UCLA Computer Science

STAR AI RESEARCH LAB UCLA

Tractable Deep Generative Models

Guy Van den Broeck

Dagstuhl - Feb 16 2023

Controlled generation is still challenging …

H

generate a sentence with "pan" as the third word and "vegetable" as the fifth word.

The chef used the pan to gently sauté the diced vegetable for their delicious stir-fry dish.

ChatGPT

more reasoning!

Generate image

Stable Diffusion

What do we have?

Prefix: "The weather is"

Constraint α: text contains "winter"

Model only does $p(\text{next-token}|\text{prefix}) =$

cold	0.05
warm	0.10

intractable

Train some $q(.\,|\alpha)$ for a specific task distribution $\alpha \sim p_{\text{task}}$

(amortized inference, encoder, masked model, seq2seq, prompt tuning,...)

Train $q(\text{next-token}|\text{prefix}, \alpha)$

What do we need?

Prefix: "The weather is"

Constraint α: text contains "winter"

Generate from $p(\text{next-token}|\text{prefix}, \alpha) =$

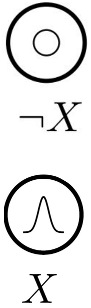
cold	0.50
warm	0.01

$$\propto \sum_{\text{text}} p(\text{next-token}, \text{text}, \text{prefix}, \alpha)$$

Marginalization!

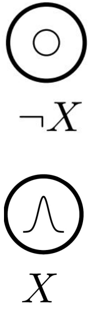
Probabilistic circuits

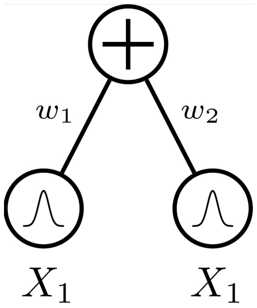
computational graphs that recursively define distributions



Probabilistic circuits

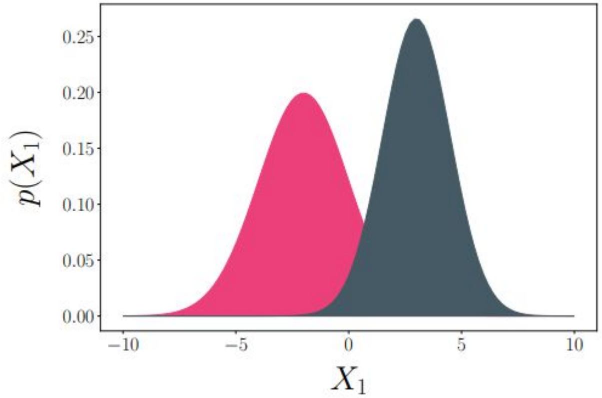
computational graphs that recursively define distributions





$$p(X_1) = w_1 p_1(X_1) + w_2 p_2(X_1)$$

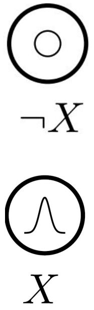
$\Rightarrow$ *mixtures*

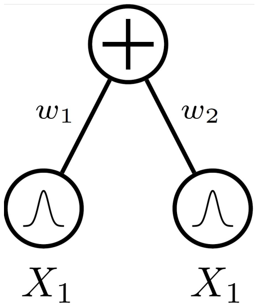


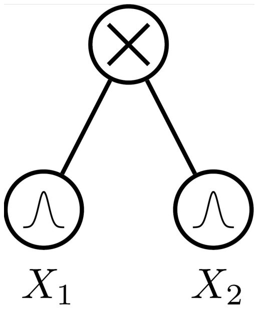
$$p(X) = p(Z = \boxed{1}) \cdot p_1(X \mid Z = \boxed{1})$$
$$+ p(Z = \boxed{2}) \cdot p_2(X \mid Z = \boxed{2})$$

