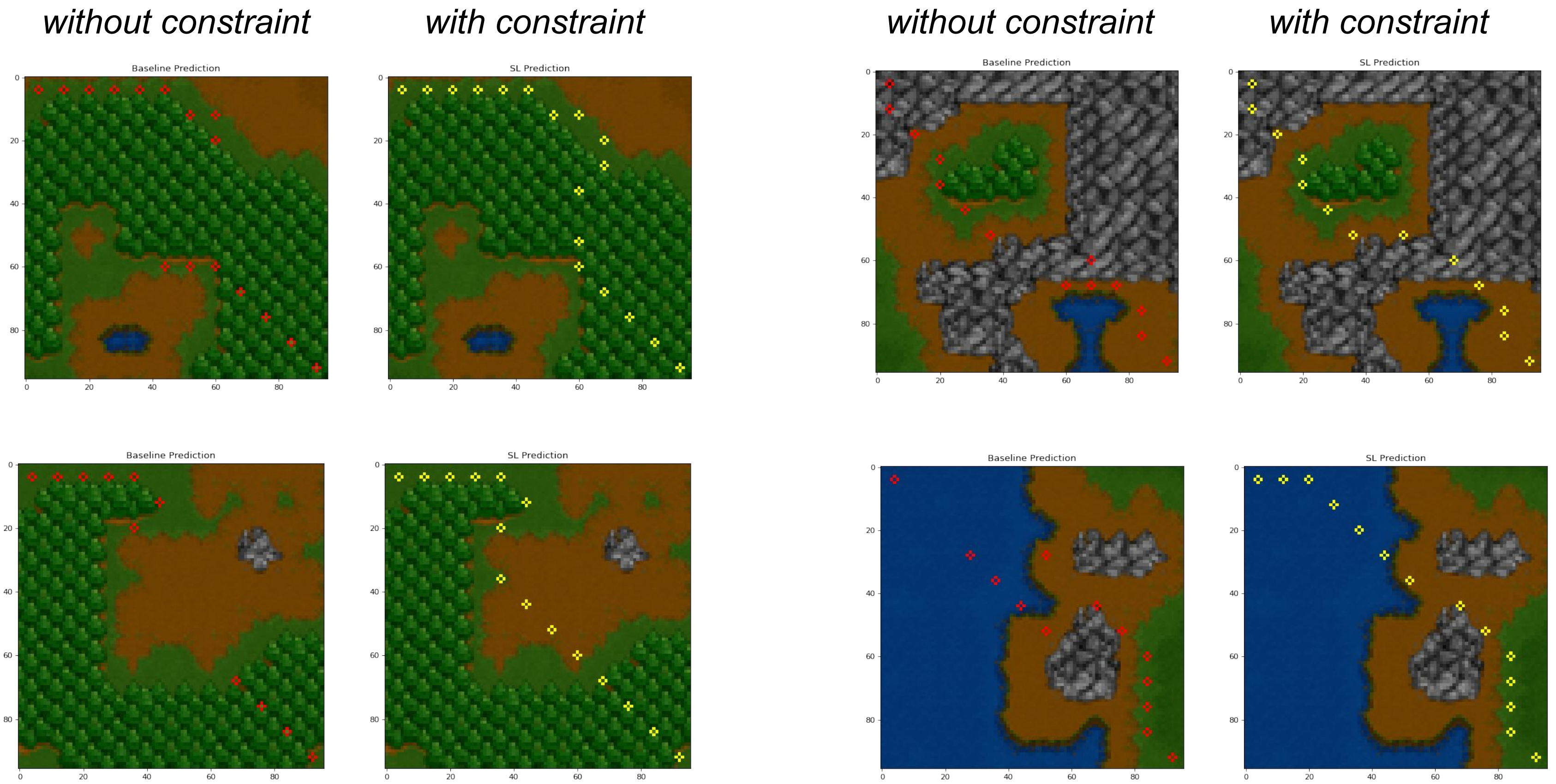
1 Specify knowledge as a predicate

```
def check(y):
    ...
    return isValid
```

