UCLA Computer Science

STAR AI
RESEARCH LAB
UCLA

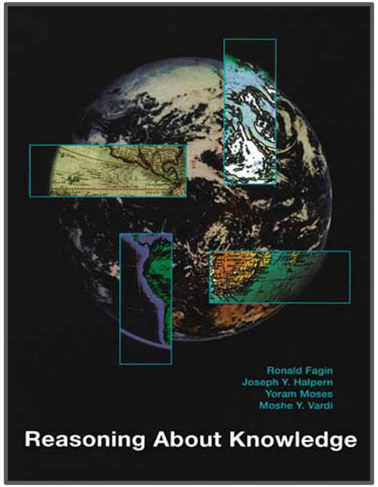
Symbolic Reasoning in the Age of Large Language Models

Guy Van den Broeck

National University of Singapore - Jan 20 2026

Reasoning with Symbolic AI

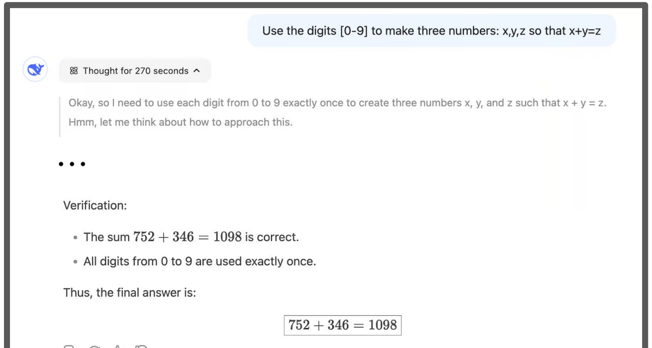
- Logic and probabilistic
- **Deductive** reasoning algorithms
- Correct on *all* problems
- Limited scope
- Intractable



30 years

Reasoning with Transformers

- Build chains of thought
- **Inductive** reasoning from data
- Correct on *many* problems
- Unlimited scope
- Tractable



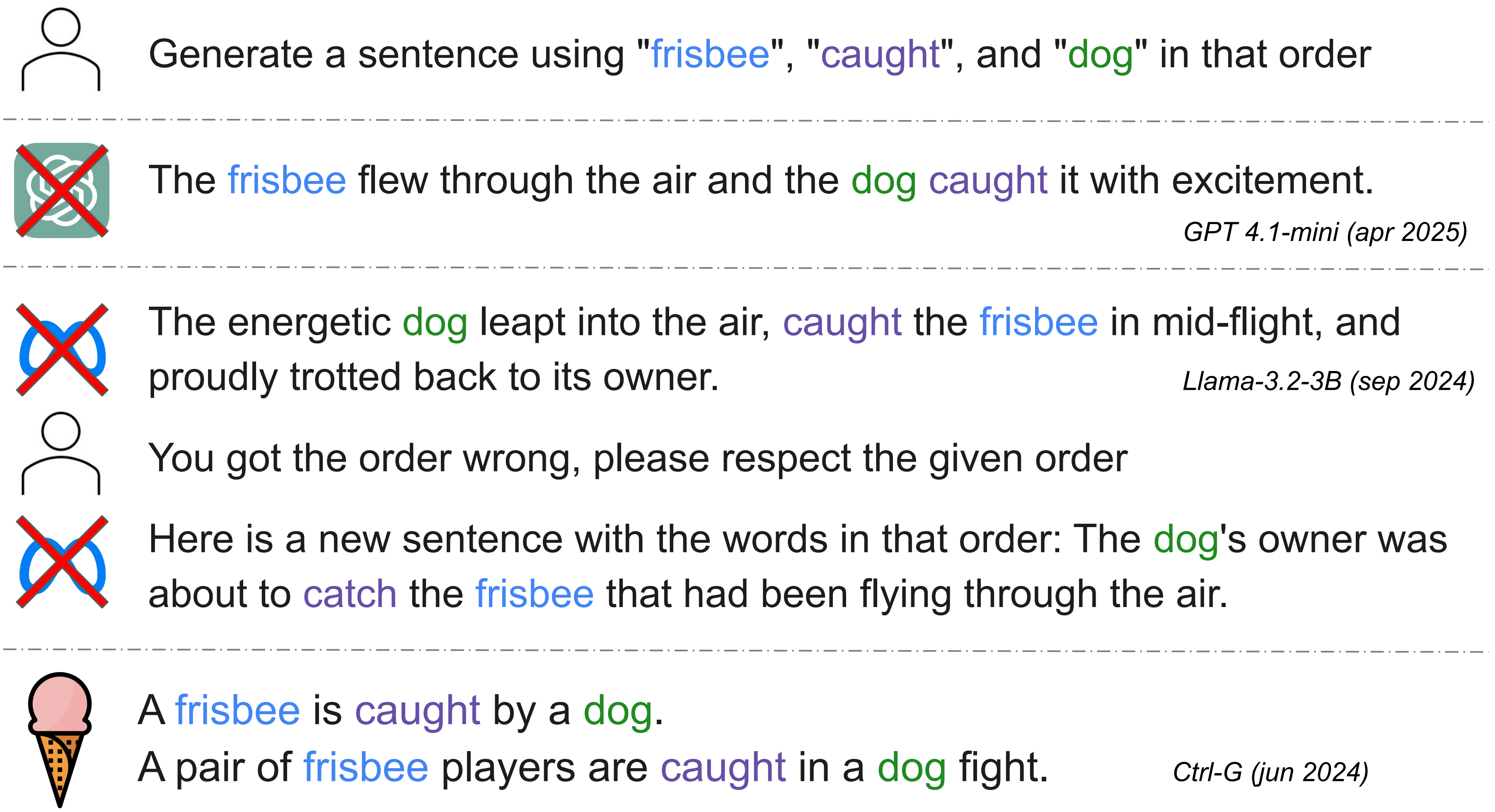
https://www.datacamp.com/blog/deepseek-r1-vs-v3

Questions for this talk:

1. Do deductive reasoning algorithms still have a purpose in the age of LLMs?
2. Where did reasoning algorithms go wrong? What should they look like today?
3. Can reasoning algorithms provide a path to language model alignment, safety?

