Neural Probabilistic Logic Programming in Discrete-Continuous Domains

Lennert De Smet[1] Pedro Zuidberg Dos Martires[2] Robin Manhaeve[1] Giuseppe Marra[1] Angelika Kimmig[1]
Luc De Raedt[1,2]

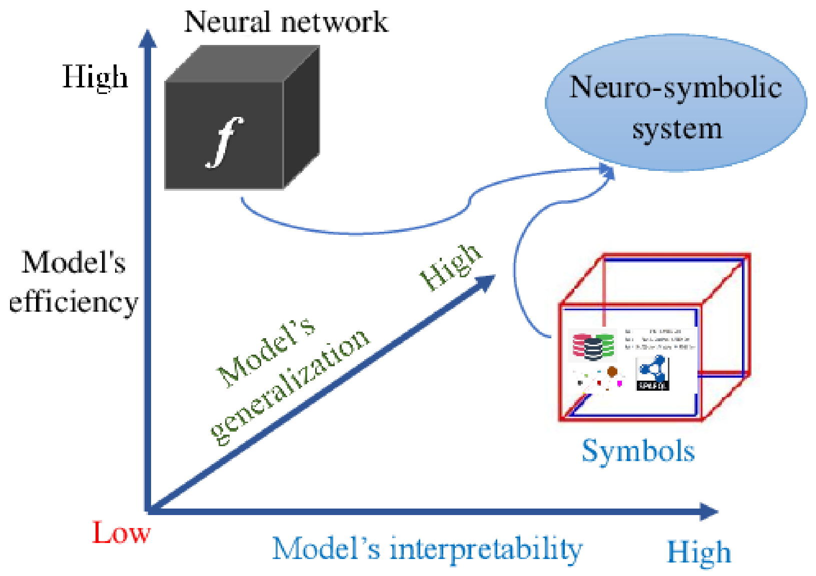
[1]Department of Computer Science, KU Leuven, Belgium
[2]Center for Applied Autonomous Systems, Örebro, Sweden

Matthew Hur

March 11, 2024

Introduction to Neural-Symbolic AI (NeSy)

- Neural-symbolic AI combines the best of both worlds: the learning capabilities of neural networks with the reasoning power of symbolic AI.
- This synergy allows for models that can learn from data while also incorporating structured, logical knowledge, enabling more robust and interpretable AI systems.



https://www.semanticscholar.org/paper/A-Survey-on-Neural-symbolic-Systems-Yu-Yang/2252279ae1e60940d53f08c0ff9113c0cb2b45b4

Motivation for DeepSeaProbLog

- Existing probabilistic NeSy systems, such as DeepProbLog, have shown great promise in their efficacy but are limited by their inability to handle continuous probability distributions.
- DeepSeaProbLog emerges as a solution to this limitation, integrating deep probabilistic programming techniques to model both discrete and continuous random variables within a logical framework.

Logic Programming Concepts

- Logic programming is a paradigm based on formal logic. It uses a set of statements or "clauses" to express facts and rules within a problem domain. Key components include:
- **Terms**: Basic units that can be constants, variables, or structured units.
- **Atoms**: The simplest form of a predicate applying to a set of terms.
- **Literals**: Atoms or their negation.
- **Rules**: Implications defining relationships between predicates, formed as "Head :- Body," where the head is derived if the body is true.

Introduction to DeepSeaProbLog

DeepSeaProbLog is an extension of probabilistic logic programming that incorporates deep learning directly into the logic programming framework.

1. **Syntax and Semantics for Continuous Distributions**: It allows defining continuous random variables as part of the logic program, enabling the direct modeling in both discrete and continuous spaces.
2. **Neural Distributional Facts (NDFs)**: These are special facts where the probability distribution is parameterized by a neural network. This approach enables modeling of problems where the integration of domain knowledge (in the form of logic rules) and data-driven insights (from neural networks) are essential for making informed decisions or predictions.

Semantics of DeepSeaProbLog

- The semantics of DeepSeaProbLog extend traditional logic programming semantics to incorporate probabilistic reasoning with both discrete and continuous distributions.
- **Probabilistic Semantics**: Defines how probabilities are assigned to logical conclusions derived from both deterministic and stochastic rules.
- **Handling of Continuous Variables**: Through the introduction of distributional facts and probabilistic comparison formulas, DeepSeaProbLog allows for the logical manipulation of continuous random variables.

Weighted Model Integration for Inference

- Inference in DeepSeaProbLog is powered by Weighted Model Integration (WMI). WMI generalizes weighted model counting by integrating over the space of variable assignments that satisfy a given formula.
- **Quantitative Reasoning**: Computing the weighted sum or integral over models that satisfy both the logical and probabilistic constraints.

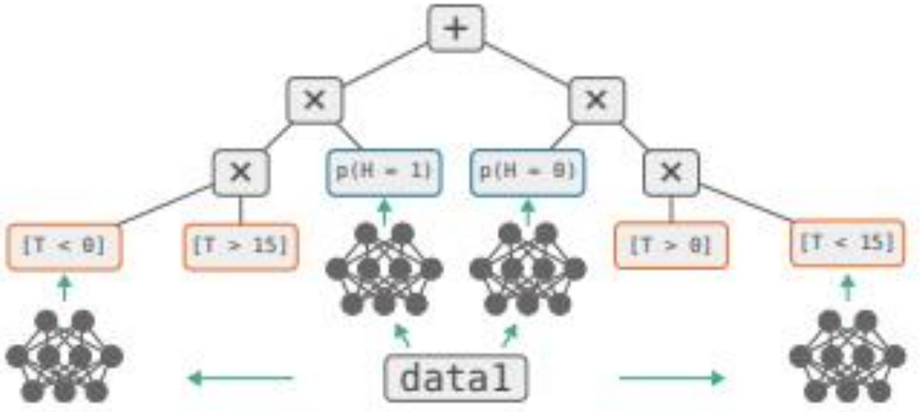


Figure 1: Diagrammatic representation of the result of knowledge compilation for the query in Example 3.3. The blue boxes originate from PCFs over discrete variables, while the orange ones are PCFs over continuous variables. Note how the discrete variable PCFs are reduced to their exact probabilities while the continuous PCFs still need to be inferred.