2 Add as loss to training

```
loss += constraint_loss(check)
```

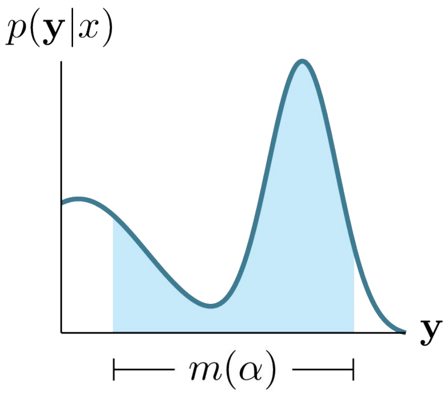
3 pylon derives the gradients
(solves a combinatorial problem)


without constraint
with constraint
without constraint
with constraint
Baseline Prediction
SL Prediction
Baseline Prediction
SL Prediction
Baseline Prediction
SL Prediction
Baseline Prediction
SL Prediction

Warcraft min-cost simple-path prediction results

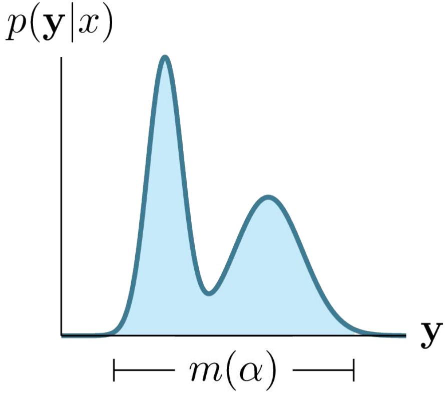
Test accuracy %	Coherent	Incoherent	Constraint
ResNet-18	44.8	**97.7**	56.9
+ Semantic loss	**50.9**	**97.7**	**67.4**

Kareem Ahmed, Eric Wang, Kai-Wei Chang and Guy Van den Broeck. Neuro-Symbolic Entropy Regularization, 2021.



a) A network uncertain over both valid & invalid predictions

Semantic Loss



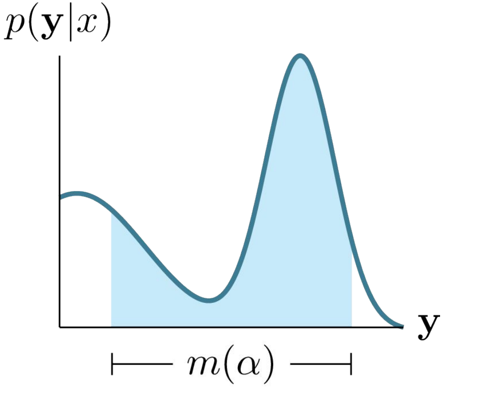
c) A network allocating most of its mass to models of constraint

$$\mathrm{L}^s(\alpha, \mathsf{p}) \propto -\log \underbrace{\sum_{\mathbf{x} \models \alpha} \prod_{i:\mathbf{x} \models X_i} \mathsf{p}_i \prod_{i:\mathbf{x} \models \neg X_i} (1 - \mathsf{p}_i)}$$

Probability of satisfying constraint α after sampling from neural net output layer **p**

