StARLinG LAB Explaining Deep Tractable Probabilistic Models: The sum-product network case

Athresh Karanam, Saurabh Mathur, David M Haas,
Predrag Radivojac, Kristian Kersting & Sriraam Natarajan

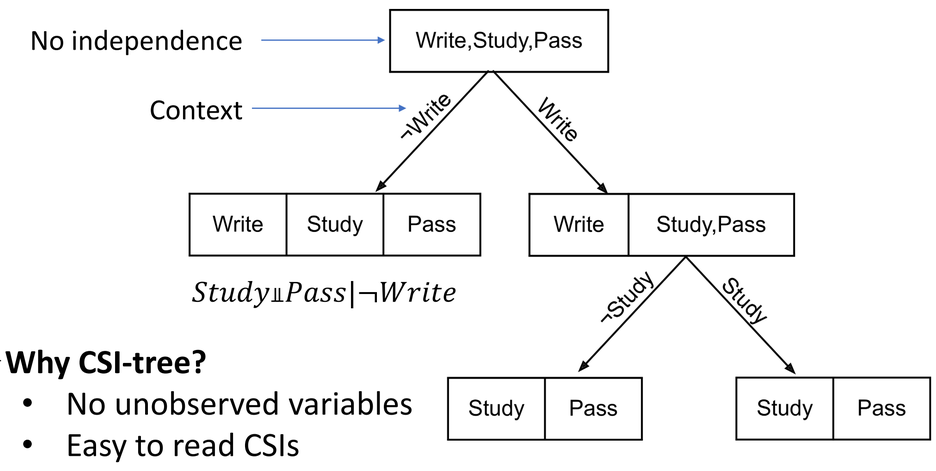
UT Dallas · Technische Universität Darmstadt · Indiana University · Northeastern University

What?

Motivation

- Sum-Product Networks (SPNs) are inherently uninterpretable – internal nodes do not correspond to any feature
- Context-specific independences (CSIs) – Conditional independences that hold under certain instantiations of conditioned variables
- SPNs encode CSIs – product nodes capture independences
- Can convert SPNs into an interpretable representation by leveraging the CSIs encoded by the SPN?

CSI-tree

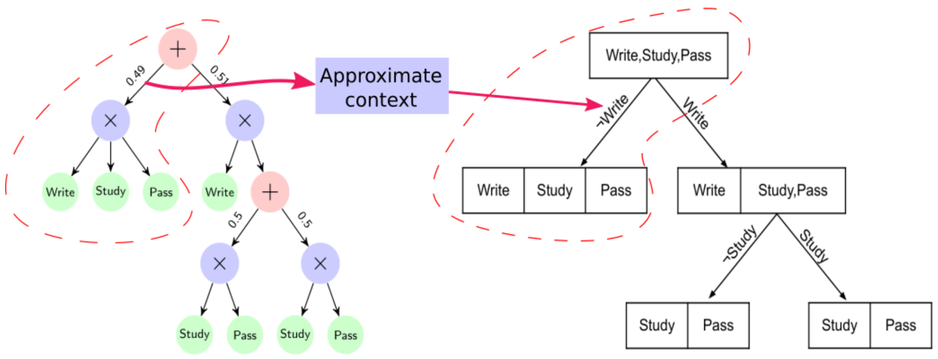


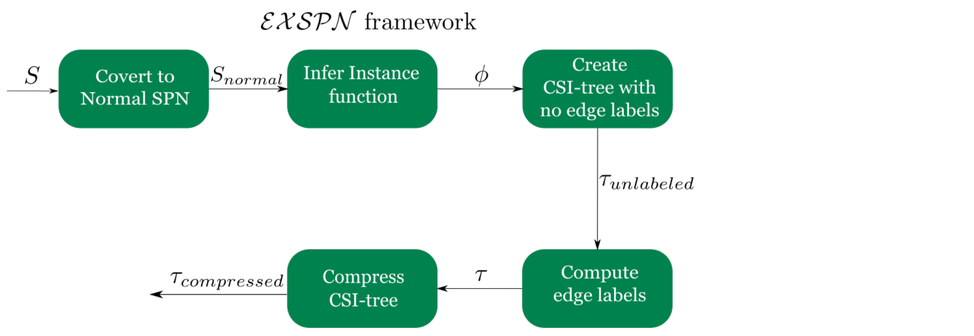
- **Why CSI-tree?**
  - No unobserved variables
  - Easy to read CSIs
  - Expressive

How?