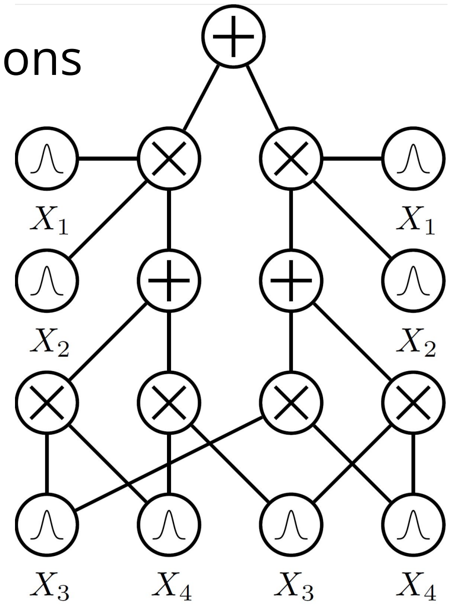
Probabilistic circuits

computational graphs that recursively define distributions









$$p(X_1) = w_1 p_1(X_1) + w_2 p_2(X_1)$$

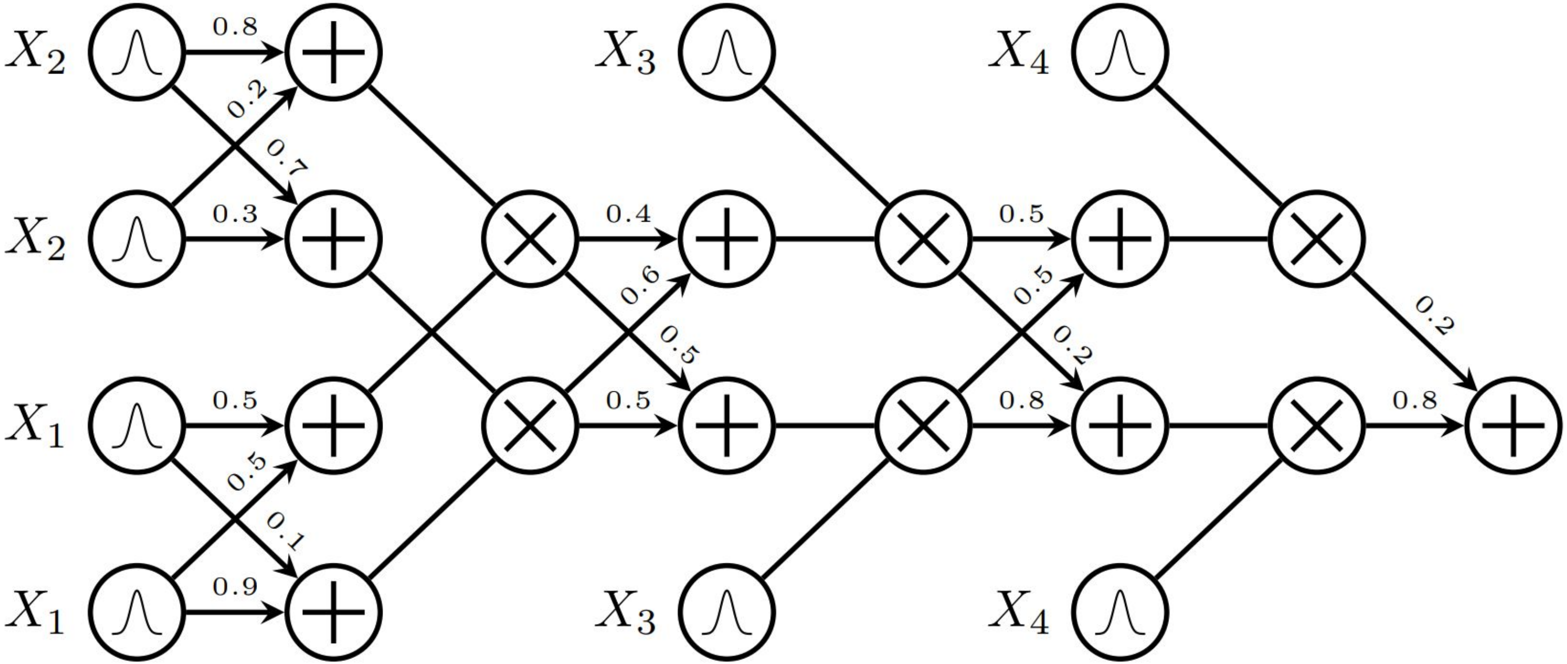
$\Rightarrow$ *mixtures*

$$p(X_1, X_2) = p(X_1) \cdot p(X_2)$$

$\Rightarrow$ *factorizations*