In general: #P-hard ☹️

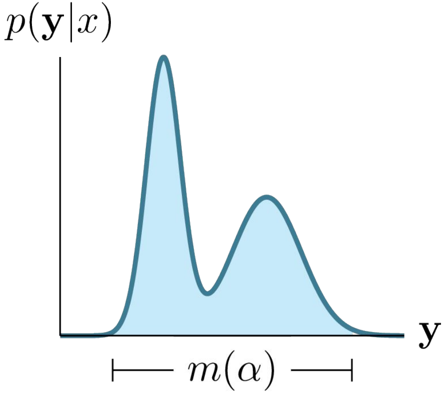
Do this probabilistic-logical reasoning during learning in a computation graph



a) A network uncertain over both valid & invalid predictions

Neuro-Symbolic Learning

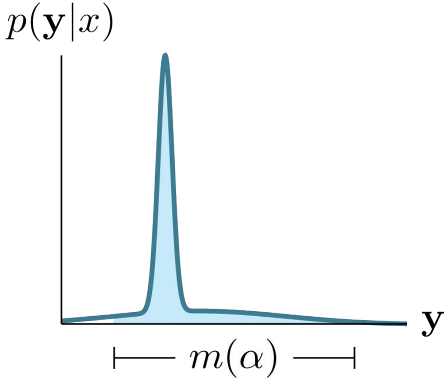
Test accuracy %	Coherent	Incoherent	Constraint
ResNet-18	44.8	97.7	56.9
Semantic loss	50.9	97.7	67.4
+ Full Entropy	51.5	97.6	67.7
+ NeSy Entropy	**55.0**	**97.9**	**69.8**



c) A network allocating most of its mass to models of constraint

Neuro-Symbolic Entropy Regularization

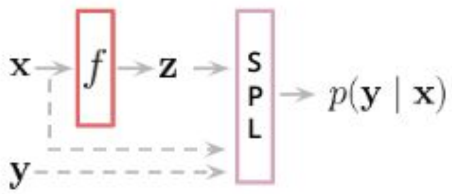
$$-\mathbb{E}_{P(\mathbf{Y}|x,\alpha)}[\log P(\mathbf{Y}|x,\alpha)]$$

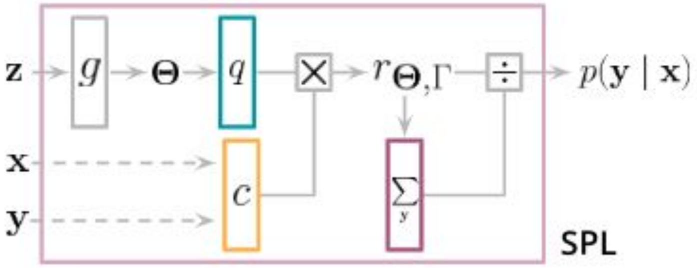


d) A network allocating most of mass to one model of formula

Semantic Probabilistic Layers

- How to give a 100% guarantee that Boolean constraints will be satisfied?
- Bake the constraint into the neural network as a special layer





- Secret sauce is tractable circuits – computation graphs for reasoning

Kareem Ahmed, Stefano Teso, Kai-Wei Chang, Guy Van den Broeck and Antonio Vergari. Semantic Probabilistic Layers for Neuro-Symbolic Learning, 2022.