Questions for this talk:

1. **Do deductive reasoning algorithms still have a purpose in the age of LLMs?**
2. Where did reasoning algorithms go wrong? What should they look like today?
3. Can reasoning algorithms provide a path to language model alignment, safety?


Generate a sentence using "frisbee", "caught", and "dog" in that order
The frisbee flew through the air and the dog caught it with excitement.
GPT 4.1-mini (apr 2025)
The energetic dog leapt into the air, caught the frisbee in mid-flight, and proudly trotted back to its owner.
Llama-3.2-3B (sep 2024)
You got the order wrong, please respect the given order
Here is a new sentence with the words in that order: The dog's owner was about to catch the frisbee that had been flying through the air.
A frisbee is caught by a dog.
A pair of frisbee players are caught in a dog fight.
Ctrl-G (jun 2024)

Reasoning about all Future Tokens: *Constraints*

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

Constrained Generation: $\Pr(x_{t+1} \mid \alpha, x_{1:t} = \text{"the weather is"})$

Lexical Constraint α: sentence contains keyword "winter"

Reasoning about all Future Tokens: *Constraints*

p(next-token | α, prefix)

Constrained Generation: $\Pr(x_{t+1} \mid \alpha, x_{1:t} =$ "the weather is"$)$

Lexical Constraint α: sentence contains keyword "winter"

$\propto$ p(next-token | prefix) $\cdot$ p(α | next-token, prefix)

Bayes' rule lets us reason backwards in time!

Reasoning about all Future Tokens: *Constraints*

p(next-token | α, prefix)

cold	0.025
warm	0.001

Constrained Generation: $\Pr(x_{t+1} \mid \alpha, x_{1:t} = \text{"the weather is"})$

Lexical Constraint α: sentence contains keyword "winter"

$\propto$ p(next-token | prefix) · p(α | next-token, prefix)

cold	0.05
warm	0.10

cold	0.50
warm	0.01

Reasoning about all Future Tokens: *Alignment*

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

Prefix: It's a pain …

Constraint α: non-toxic

Reasoning about all Future Tokens: *Alignment*

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

Prefix: It's a pain …

Constraint α: non-toxic

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

in	0.3
to	0.1

in →

the ass	0.3
the butt	0.15
the neck	0.05
	...

to →

deal with	0.2
handle	0.1
...	...

Reasoning about all Future Tokens: *Alignment*

p(next-token | α, prefix)

in	0.03
to	0.08

Prefix: It's a pain …

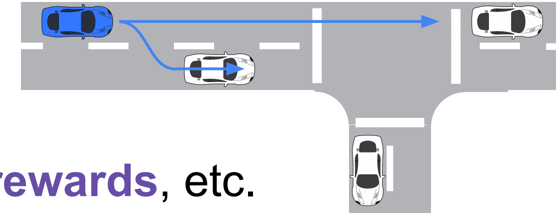
Constraint α: non-toxic

$\propto$ p(next-token | prefix) · p(α | next-token, prefix)

in	0.3
to	0.1

in	0.1
to	0.8

Reasoning about all Future Tokens: *Offline RL*

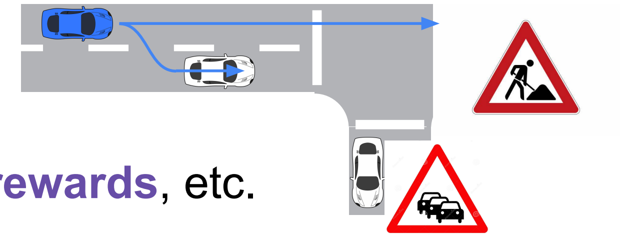


Training: model the joint distribution over **states**, **actions**, **rewards**, etc.

Inference: sample next **states** and **actions**

$\cdots$ state_{t-1} action_{t-1} R_{t-1} state_t action_t R_t state_{t+1} action_{t+1} R_{t+1} $\cdots$

Reasoning about all Future Tokens: *Offline RL*



Training: model the joint distribution over **states**, **actions**, **rewards**, etc.

Inference: sample next **states** and **actions**, as well as **constraints**.

$\cdots$ $\boxed{\text{state}_{t-1}}$ $\boxed{\text{action}_{t-1}}$ $\boxed{\text{R}_{t-1}}$ $\boxed{\text{state}_{t}}$ $\boxed{\text{action}_{t}}$ $\boxed{\text{Constraint } \alpha}$

Reward: $\sum_{t' \geq t} \boxed{\text{R}_{t'}} \geq \text{threshold}$

State: $\boxed{\text{state}_t} \in \boxed{\text{safe states}}$

Action: $\boxed{\text{action}_t} \in \boxed{\text{safe actions}}$

$$p(\text{action} \mid \alpha, \text{prefix}) \propto p(\text{action} \mid \text{prefix}) \cdot p(\alpha \mid \text{action}, \text{prefix})$$

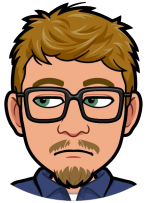
Reasoning about all Future Tokens

p_{lm}(next-token | α, prefix)

Using Bayes rule,

$\propto p_{lm}$(next-token | prefix) $\cdot$ ~~p_{lm}(α | next-token, prefix)~~

Intractable



Looking 20 tokens into the future amounts to more sentences than atoms in the universe….

Reasoning about all Future Tokens

Abusing Bayes rule,

p_{lm}(next-token | *α*, prefix)

$\propto$ p_{lm}(next-token | prefix) $\cdot$ $p_{circuit}$(*α* | next-token, prefix)

Use a tractable circuit model distilled from the transformer LLM…

A `digital twin' that can do symbolic reasoning

Reasoning about all Future Tokens: *Constraints*

p(next-token | *α*, prefix)

cold	0.025
warm	0.001

Constrained Generation: $\Pr(x_{t+1} \mid \alpha, x_{1:t} =$ "the weather is"$)$

Lexical Constraint α: sentence contains keyword "winter"

∝ *p*(next-token | prefix) · *p*(*α* | next-token, prefix)