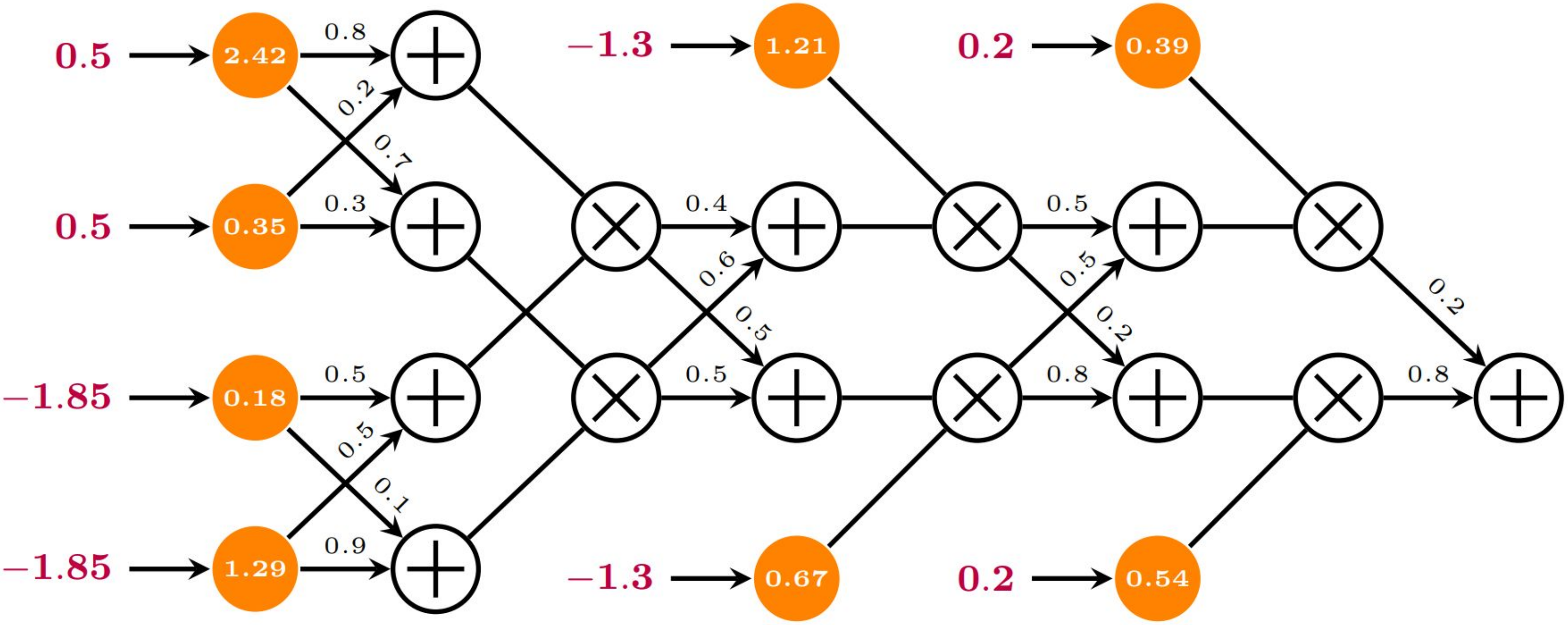
Likelihood

$$p(X_1 = \mathbf{-1.85}, X_2 = \mathbf{0.5}, X_3 = \mathbf{-1.3}, X_4 = \mathbf{0.2})$$



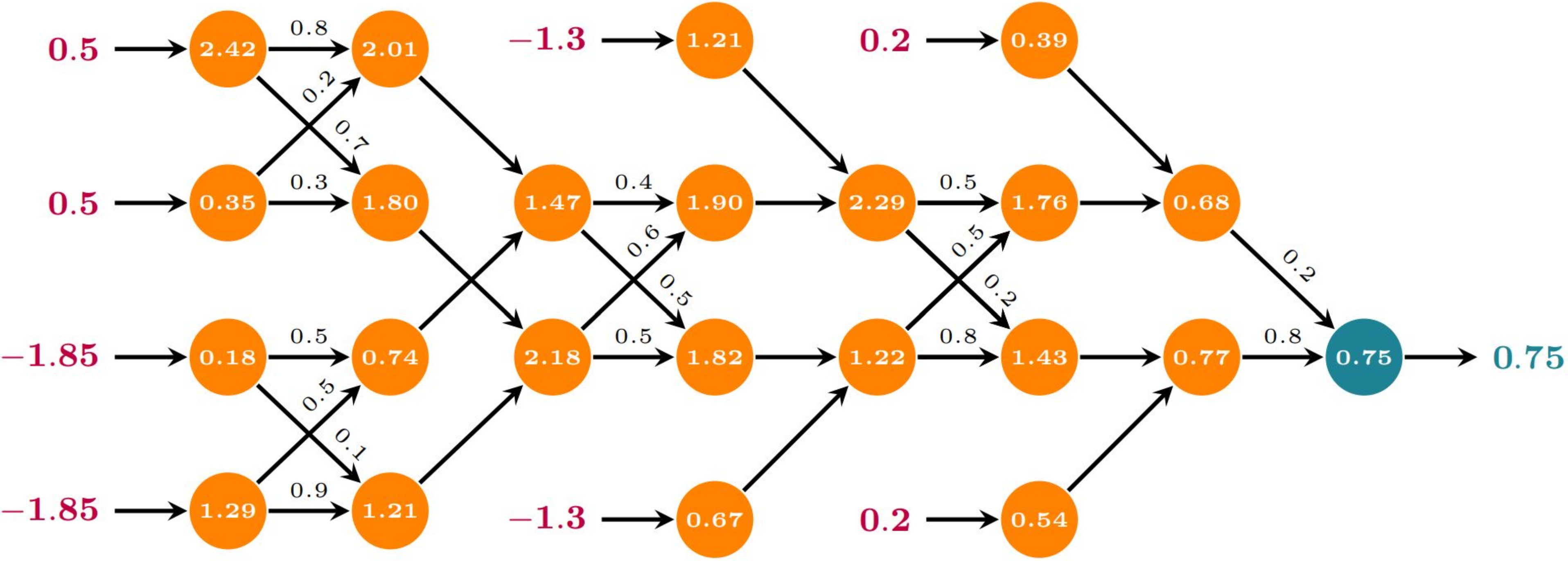
Likelihood

$$p(X_1 = \mathbf{-1.85}, X_2 = \mathbf{0.5}, X_3 = \mathbf{-1.3}, X_4 = \mathbf{0.2})$$