Probabilistic Programming and Neural Networks

- DeepSeaProbLog represents a significant step forward in bridging probabilistic programming with neural networks. It enables the explicit modeling of complex phenomena that are governed by both deterministic logical rules and stochastic processes.
- By embedding neural models within a probabilistic logic programming framework, DeepSeaProbLog allows for the construction of systems that can reason about uncertainty and learn from data in a way that is both flexible and interpretable.

Learning in DeepSeaProbLog

- The learning framework within DeepSeaProbLog optimizes neural network parameters embedded in logical models
  - **Continuous Relaxations**: Smooth approximations of discrete operations to allow for gradient flow.
  - **Reparametrization Trick**: For continuous variables, this trick enables the computation of gradients of stochastic objectives by transforming the randomness into a deterministic function of inputs and noise.

Inference in DeepSeaProbLog

- Inference in DeepSeaProbLog leverages weighted model integration (WMI), an approach that allows for handling infinite sample spaces by mapping ground programs onto satisfiability modulo theory (SMT) formulas.
- SMT concerns satisfiability of logical formulas composed of Boolean variables.
- This enables efficient inference over continuous distributions, a novel capability in the realm of probabilistic logic programming.

Key Contributions

DeepSeaProbLog's main contributions include:

1. Well-defined probabilistic semantics and the corresponding inference algorithm.

2. An asymptotically unbiased learning algorithm that supports gradient estimation for both discrete and continuous variables named weighted model integration (WMI).

3. Learning algorithm model parameters that can backpropagate through logical expressions, enabling the optimization of parameters in neural networks based on the outcomes of logical inferences.

- Gradients estimated through techniques such as the reparametrization trick for continuous variables
- Relaxation methods for discrete logic components.

Experimental Evaluation

- **Neural-Symbolic Attention Mechanisms**: Integrating attention models with symbolic reasoning for tasks like visual question answering.
- **Optimization of Neural Hybrid Bayesian Networks**: Using DeepSeaProbLog for learning and inference in models that combine neural networks with Bayesian networks.
- **Neural-Symbolic Variational Autoencoders**: Extending VAEs with logical constraints for generative modeling tasks.

Figure 2: On the left, an example of a handwritten year. On the right, the attention map for the digit '8' as a generalised normal distribution. Intuitively, we can view generalised normal distributions as differentiable bounding boxes. This allows gradients to flow from a downstream classification network to the regression component.

Experimental Evaluation (cont.)

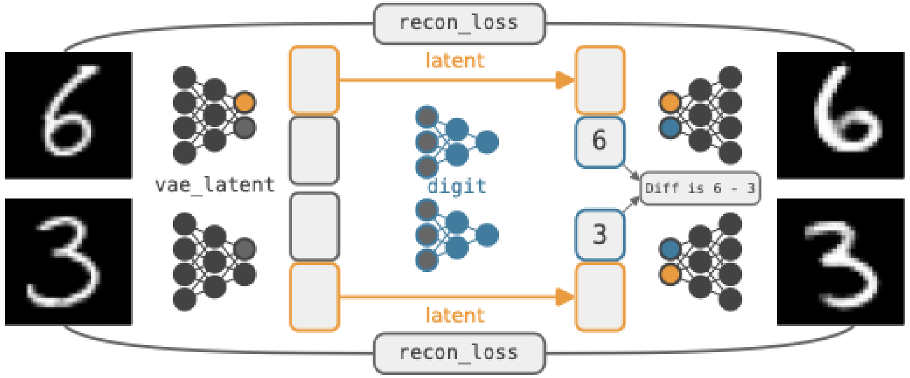


Figure 4: Given example pairs of images and the value of their subtraction, e.g., (6, 3) and 3, the CVAE encoder `vae_latent` first encodes each image into a multivariate normal NDF (`latent`) and a latent vector. The latter is the input of a categorical NDF `digit`, completing the CVAE latent space. Supervision is dual; generated images are compared to the original ones in a probabilistic reconstruction loss, while both digits need to subtract to the given value.

Deep Learning and Logic Programming Integration

- DeepSeaProbLog integrates deep learning directly into logic programming by utilizing Neural Distributional Facts (NDFs). These NDFs enable the embedding of neural network outputs as probabilistic facts within the logic program. This integration is achieved through:
- **Definition of NDFs**: Specifying probabilistic facts whose distributions are parameterized by neural network outputs.

Example Code

```
humid(Data) ~
    bernoulli(humid_detector(Data)).
temp(Data) ~
    normal(temperature_predictor(Data)).

good_weather(Data):-
    humid(Data) =:= 1, temp(Data) < 0.

good_weather(Data):-
    humid(Data) =:= 0, temp(Data) > 15.

query(good_weather(🌍)).
```

Conclusions and Future Directions

- DeepSeaProbLog integrates logic, probability, and neural networks to address complex problems in discrete-continuous domains. Its ability to perform end-to-end learning and reasoning across both types of variables opens up new possibilities for AI research and applications.
- **Future Directions:**
  - Expanding the scalability and efficiency of inference and learning mechanisms.
  - Exploring additional applications in fields such as robotics, natural language processing, and bioinformatics.
  - Enhancing the expressiveness to accommodate more complex logical structures and probabilistic models.

Thank You