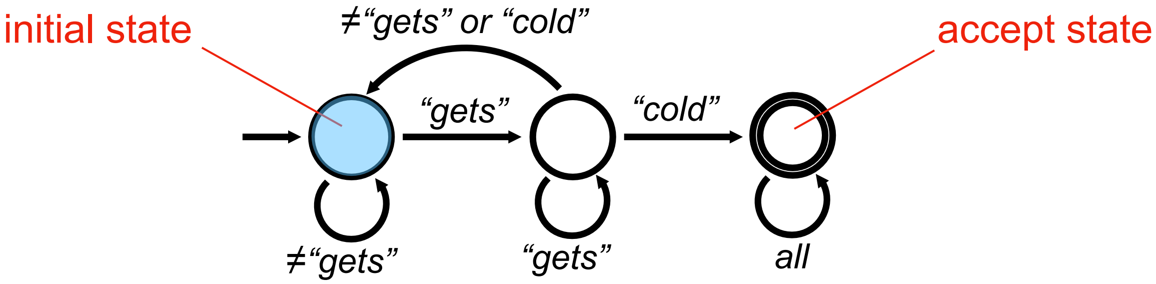
cold	0.05
warm	0.10

cold	0.50
warm	0.01

Representing Logical Constraints

*as a **deterministic finite automaton** (DFA)*

Example. Check if a string contains “gets cold”.

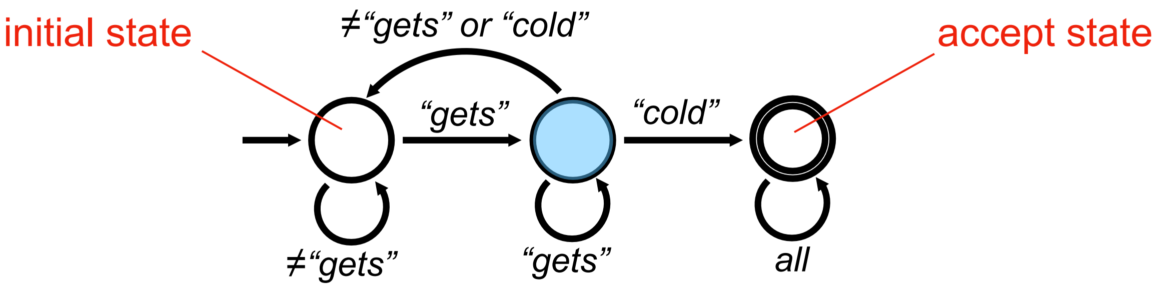


String: “The weather gets cold in the winter.”

Representing Logical Constraints

*as a **deterministic finite automaton** (DFA)*

Example. Check if a string contains “gets cold”.

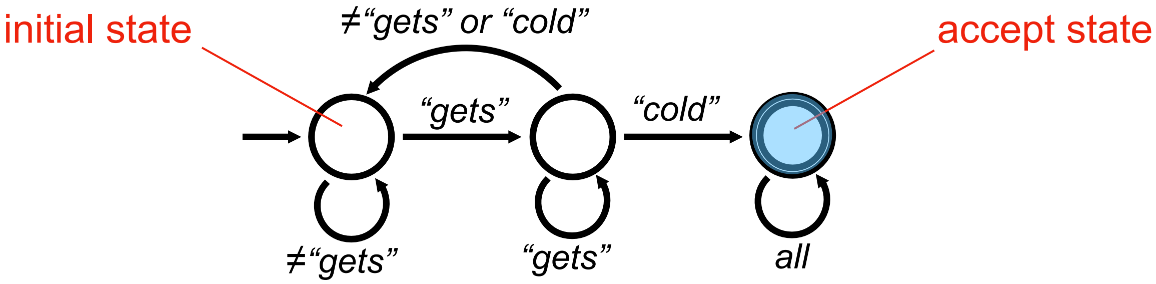


String: “The weather gets cold in the winter.”

Representing Logical Constraints

as a ***deterministic finite automaton*** *(DFA)*

Example. Check if a string contains “gets cold”.

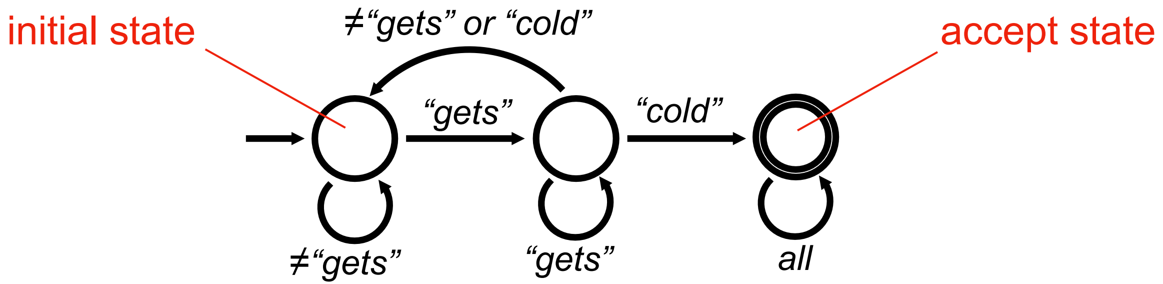


String: “The weather gets cold in the winter.”

Representing Logical Constraints

as a ***deterministic finite automaton*** *(DFA)*

Example. Check if a string contains “gets cold”.



Can represent:

Phrases/words must/must not appear

From a restricted vocabulary.

Exactly k times.

Must end a certain way

Any regex

Anything over fixed sequence lengths (BDD)

…

Reasoning about all Future Tokens: Constraints

p_{lm}(next-token | α, prefix)

Abusing Bayes rule,

$\propto$ p_{lm}(next-token | prefix) $\cdot$ $p_{circuit}$(α | next-token, prefix)

***Theorem**. Given*

1. *a <u>deterministic finite automata</u> constraint **α** with m edges and*
2. *a <u>probabilistic circuit</u> **p**(.) with h hidden states (representing a Hidden Markov Model) ,*

*computing **p**(**α** | $x_{1:t}$) over a sequence of n future tokens takes $O(nmh^2)$ time.*

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

CommonGen Benchmark

Generate a sentence using 3 to 5 concepts (keywords).

Input: snow drive car

α = ("car" ∨ "cars"…) ∧ ("drive" ∨ "drove"…) ∧

Reference 1: A car drives down a snow-covered road.

Reference 2: Two cars drove through the snow.