Warcraft Shortest Path

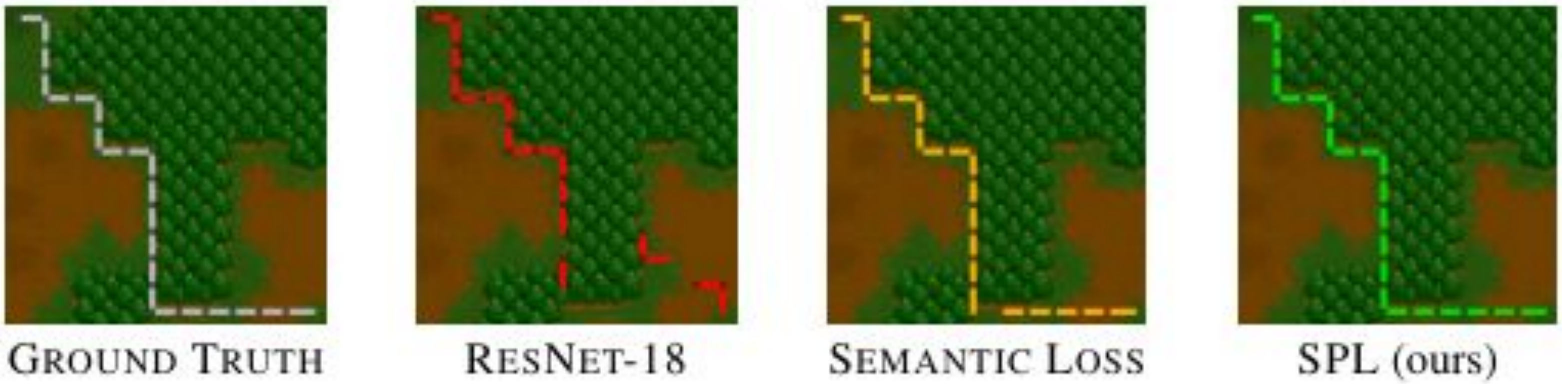
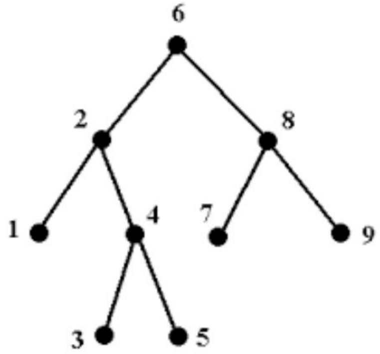


Table 3: Warcraft shortest path prediction results

ARCHITECTURE	EXACT MATCH	HAMMING SCORE	CONSISTENCY
RESNET-18+FIL	55.0	**97.7**	56.9
RESNET-18+$\mathcal{L}_{\text{SL}}$	59.4	**97.7**	61.2
RESNET-18+SPL	75.1	97.6	**100.0**
OVERPARAM. SDD	**78.2**	96.3	**100.0**

Kareem Ahmed, Stefano Teso, Kai-Wei Chang, Guy Van den Broeck and Antonio Vergari. Semantic Probabilistic Layers for Neuro-Symbolic Learning, 2022.

Hierarchical Multi-Label Classification



“if the image is classified as a dog, it must also be classified as an animal”

“if the image is classified as an animal, it must be classified as either cat or dog”

DATASET	EXACT MATCH	
	HMCNN	MLP+SPL
CELLCYCLE	3.05 ± 0.11	$\mathbf{3.79 \pm 0.18}$
DERISI	1.39 ± 0.47	$\mathbf{2.28 \pm 0.23}$
EISEN	5.40 ± 0.15	$\mathbf{6.18 \pm 0.33}$
EXPR	4.20 ± 0.21	$\mathbf{5.54 \pm 0.36}$
GASCH1	3.48 ± 0.96	$\mathbf{4.65 \pm 0.30}$
GASCH2	3.11 ± 0.08	$\mathbf{3.95 \pm 0.28}$
SEQ	5.24 ± 0.27	$\mathbf{7.98 \pm 0.28}$
SPO	$\mathbf{1.97 \pm 0.06}$	1.92 ± 0.11
DIATOMS	48.21 ± 0.57	$\mathbf{58.71 \pm 0.68}$
ENRON	5.97 ± 0.56	$\mathbf{8.18 \pm 0.68}$
IMCLEF07A	79.75 ± 0.38	$\mathbf{86.08 \pm 0.45}$
IMCLEF07D	76.47 ± 0.35	$\mathbf{81.06 \pm 0.68}$

Outline

1. The paradox of learning to reason from data

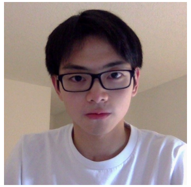
   ~~deep learning~~

2. Learning with symbolic knowledge

   logical (and probabilistic) reasoning + deep learning

Thanks

This was the work of many wonderful students/postdocs/collaborators!



Honghua

Kareem

References: http://starai.cs.ucla.edu/publications/