Likelihood

$$p(X_1 = \mathbf{-1.85}, X_2 = \mathbf{0.5}, X_3 = \mathbf{-1.3}, X_4 = \mathbf{0.2})$$



Smoothness + **decomposability** = **tractable MAR**

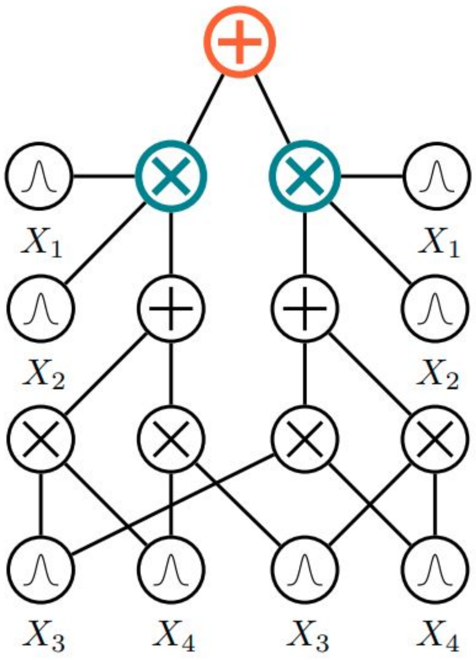
Smoothness + decomposability = tractable MAR

If $p(\mathbf{x}) = \sum_i w_i p_i(\mathbf{x})$, (***smoothness***):

$$\int p(\mathbf{x}) d\mathbf{x} = \int \sum_i w_i p_i(\mathbf{x}) d\mathbf{x} =$$

$$= \sum_i w_i \int p_i(\mathbf{x}) d\mathbf{x}$$

$\Rightarrow$ *integrals are "pushed down" to children*



[Darwiche & Marquis JAIR 2001, Poon & Domingos UAI11]

Smoothness + **decomposability** = **tractable MAR**

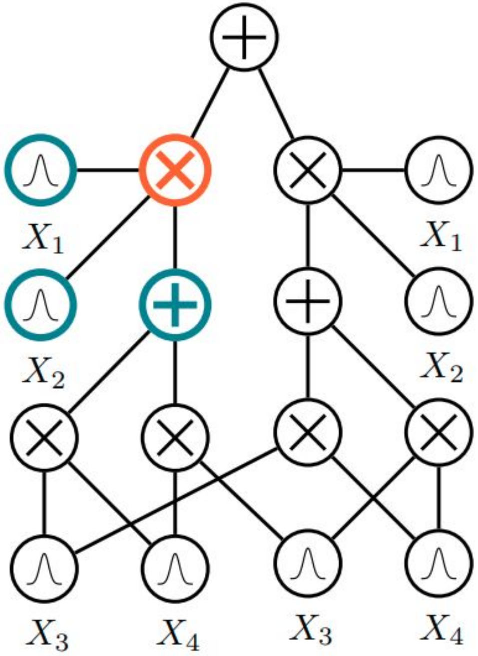
If $p(\mathbf{x}, \mathbf{y}, \mathbf{z}) = p(\mathbf{x})p(\mathbf{y})p(\mathbf{z})$, (***decomposability***):

$$\int\int\int p(\mathbf{x}, \mathbf{y}, \mathbf{z})d\mathbf{x}d\mathbf{y}d\mathbf{z} =$$

$$= \int\int\int p(\mathbf{x})p(\mathbf{y})p(\mathbf{z})d\mathbf{x}d\mathbf{y}d\mathbf{z} =$$

$$= \int p(\mathbf{x})d\mathbf{x}\int p(\mathbf{y})d\mathbf{y}\int p(\mathbf{z})d\mathbf{z}$$