	BLEU-4		ROUGE-L		CIDEr		SPICE		Constraint	
	dev	*test*	*dev*	*test*	*dev*	*test*	*dev*	*test*	*dev*	*test*
supervised - base models trained with full supervision										
FUDGE	-	24.6	-	40.4	-	-	-	-	-	47.0%
A*esque	-	28.2	-	43.4	-	15.2	-	30.8	-	98.8%
NADO	30.8	-	44.4	-	16.1	-	32.0	-	88.8%	-
Ctrl-G	**35.1**	**34.4**	**46.7**	**46.4**	**17.4**	**17.6**	**32.7**	**33.3**	**100.0%**	**100.0%**
unsupervised - base models not trained with keywords as supervision										
A*esque	-	28.6	-	44.3	-	15.6	-	29.6	-	-
NADO	26.2	-	-	-	-	-	-	-	-	-
Ctrl-G	**32.1**	**31.5**	**45.2**	**44.8**	**16.0**	**16.2**	**30.8**	**31.2**	**100.0%**	**100.0%**

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In Arxiv*, 2024.

Interactive Text Editing

"First they've defeated a small squad [BLANK] are few humans left, and despite their magical power, their numbers are getting fewer."

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

Interactive Text Editing

User: given the following context, generate infilling text for [BLANK] using key phrases "alien mothership", "far from over"; generated text must contain 25 - 30 words.

"First they've defeated a small squad [BLANK] are few humans left, and despite their magical power, their numbers are getting fewer."

Ctrl-G

"First they've defeated a small squad of aliens, then a larger fleet of their ships. Eventually they've even managed to take down the alien mothership. But their problems are far from over. There are few humans left, and despite their magical power, their numbers are getting fewer."

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

Interactive Text Editing with key phrase (K) or length (L) constraints

CoAuthor 

	K	*L*	*K&L*
Quality			
TULU2	2.64	2.78	2.74
GPT3.5	2.22	2.27	2.31
GPT4	3.33	3.53	3.10
Ctrl-G	**3.56**	**3.73**	**3.59**

→ *How many stars by humans?*

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

Interactive Text Editing with key phrase (K) or length (L) constraints

CoAuthor

	K	*L*	*K&L*
Quality			
TULU2	2.64	2.78	2.74
GPT3.5	2.22	2.27	2.31
GPT4	3.33	3.53	3.10
Ctrl-G	**3.56**	**3.73**	**3.59**
Success			
TULU2	12%	20%	3%
GPT3.5	22%	54%	10%
GPT4	60%	20%	27%
Ctrl-G	**100%**	**100%**	**100%**

→ *How many stars by humans?*

→ *Follows instructions?*

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

Interactive Text Editing with key phrase (K) or length (L) constraints

CoAuthor 

	K	*L*	*K&L*
Quality			
TULU2	2.64	2.78	2.74
GPT3.5	2.22	2.27	2.31
GPT4	3.33	3.53	3.10
Ctrl-G	**3.56**	**3.73**	**3.59**
Success			
TULU2	12%	20%	3%
GPT3.5	22%	54%	10%
GPT4	60%	20%	27%
Ctrl-G	**100%**	**100%**	**100%**
Overall			
TULU2	7%	10%	1%
GPT3.5	0%	5%	2%
GPT4	41%	17%	14%
Ctrl-G	**76%**	**78%**	**82%**

→ *How many stars by humans?*

→ *Follows instructions?*

→ ★★★☆☆ *& Up* + *Follows instructions?*

→ **Ctrl-G** based on Llama2-7B **wipes the floor with GPT4**, which is a >100x bigger LLM

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

Grade School Math Benchmark

Question: *Kylar went to the store to buy glasses for his new apartment. One glass costs $5, but every second glass costs only 60% of the price. Kylar wants to buy 16 glasses. How much does he need to pay for them?*

Vanilla LLM Answer: The price of the 2nd glass is (16 / 2) * 60% = 8 dollars. So one pair of glasses costs 16 + 8 = 24 dollars. So the answer is 24.

Grade School Math Benchmark

***Question:** Kylar went to the store to buy glasses for his new apartment. One glass costs $5, but every second glass costs only 60% of the price. Kylar wants to buy 16 glasses. How much does he need to pay for them?*

Vanilla LLM Answer: The price of the 2nd glass is (16 / 2) * 60% = 8 dollars. So one pair of glasses costs 16 + 8 = 24 dollars. So the answer is 24.

Ctrl-G Answer: The second glass costs 5 * .6 = $3. So each set of two glasses actually costs 5 + 3 = $8. He wants 16 / 2 = 8 sets of two. That means he needs to pay 8 * 8 = $64. So the answer is 64.

Which constraint improves accuracy?

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

Grade School Math Benchmark

Question: *Kylar went to the store to buy glasses for his new apartment. One glass costs $5, but every second glass costs only 60% of the price. Kylar wants to buy 16 glasses. How much does he need to pay for them?*

Vanilla LLM Answer: The price of the 2nd glass is (16 / 2) * 60% = 8 dollars. So one pair of glasses costs 16 + 8 = 24 dollars. So the answer is 24.

Ctrl-G Answer: The second glass costs 5 * .6 = $3. So each set of two glasses actually costs 5 + 3 = $8. He wants 16 / 2 = 8 sets of two. That means he needs to pay 8 * 8 = $64. So the answer is 64.

Use all the numbers in the problem statement!

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In NeurIPS*, 2024.

Advantages of Ctrl-G:

1. Constraint α is guaranteed to be satisfied:
   if next-token makes α unsatisfiable, p_{lm}(next-token | α, prefix) = **0**.

   $$p_{lm}(\text{next-token} \mid \text{prefix}) \cdot p_{circuit}(\alpha \mid \text{next-token, prefix}) = \mathbf{0}$$

