DeepProbLog: Neural Probabilistic Logic Programming

Robin Manhaeve, Sebastijan Dumančić, Angelika Kimmig, Thomas Demeester*, Luc De Raedt*

* Joint last authors
Real-life problems involve two important aspects.

Sub-symbolic perception

Reasoning with knowledge under uncertainty

Deep Learning

Probabilistic logic program

DeepProbLog = ProbLog + neural predicate

The neural predicate

- Classifier defines a probability distribution over its output
- Uncertainty in the prediction
- Neural predicate: output = probabilistic choices in program
- No changes needed in the ProbLog inference or its semantics
- ProbLog can natively calculate the gradient
Perception
Perception

Reasoning

Stop in front of a red light
Obey the speed limit
Be in the correct lane
\[ P(\text{light} = \text{red}) = 0.9 \]
Perception

Stop in front of a red light
Obey the speed limit
Be in the correct lane

P(light = red) = 0.9

P(obj1 = car) = 0.8
P(obj1 turn right) = 0.7

Reasoning

ProbLog

Neural predicate
Related work

<table>
<thead>
<tr>
<th>Related work</th>
<th>DeepProbLog</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic is made less expressive</td>
<td>Full expressivity is retained</td>
</tr>
<tr>
<td>Logic is pushed into the neural part</td>
<td>Clean separation</td>
</tr>
<tr>
<td>Fuzzy logic</td>
<td>Probabilistic logic</td>
</tr>
<tr>
<td>Language semantics is unclear</td>
<td>Clear semantics</td>
</tr>
</tbody>
</table>

- Neural-symbolic integration (Garcez)
- Logical constraints as a regularizer (Xu, Diligenti, …)
- Differentiable logical framework (Rocktäschel and Riedel, Evans and Grefenstette)
- Differentiable interpreters (Graves, Bosnjak)
- …
Example task: MNIST addition

3 5 0 4 1 + 9 2 1 = ?

• Only labeled sums, not single digits
• Train using only neural networks? Not suited!
• DeepProbLog can solve this:
  • Neural predicate
    • From pixels to distribution over digits
    • NN trained from scratch
  • Logic:
    • Combine predictions into larger numbers
    • Perform addition
Combined reasoning

Unknown distribution
Combined reasoning

Unknown distribution

Perception

0.7::Red

0.8::Blue

0.6::Heads
Combined reasoning

Unknown distribution → Perception → Logical inference

Update parameters → Train NN → Loss

Backpropagation

0.7 :: Red
0.8 :: Blue
0.6 :: Heads

Rules → 0.7 :: win
Conclusion

DeepProbLog: Neural Probabilistic Logic Programming

• Integration of DL and PLP
• Probabilistic
• Clean semantics, clear separation
• Retain power of both worlds
• Power of ProbLog

Code is available at:
https://bitbucket.org/problog/deepproblog