The ExSPN framework

Given: SPN $S = (G, \psi, w, \theta)$, data $\mathcal{D}$
To Do: Extract CSI-tree $\tau = (G_{csi}, \psi_{csi}, \chi, \zeta)$





Algorithm 1: $\mathcal{EXSPN}$

input : $\mathcal{D}$, $\mathbf{X}$, $S = (G = (\mathbf{N}, \mathbf{E}), \psi, w, \theta)$, λ
output: CSI-tree $\tau = (G_{csi}, \psi_{csi}, \chi, \zeta)$

Convert S to a normal-spn S_{normal}
Infer ϕ using alg. 2
Initialize $G_{csi} = G_{normal}, \psi_{csi} = \psi_{normal}, \tau = (G_{csi}, \psi_{csi}, \chi, \zeta)$
st = [G_{csi}.root]
while *st is not empty* **do**
  $N_{current} = st.pop()$
  if $N_{current}$ *is not the root node* **then**
    if $N_{current}$ *is a leaf node* **then**
      Add $\psi(N_{current})$ to $\chi(pa(N_{current}))$
    end
    if $N_{current}$ *is a sum node* **then**
      Delete $N_{current}$ from G_{csi} Add $\psi(N_{current})$ to $\chi(pa(N_{current}))$
      Connect parents and children of $N_{current}$
      Duplicate children with multiple parents and ensure tree-structure
    end
  end
  Add $ch(N_{current})$ to st
end
$\tau = ComputeLabels(\tau, \mathcal{D}, \phi, \lambda)$
return τ

Algorithm 2: InferInstanceFunction

input : $\mathcal{D}$, $\mathbf{X}$, normal SPN S
output: Instance function ϕ

for *Each instance* $\mathcal{D}_j \in \mathcal{D}$ **do**
  Perform upward pass for $\mathcal{D}$ and compute $S_N^{max}(\mathcal{D}_j)$ for each node N
  Perform a downward pass, appending $\mathcal{D}_j$ to the $\phi(N_{ch})$ of a child N_{ch} of a sum node that led to $S^{max}(\mathcal{D}_j)$ and to $\phi(N_{product})$ for all product nodes $N_{product}$ along the path from the root to a leaf node
end
return ϕ

Algorithm 3: ComputeLabels

input : $\tau = (G, \psi, \chi, NULL), \mathcal{D}, \phi, \lambda$
output: Labeled CSI-tree $\tau = (G, \psi, \chi, \zeta)$

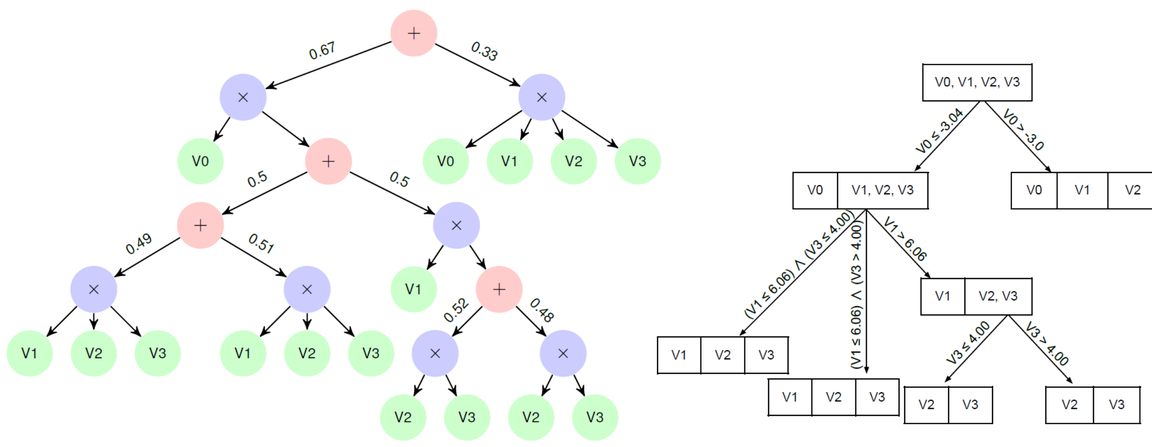
for *Each edge* $e(N_{from}, N_{to})$ *in* G **do**
  Compute class labels y
  $f = TrainModel(\mathcal{D}, \phi(N_{from}), \mathbf{y})$
  Compute a set of important features I for f
  $\zeta(e) = \wedge_{i \in I} i$ thresholded by λ
end
return $\tau = (G, \psi, \chi, \zeta)$

ExSPN:

- outputs CSI-tree that provably recovers the structure of the original normal SPN
- is robust to noise – function approximator used to approximate the context is a generalizable discriminator
- extracts CSI-tree that is linear in the size of a normal SPN

Results

Empirical Evaluations



- The CSIs in the CSI-tree match the ones in the mixture of gaussians used to generate the synthetic data
- For example, V1 ⫫ V2 | V0 > -3 present in the data is also captured by the CSI-tree

ExSPN:

- recovers the CSIs encoded in an SPN – 83% of the CSIs recovered match the ground truth
- extracts CSIs that are concise compared to the Apriori algorithm baseline – up to two orders of magnitude more concise
- outputs CSI-tree on real-world clinical domain that is human-interpretable – validated by our medical expert Dr. Haas

Future Work

- Validating ExSPN on more relevant clinical studies
- Allowing domain experts to interact with the learned model
- Extending ExSPN to a broader class of models – TDPMs and beyond
- More types of explanations – beyond CSIs

Acknowledgement

The authors gratefully acknowledge the support of 1R01HD101246 from NICHD and Precision Health Initiative of Indiana University.

https://starling.utdallas.edu/
@STARLing_lab

References