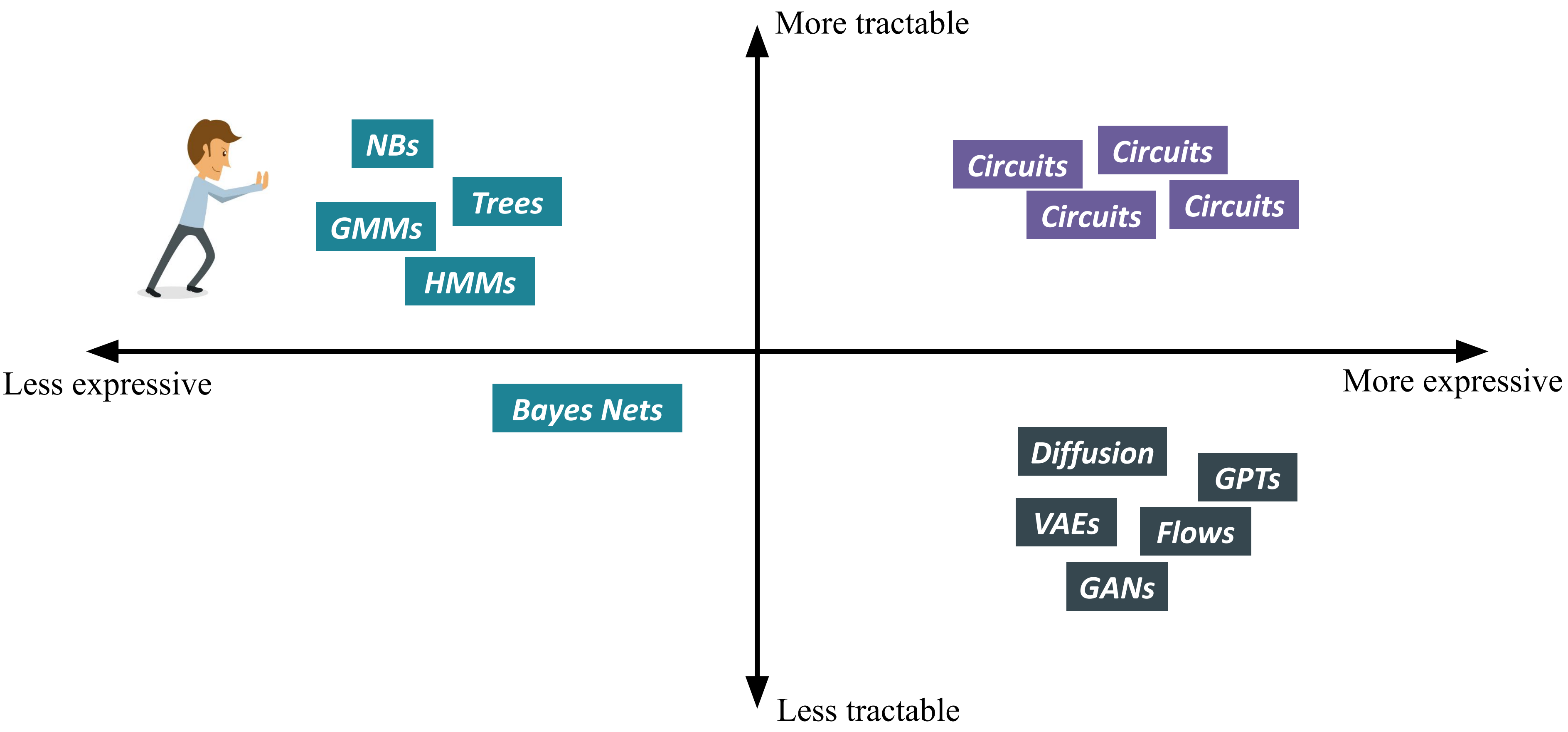
2. Generalizes well to unseen reasoning tasks, because all tasks are unseen :-)
   (training on a distribution over tasks is slow and brittle!)

3. Bayesian = goal-oriented (↔ structured generation tools)

You can control an intractable generative model using a generative model that is *tractable for symbolic reasoning*.

Questions for this talk:

1. Do deductive reasoning algorithms still have a purpose in the age of LLMs?
2. **Where did reasoning algorithms go wrong? What should they look like today?**
3. Can reasoning algorithms provide a path to language model alignment, safety?


More tractable
Less tractable
Less expressive
More expressive
NBs
Trees
GMMs
HMMs
Bayes Nets
Circuits
Circuits
Circuits
Circuits
Diffusion
GPTs
VAEs
Flows
GANs

Generative Models

polynomials model joint distributions

$$p(x_1, x_2, x_3) = .1x_1 + .05x_2 + .1x_1x_2 + .01x_3 - .07x_2x_3 + .02x_1x_3 - .14x_1x_2x_3 + .05$$

X_1	X_2	X_3	p
0	0	0	0.05
1	0	0	0.15
0	1	0	0.1
1	1	0	0.3
0	0	1	0.06
1	0	1	0.18
0	1	1	0.04
1	1	1	0.12

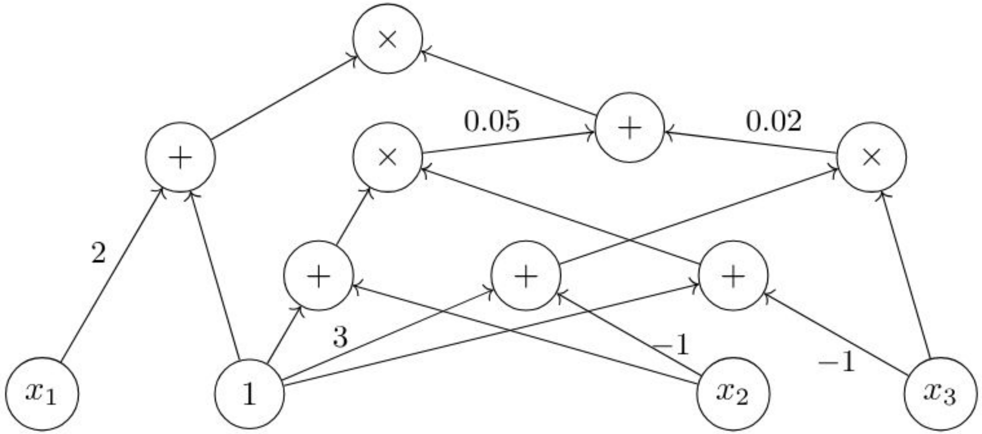
Oliver Broadrick, Sanyam Agarwal, Guy Van den Broeck and Markus Bläser. The Limits of Tractable Marginalization, 2025.

Deep Generative Models

circuit polynomials model joint distributions compactly
(and can have billions of trainable parameters)

$$p(x_1, x_2, x_3) = .1x_1 + .05x_2 + .1x_1x_2 + .01x_3 - .07x_2x_3 + .02x_1x_3 - .14x_1x_2x_3 + .05$$

X_1	X_2	X_3	p
0	0	0	0.05
1	0	0	0.15
0	1	0	0.1
1	1	0	0.3
0	0	1	0.06
1	0	1	0.18
0	1	1	0.04
1	1	1	0.12



Oliver Broadrick, Sanyam Agarwal, Guy Van den Broeck and Markus Bläser. The Limits of Tractable Marginalization, 2025.