$\Rightarrow$ *integrals decompose into easier ones*



Smoothness + decomposability = tractable MAR

Forward pass evaluation for MAR

$\Rightarrow$ *linear in circuit size!*

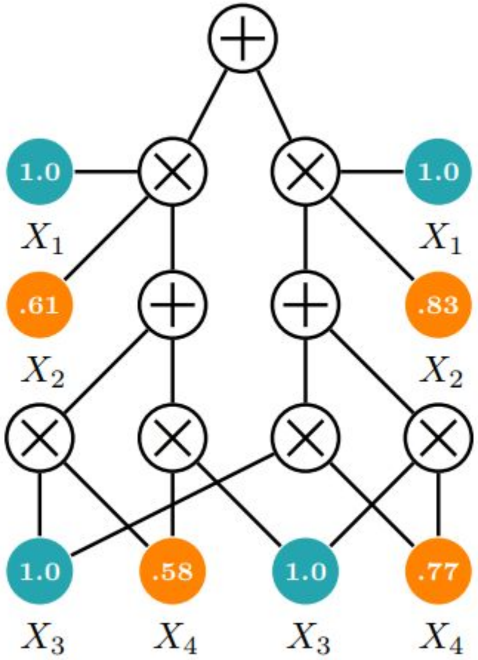
E.g. to compute $p(x_2, x_4)$:

- leafs over X_1 and X_3 output $Z_i = \int p(x_i) dx_i$

  $\Rightarrow$ *for normalized leaf distributions:* 1.0

- leafs over X_2 and X_4 output ***EVI***

- feedforward evaluation (bottom-up)



Smoothness + decomposability = tractable MAR

Forward pass evaluation for MAR

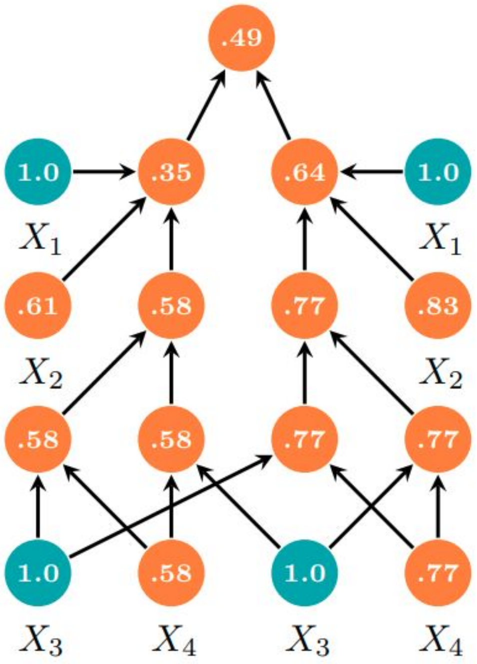
$\Rightarrow$ *linear in circuit size!*

E.g. to compute $p(x_2, x_4)$:

- leafs over X_1 and X_3 output $\boldsymbol{Z}_i = \int p(x_i)dx_i$

  $\Rightarrow$ *for normalized leaf distributions:* 1.0

- leafs over X_2 and X_4 output **EVI**
- feedforward evaluation (bottom-up)



Cute, but these models cannot compete?

	2008-2020
Tabular	😐
MNIST	😱
F-MNIST	😱
EMNIST-L	😱
CIFAR	😱
Imagenet32	😱
Imagenet64	😱

Cute, but these models cannot compete?

bpd	2008-2020	2020-2021
Tabular	😐	😊
MNIST	😱	😱 > 1.67
F-MNIST	😱	😱 > 4.29
EMNIST-L	😱	😱 > 2.73
CIFAR	😱	😱
Imagenet32	😱	😱
Imagenet64	😱	😱

Cute, but these models cannot compete?

	2008-2020	2020-2021	ICLR 22
Tabular	😐	😊	🍰
MNIST	😱	😱 > 1.67	1.20
F-MNIST	😱	😱 > 4.29	3.34
EMNIST-L	😱	😱 > 2.73	1.80
CIFAR	😱	😱	😱 > 5.50
Imagenet32	😱	😱	😱
Imagenet64	😱	😱	😱

Cute, but these models cannot compete?

	2008-2020	2020-2021	ICLR 22	NeurIPS 22
Tabular	😐	😊	🍰	🍰
MNIST	😱	😱 > 1.67	1.20	1.14
F-MNIST	😱	😱 > 4.29	3.34	3.27
EMNIST-L	😱	😱 > 2.73	1.80	1.58
CIFAR	😱	😱	😱 > 5.50	😱
Imagenet32	😱	😱	😱	😱
Imagenet64	😱	😱	😱	😱

	Discrete Flow	Hierarchical VAE	PixelVAE
MNIST	1.90	1.27	1.39
F-MNIST	3.47	3.28	3.66
EMNIST-L	1.95	1.84	2.26

Cute, but these models cannot compete?

