NEURAL INFORMATION PROCESSING SYSTEMS

KU LEUVEN

LEUVEN.AI

ÖREBRO UNIVERSITY

WASP

Soft-unification in Deep Probabilistic Logic

Jaron Maene & Luc De Raedt

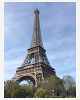
DTAI

DECLARATIEVE TALEN EN ARTIFICIËLE INTELLIGENTIE

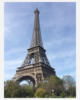
Limitations of logic

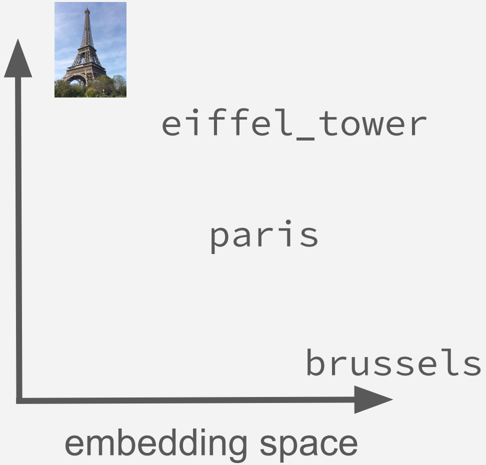
```
locatedIn(eiffel_tower, paris)
```

```
isIn(eiffel_tower, paris)?        ❌
```

locatedIn( , paris)? ❌

Soft-unification: symbols → embeddings

`locatedIn(eiffel_tower, paris) =? locatedIn(`  `, brussels)`



Generalizes knowledge graph embeddings: we retain the full power of first-order logical reasoning.

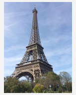
Contributions in short

(1) We give sound probabilistic semantics to learnable soft-unification.

(2) We show the equivalence of soft-unification with existing (neuro-)symbolic frameworks based on (neural) probabilistic facts.

How it works: learning embeddings inside of logic

Query:

locatedIn( , france)?

Program:

locatedIn(eiffel_tower, paris)
locatedIn(paris, france)
locatedIn(X, Y) ←
    locatedIn(X, Z) ∧ locatedIn(Z, Y)

Proofs:

(( ≃ eiffel_tower) ∧ (france ≃ paris))

∨ ( ≃ paris)

∨ ( ≃ eiffel_tower)

∨ ...

Semantics

(( ≃ eiffel_tower) ∧ (france ≃ paris))

∨  ≃ paris)

∨  ≃ eiffel_tower)

Score

Neural Theorem Prover: soft-unification with fuzzy semantics

(0.9 ∧ 0.5) ∨ (0.6) ∨ (0.9)

= max(min(0.9, 0.5), 0.6, 0.9) = 0.9 (Gödel t-norm)

=> end-to-end differentiable!

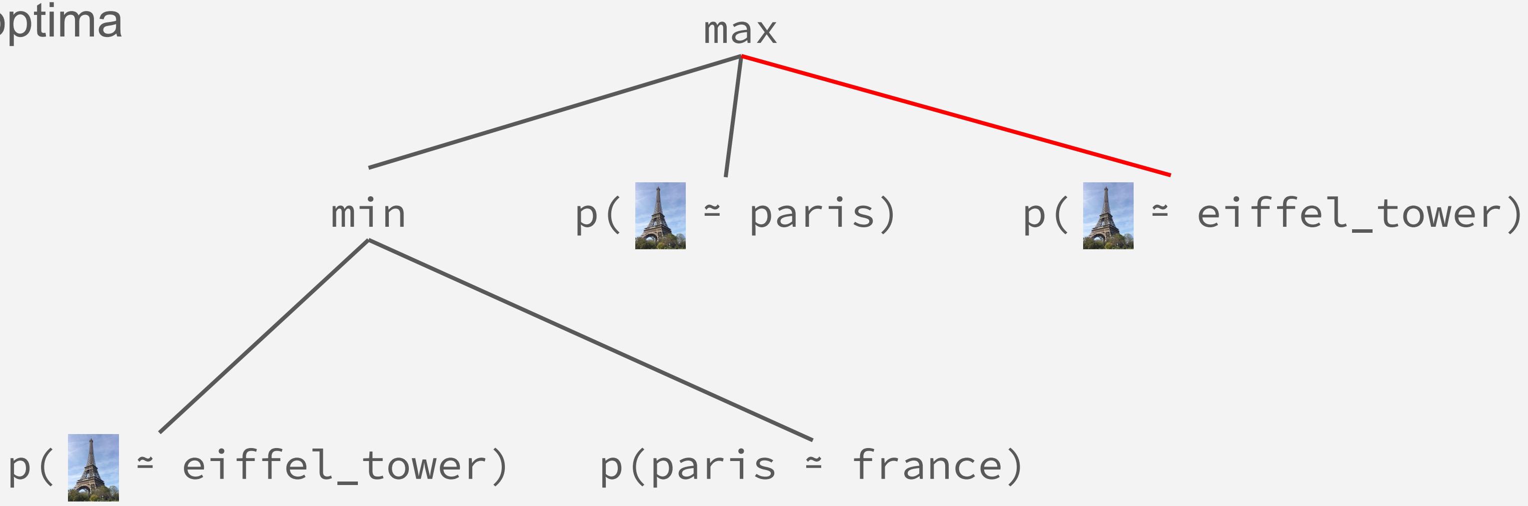
Rocktäschel, Tim, and Sebastian Riedel. "End-to-end differentiable proving."
Advances in neural information processing systems 30 (2017).