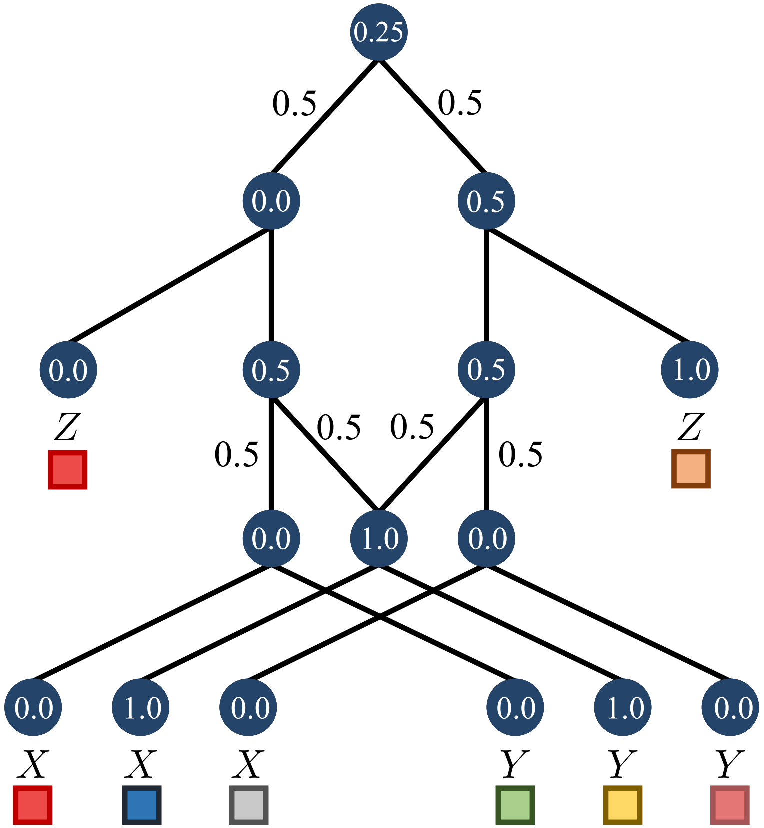
Compute Likelihood

Compute $p(x = \square, y = \square, z = \square) = 0.25$

- Readout likelihood from the **output node**.

- Compute the likelihood of every **sum/product node**.

- Compute the likelihood of every **input node**.



Probabilistic Reasoning Task

Marginal inference:

X_1	X_2	Pr
0	0	.1
0	1	.2
1	0	.3
1	1	.4

$$\begin{aligned}\Pr[X_1 = 1] &= \Pr[X_1 = 1, X_2 = 0] + \Pr[X_1 = 1, X_2 = 1]\\ &= 0.3 + 0.4\\ &= 0.7\end{aligned}$$

Application: Ctrl-G

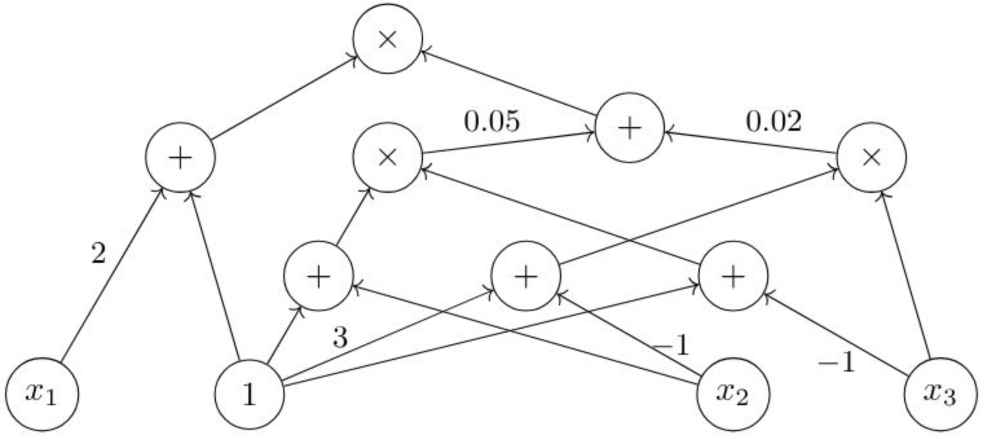
$p_{circuit}(\alpha \mid \text{next-token, prefix})$ is summing over all future text

Deep Generative Models

circuit polynomials model joint distributions compactly
(and can have billions of trainable parameters)

$$p(x_1, x_2, x_3) = .1x_1 + .05x_2 + .1x_1x_2 + .01x_3 - .07x_2x_3 + .02x_1x_3 - .14x_1x_2x_3 + .05$$

X_1	X_2	X_3	p
0	0	0	0.05
1	0	0	0.15
0	1	0	0.1
1	1	0	0.3
0	0	1	0.06
1	0	1	0.18
0	1	1	0.04
1	1	1	0.12



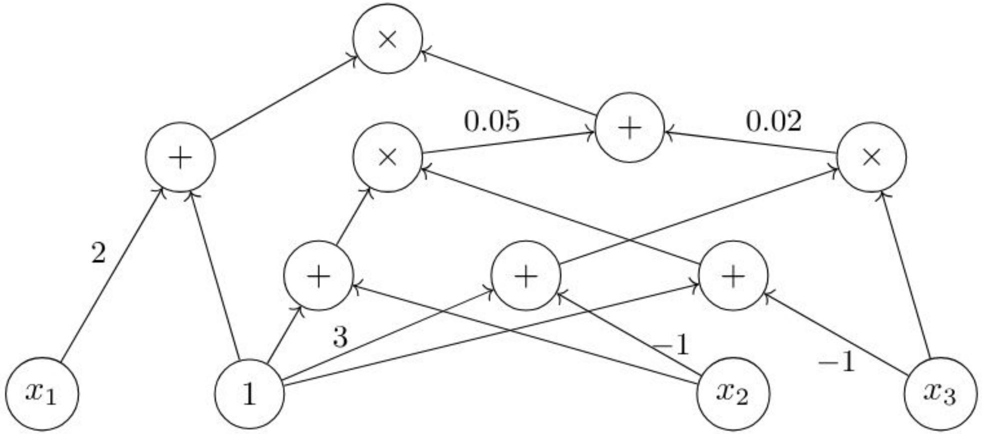
Oliver Broadrick, Sanyam Agarwal, Guy Van den Broeck and Markus Bläser. The Limits of Tractable Marginalization, 2025.