	2008-2020	2020-2021	ICLR 22	NeurIPS 22	ICLR 23
Tabular	😐	😊	🍰	🍰	🍰
MNIST	😱	😱 > 1.67	1.20	1.14	🍰
F-MNIST	😱	😱 > 4.29	3.34	3.27	🍰
EMNIST-L	😱	😱 > 2.73	1.80	1.58	🍰
CIFAR	😱	😱	😱 > 5.50	😱	4.38
Imagenet32	😱	😱	😱	😱	4.39
Imagenet64	😱	😱	😱	😱	4.12

Cute, but these models cannot compete?

	2008-2020	2020-2021	ICLR 22	NeurIPS 22	ICLR 23	Today
Tabular	😐	😊	🍰	🍰	🍰	🍰
MNIST	😱	😱 > 1.67	1.20	1.14	🍰	🍰
F-MNIST	😱	😱 > 4.29	3.34	3.27	🍰	🍰
EMNIST-L	😱	😱 > 2.73	1.80	1.58	🍰	🍰
CIFAR	😱	😱	😱 > 5.50	😱	4.38	3.87
Imagenet32	😱	😱	😱	😱	4.39	4.06
Imagenet64	😱	😱	😱	😱	4.12	3.80

	Flow	Hierarchical VAE	Diffusion
CIFAR	3.35	3.08	2.65
Imagenet32	4.09	3.96	3.72
Imagenet64	3.81	-	3.40

How?

- The *better* bitter lesson:
  Scale up the fancy method!
  - Custom GPU kernels [AAAI21]

- General-purpose architecture [NeurIPS21, ICLR22]
- Pruning without losing likelihood [NeurIPS22]
- Latent variable distillation [ICLR23]
  - Expectation Maximization < Embeddings

Anji Liu, Honghua Zhang, Guy Van den Broeck. Scaling Up Probabilistic Circuits by Latent Variable Distillation, ICLR, 2023

Controlled generation is still challenging …

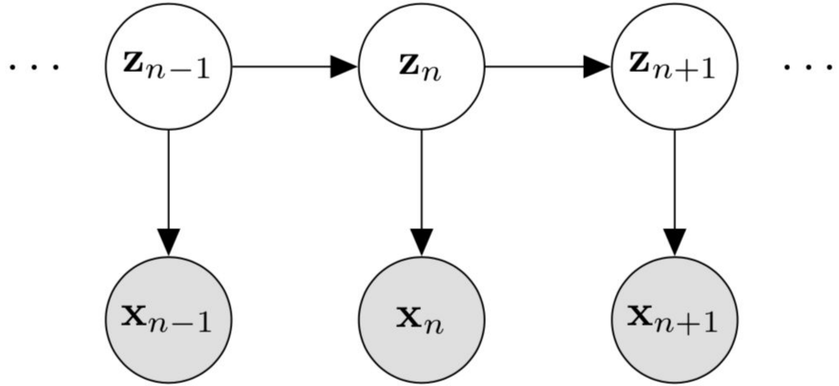
H

generate a sentence with "pan" as the third word and "vegetable" as the fifth word.

The chef used the pan to gently sauté the diced vegetable for their delicious stir-fry dish.

ChatGPT

Step 1: distill a PC that *approximates* the distribution of a LLM.



- Generate a *Probabilistic Circuit* architecture from a *Hidden Markov Model*
  - 50k emission tokens *x*
  - 4096 hidden states *z*
- Train on data sampled from GPT2-Large
- Same tricks as before (latent variable distillation)

Step 2: compute p(next-token | prefix, α) via PC

Dynamic programming in PyTorch using constraint α

Can be complex: many keywords, inflections, positions, …

CommonGen: a challenging constrained generation benchmark:

Method	*Quality* BLEU-4		*Constraint Satisfaction*	
Unsupervised	*test1*	*test2*	*test1*	*test2*
InsNet (Lu et al., 2022a)	18.7	-	**100.0**	
NeuroLogic (Lu et al., 2021)	-	24.7	-	<96.7
A*esque (Lu et al., 2022b)	-	28.6	-	<97.1
NADO (Meng et al., 2022)	26.2	-	<96.1	-
PC	27.5	-	**100.0**	**100.0**