Problems with fuzzy semantics

Sparse gradients

→ Inefficient training

→ Local optima



de Jong, Michiel and Fei Sha. “Neural Theorem Provers Do Not Learn Rules Without Exploration.” *ArXiv* abs/1906.06805 (2019).

Problems with fuzzy semantics

Well-defined succes scores

→ Equivalent logic should give equivalent results

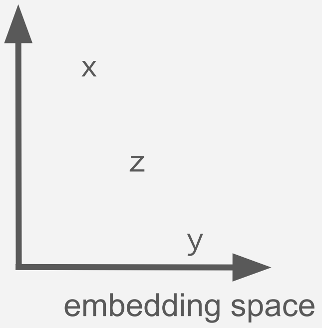
→ Impossible for non-sparse fuzzy semantics

```
a =? a ∧ a
(a ∧ b) ∨ (a ∧ c) =? a ∧ (b ∧ c)
```

Problems with fuzzy semantics

Connected embedding space

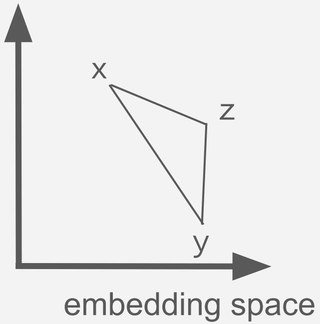
→ Between two embedded symbols x and y there exists a z.



No redundant soft-unifications

→ You can't increase a proof score by inserting soft-unifications.

$$p(x \simeq y) \geq p((x \simeq z) \land (z \simeq y))$$



Theorem: In Gödel t-norm semantics, these properties are mutually exclusive.

Contribution (1)
probabilistic semantics satisfy properties

Theorem: If we interpret the soft-unification as a probability, we and take a soft-unification function of the form $e^{-d(x,y)}$ with d a distance, we get:

1. (1) Well-defined proof scores
2. (2) No redundancy in proofs
3. (3) Connected embedding space
4. (4) Non-sparse gradients

Contribution (2)
soft-unification ↔ (neural) probabilistic facts

```
locatedIn(eiffel_tower, paris)
locatedIn(paris, france)
locatedIn(X, Y) ← locatedIn(X, Z) ∧ locatedIn(X, Y)
```

⇩ source transformation

```
locatedIn(X, Y) ← (X ≃ eiffel_tower) ∧ (Y ≃ paris)
locatedIn(X, Y) ← (X ≃ paris) ∧ (Y ≃ france)
locatedIn(X, Y) ← locatedIn(X, Z) ∧ locatedIn(X, Y)
```

+ *non-linear rules*
+ *grounding of soft-unification*
(cf. paper)

Manhaeve, Robin et al. "DeepProbLog: Neural Probabilistic Logic Programming." *Advances in neural information processing systems* 31 (2018).

DeepSoftLog = ProbLog + soft-unification + neural networks

Extend ProbLog with embedded terms: `~paris, ~vision_model(``), …`

- Embedding is optional
- Embedded functors are neural networks
- Predicates cannot be embedded (but easy to simulate)
- Semantics based on ProbLog

Experiment: knowledge graphs

```
t(~neighbourOf, ~france, ~germany).
t(~locatedIn, ~germany, ~western_europe).
t(~locatedIn, ~western_europe, ~europe).

t(~r1, X, Y) :- t(~r2, Y, X).
t(~r3, X, Y) :- t(~r4, X, Z), t(~r5, Z, Y).
```

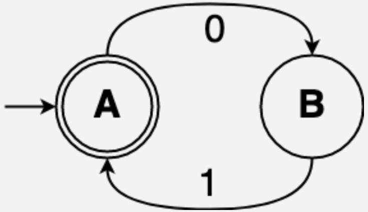
Countries	S1	S2	S3
NTP [29]	90.93 ± 15.4	87.40 ± 11.7	56.68 ± 17.6
GNTP [26]	99.98 ± 0.05	90.82 ± 0.88	87.70 ± 4.79
DeepSoftLog (Ours)	$\mathbf{100.0} \pm 0.00$	$\mathbf{97.67} \pm 0.98$	$\mathbf{97.90} \pm 1.00$
NeuralLP [34]	$\mathbf{100.0} \pm 0.0$	75.1 ± 0.3	92.2 ± 0.2
CTP [27]	$\mathbf{100.0} \pm 0.00$	91.81 ± 1.07	94.78 ± 0.0
MINERVA [8]	$\mathbf{100.0} \pm 0.00$	92.36 ± 2.41	95.10 ± 1.2

Experiment: differentiable finite state machines

Jointly learn perception network finite state machine transitions

Positive examples: , , …

Negative examples: , , …



Language	$(01)^*$	0^*10^*	$(0 \mid 10^*10^*)^*$
RNN	77.63 ± 15.05	61.59 ± 10.09	50.14 ± 1.36
DeepSoftLog	**83.93** ± 25.87	**87.01** ± 7.18	**56.12** ± 15.98

Results: DeepSoftLog is more interpretable and generalizes better, compared to a purely neural baseline.

Thank you!

Paper: https://openreview.net/forum?id=s86M8naPSv
Code: https://github.com/jjcmoon/DeepSoftLog
Twitter: @jjcmoon & @lucderaedt