Tractable Deep Generative Models

Multilinear circuit polynomials model joint distributions compactly *and* allow efficient probabilistic reasoning (marginalization)

$$p(x_1, x_2, x_3) = .1x_1 + .05x_2 + .1x_1x_2 + .01x_3 - .07x_2x_3 + .02x_1x_3 - .14x_1x_2x_3 + .05$$

X_1	X_2	X_3	p
0	0	0	0.05
1	0	0	0.15
0	1	0	0.1
1	1	0	0.3
0	0	1	0.06
1	0	1	0.18
0	1	1	0.04
1	1	1	0.12

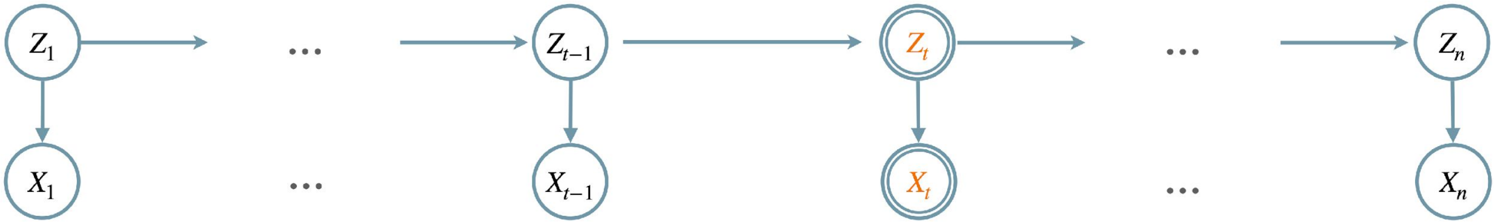


Oliver Broadrick, Sanyam Agarwal, Guy Van den Broeck and Markus Bläser. The Limits of Tractable Marginalization, 2025.

Probabilistic Circuit Language Model

How did we train a probabilistic circuit to solve Ctrl-G?

Keep it simple… just a classic **Hidden Markov Model** (HMM) with 32,768 hidden states and 2 billion parameters… on the GPU



***Theorem**. Given a DFA constraint α with m edges and an HMM p(x) with h hidden states, computing $p(\alpha \mid x_{1:t+1})$ over a sequence of n tokens takes $O(nmh^2)$ time.*

Honghua Zhang, Po-Nien Kung, Masahiro Yoshida, Guy Van den Broeck and Nanyun Peng. Adaptable Logical Control for Large Language Models, *In Arxiv*, 2024.

An Open-Source Package: PyJuice

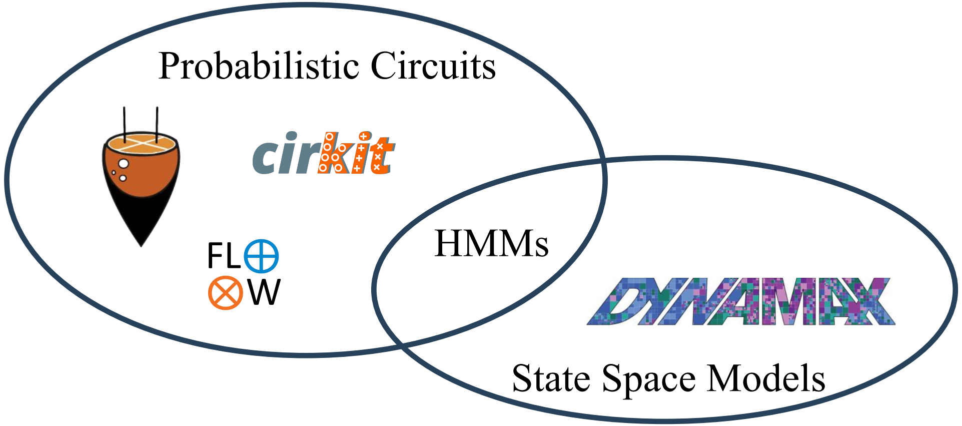
Runtime (in seconds) for training on **60K** samples

	PD (Poon & Domingos, 2011)				
# nodes	172K	344K	688K	1.38M	2.06M
# edges	15.6M	56.3M	213M	829M	2.03B
SPFlow	>25000	>25000	>25000	>25000	>25000
EiNet	34.2±0.0	88.7±0.2	456.1±2.3	1534.7±0.5	OOM
Juice.jl	12.6±0.5	37.0±1.7	141.7±6.9	OOM	OOM
PyJuice	**2.0**±0.0	**5.3**±0.0	**15.4**±0.0	**57.1**±0.2	**203.7**±0.1
	RAT-SPN (Peharz et al., 2020b)				
# nodes	58K	116K	232K	465K	930K
# edges	616K	2.2M	8.6M	33.4M	132M
SPFlow	6372.1±4.2	>25000	>25000	>25000	>25000
EiNets	38.5±0.0	83.5±0.0	193.5±0.1	500.6±0.2	2445.1±2.6
Juice.jl	6.0±0.3	9.4±0.3	25.5±2.4	84.0±4.0	375.1±3.4
PyJuice	**0.6**±0.0	**0.9**±0.1	**1.6**±0.0	**5.8**±0.1	**13.8**±0.0
	HCLT (Liu & Van den Broeck, 2021)				
# nodes	89K	178K	355K	710K	1.42M
# edges	2.56M	10.1M	39.9M	159M	633M
SPFlow	22955.6±18.4	>25000	>25000	>25000	>25000
EiNet	52.5±0.3	77.4±0.4	233.5±2.8	1170.7±8.9	5654.3±17.4
Juice.jl	4.7±0.2	6.4±0.5	12.4±1.3	41.1±0.1	143.2±5.1
PyJuice	**0.8**±0.0	**1.3**±0.0	**2.6**±0.0	**8.8**±0.0	**24.9**±0.1
	HMM (Rabiner & Juang, 1986)				
# nodes	33K	66K	130K	259K	388K
# edges	8.16M	32.6M	130M	520M	1.17B
Dynamax	111.3±0.4	441.2±3.9	934.7±6.3	2130.5±19.5	4039.8±38.3
Juice.jl	4.6±0.1	18.8±0.1	91.6±0.1	OOM	OOM
PyJuice	**0.6**±0.0	**1.0**±0.0	**2.9**±0.1	**10.1**±0.2	**39.9**±0.1

https://github.com/Tractables/pyjuice

- Orders of magnitude **faster**!
- Extremely **scalable**!