Step 3: let LLM & PC control auto-regressive generation together

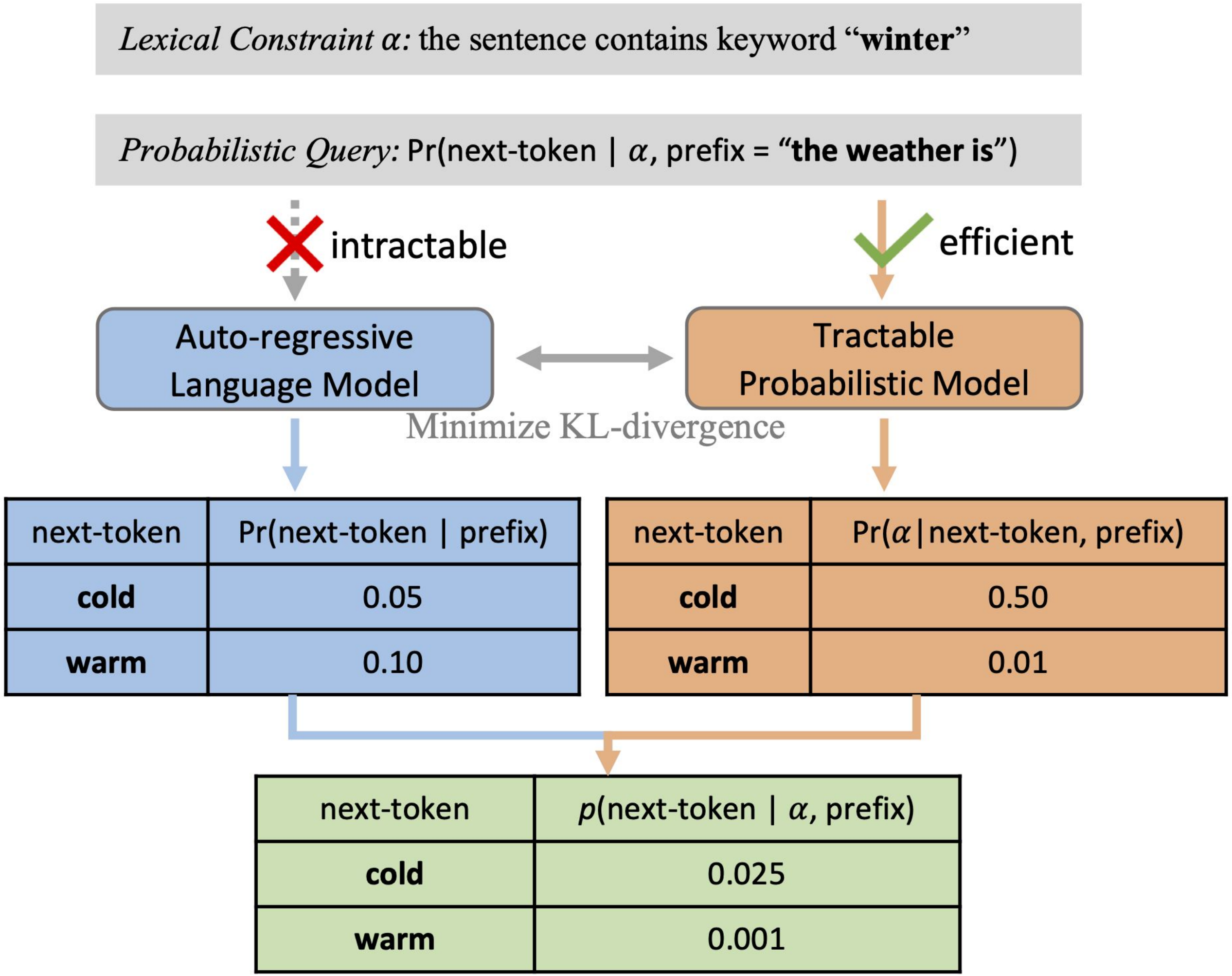
Require both fluency β and constraint α: $p_{\text{PC}}(x_{t+1} | x_{1:t}, \alpha, \beta)$

$$\propto p_{\text{PC}}(\alpha | x_{1:t+1}, \beta) \cdot p_{\text{PC}}(x_{t+1} | x_{1:t}, \beta)$$

$$\propto p_{\text{PC}}(\alpha | x_{1:t+1}) \cdot p_{\text{PC}}(x_{t+1} | x_{1:t}, \beta) \qquad \textit{(independence)}$$

$$\propto p_{\text{PC}}(\alpha | x_{1:t+1}) \cdot p_{\text{LLM}}(x_{t+1} | x_{1:t}) \qquad \textit{(sleight of hand)}$$

Method	*Quality* BLEU-4		*Constraint Satisfaction*	
Unsupervised	*test1*	*test2*	*test1*	*test2*
InsNet (Lu et al., 2022a)	18.7	-	**100.0**	
NeuroLogic (Lu et al., 2021)	-	24.7	-	<96.7
A*esque (Lu et al., 2022b)	-	28.6	-	<97.1
NADO (Meng et al., 2022)	26.2	-	<96.1	-
PC	27.5	-	**100.0**	**100.0**
PC & GPT2-Large	**29.9**	**29.4**	**100.0**	**100.0**


Lexical Constraint α: the sentence contains keyword "winter"
Probabilistic Query: Pr(next-token | α, prefix = "the weather is")
intractable
efficient
Auto-regressive Language Model
Tractable Probabilistic Model
Minimize KL-divergence
next-token
Pr(next-token | prefix)
cold
0.05
warm
0.10
next-token
Pr(α | next-token, prefix)
cold
0.50
warm
0.01
next-token
p(next-token | α, prefix)
cold
0.025
warm
0.001

CommonGen: a challenging constrained generation task

Method	*Quality* BLEU-4		*Constraint Satisfaction*	
Unsupervised	*test1*	*test2*	*test1*	*test2*
InsNet (Lu et al., 2022a)	18.7	-	**100.0**	
NeuroLogic (Lu et al., 2021)	-	24.7	-	<96.7
A*esque (Lu et al., 2022b)	-	28.6	-	<97.1
NADO (Meng et al., 2022)	26.2	-	<96.1	-
PC	27.5	-	**100.0**	**100.0**
PC & GPT2-Large	**29.9**	**29.4**	**100.0**	**100.0**
Supervised	*test1*	*test2*	*test1*	*test2*
NeuroLogic (Lu et al., 2021)	-	26.7	-	93.9
A*esque (Lu et al., 2022b)	-	28.2	-	97.9
NADO (Meng et al., 2022)	30.8	-	88.8	-
PC & GPT2-Large	**34.1**	**32.9**	**100.0**	**100.0**

State-of-the-art performance on the CommonGen dataset, beating baselines from various families of constrained generation techniques with a large margin. All baselines use GPT2-large as the base model.

What else can this do?

- Restrict the support of the learned distribution
  - *“if the image is classified as a dog, it must also be an animal”*

SotA
Hierarchical
Multi-Label
Classification

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	$\mathbf{81.06 \pm 0.68}$

What else can this do?

- Restrict the support of the learned distribution
    - *"if the image is classified as a dog, it must also be an animal"*
    - *"predict a sparse vector/subset"*

SotA Learning to Explain

Results for three aspects with *k* = 10

Method	Appearance		Palate		Taste	
	Test MSE	Precision	Test MSE	Precision	Test MSE	Precision
SIMPLE (Ours)	**2.35 ± 0.28**	**66.81 ± 7.56**	**2.68 ± 0.06**	**44.78 ± 2.75**	**2.11 ± 0.02**	**42.31 ± 0.61**
L2X (t = 0.1)	10.70 ± 4.82	30.02 ± 15.82	6.70 ± 0.63	**50.39 ± 13.58**	6.92 ± 1.61	32.23 ± 4.92
SoftSub (t = 0.5)	**2.48 ± 0.10**	52.86 ± 7.08	2.94 ± 0.08	39.17 ± 3.17	2.18 ± 0.10	**41.98 ± 1.42**
I-MLE (τ = 30)	**2.51 ± 0.05**	**65.47 ± 4.95**	2.96 ± 0.04	40.73 ± 3.15	2.38 ± 0.04	**41.38 ± 1.55**

Results for aspect Aroma, for *k* in {5, 10, 15}

Method	$k = 5$		$k = 10$		$k = 15$	
	Test MSE	Precision	Test MSE	Precision	Test MSE	Precision
SIMPLE (Ours)	**2.27 ± 0.05**	**57.30 ± 3.04**	**2.23 ± 0.03**	**47.17 ± 2.11**	3.20 ± 0.04	**53.18 ± 1.09**
L2X (t = 0.1)	5.75 ± 0.30	33.63 ± 6.91	6.68 ± 1.08	26.65 ± 9.39	7.71 ± 0.64	23.49 ± 10.93
SoftSub (t = 0.5)	2.57 ± 0.12	**54.06 ± 6.29**	2.67 ± 0.14	44.44 ± 2.27	**2.52 ± 0.07**	37.78 ± 1.71
I-MLE (τ = 30)	2.62 ± 0.05	**54.76 ± 2.50**	2.71 ± 0.10	**47.98 ± 2.26**	2.91 ± 0.18	39.56 ± 2.07

What else can this do?

- Restrict the support of the learned distribution
  - *"if the image is classified as a dog, it must also be an animal"*
  - *"predict a sparse vector/subset"*
  - Neurosymbolic AI

What else can this do?

- Restrict the support of the learned distribution
  - *"if the image is classified as a dog, it must also be an animal"*
  - *"predict a sparse vector/subset"*
  - Neurosymbolic AI
- Information-theoretic queries (Entropy, KLD)
- Marginal MAP inference
- Causal inference
- …

Thanks

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

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

Discussion

1. Exact likelihood vs. ELBO vs. implicit GAN objective.

   Does likelihood-tractability matter?

2. *Does tractability help (bias) or hurt (capacity) learning?*

3. Learn $p(\text{next-token}|\text{prefix})$ then run $p(\text{next-token}|\text{prefix}, \alpha)$

   vs. learn $q(\text{next-token}|\text{prefix}, \alpha)$ for $\alpha \sim p_{\text{task}}$

   When do we care?

4. Which task do you want a tractable generative model for?