Custom data structure + CUDA kernels



FL⊕ ⊗W by Cambridge, TU Darmstadt, Max-Planck-Institute et al.

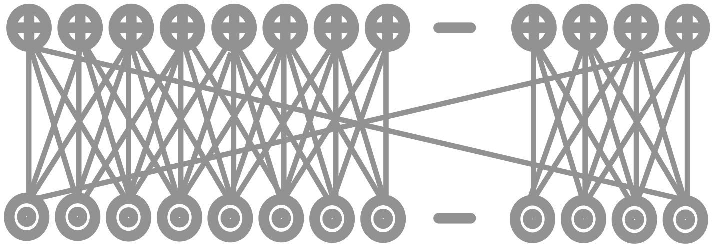
cirkit by Edinburgh, EPFL et al.

DYNAMAX by Google Deepmind et al.

Anji Liu, Kareem Ahmed and Guy Van den Broeck. Scaling Tractable Probabilistic Circuits: A Systems Perspective, *In Proceedings of the 41th International Conference on Machine Learning (ICML)*, 2024.

Scaling Up Probabilistic Circuits

Linear Layers



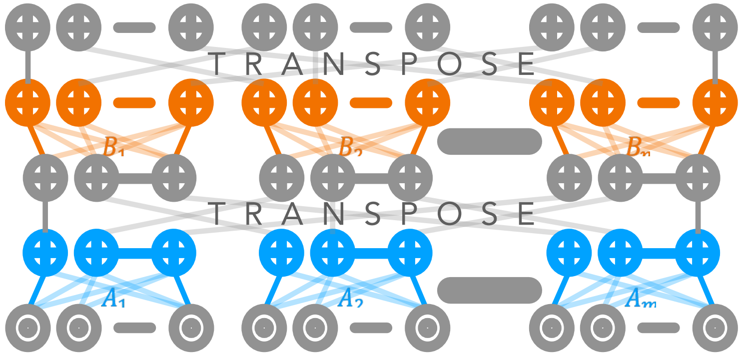
d nodes

$O(d^2)$ edges

$$y_{ij} = \sum_{kl} A_{ijkl} x_{kl}$$

Dense Matrices

e.g. a model w/ just 250K nodes requires 69B parameters (memory + time)…



d nodes

$O(d^{3/2})$ edges

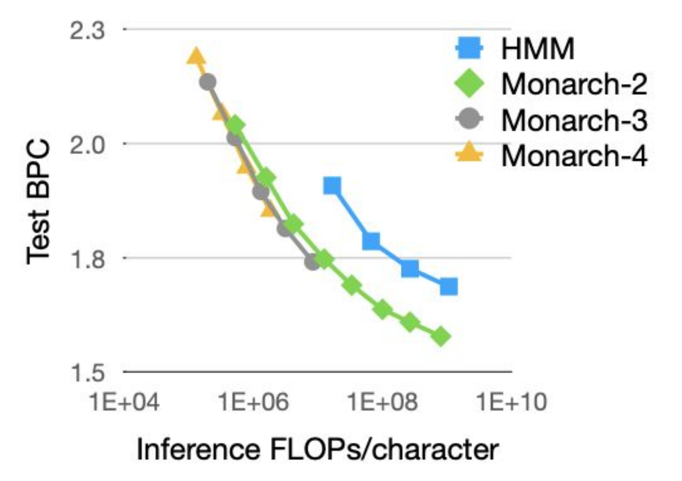
$$y_{ij} = \sum_{kl} B_{ijk} A_{jkl} x_{kl}$$

Monarch Matrices

… now just 134M parameters required!

Honghua Zhang, Meihua Dang, Benjie Wang, Stefano Ermon, Nanyun Peng and Guy Van den Broeck. Scaling Probabilistic Circuits via Monarch Matrices, *In Proceedings of the 42th International Conference on Machine Learning (ICML)*, 2025.

Scaling Up Probabilistic Circuits



Type	Model	BPC (↓)	Time (s) (↓)
Flow	IAF/SCF	1.88	0.04
Flow	Argmax Coup Flow	1.80	0.40
Diffusion	D3PM Uniform	$\leq$ 1.61	3.60
Diffusion	SEDD Uniform	$\leq$ 1.47	-
PC	SparsePC	2.60	-
PC	NPC^2	3.17	-
PC	HMM	1.69	0.006
PC	Monarch-HMM	**1.57**	0.017

Text8 Character-Level Language Modelling
Roughly on par with Flow and Diffusion models

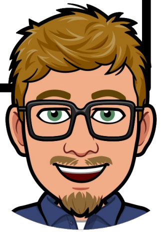
Honghua Zhang, Meihua Dang, Benjie Wang, Stefano Ermon, Nanyun Peng and Guy Van den Broeck. Scaling Probabilistic Circuits via Monarch Matrices, *In Proceedings of the 42th International Conference on Machine Learning (ICML)*, 2025.

You Tricked Us

You promised us reasoning algorithms…

… and all we got was another lousy feedforward neural network!

***Theorem**. If there exists a polynomial time (real RAM) **algorithm** that computes (virtual evidence) marginal probabilities for a class of distributions, then there exist **poly-size circuits** for their **multilinear** polynomials.*



Oliver Broadrick, Sanyam Agarwal, Guy Van den Broeck and Markus Bläser. The Limits of Tractable Marginalization, 2025.

Questions for this talk:

1. Do deductive reasoning algorithms still have a purpose in the age of LLMs?
2. Where did reasoning algorithms go wrong? What should they look like today?
3. **Can reasoning algorithms provide a path to language model alignment, safety?**

Reasoning about all Future Tokens: *Alignment*

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

in	0.03
to	0.08

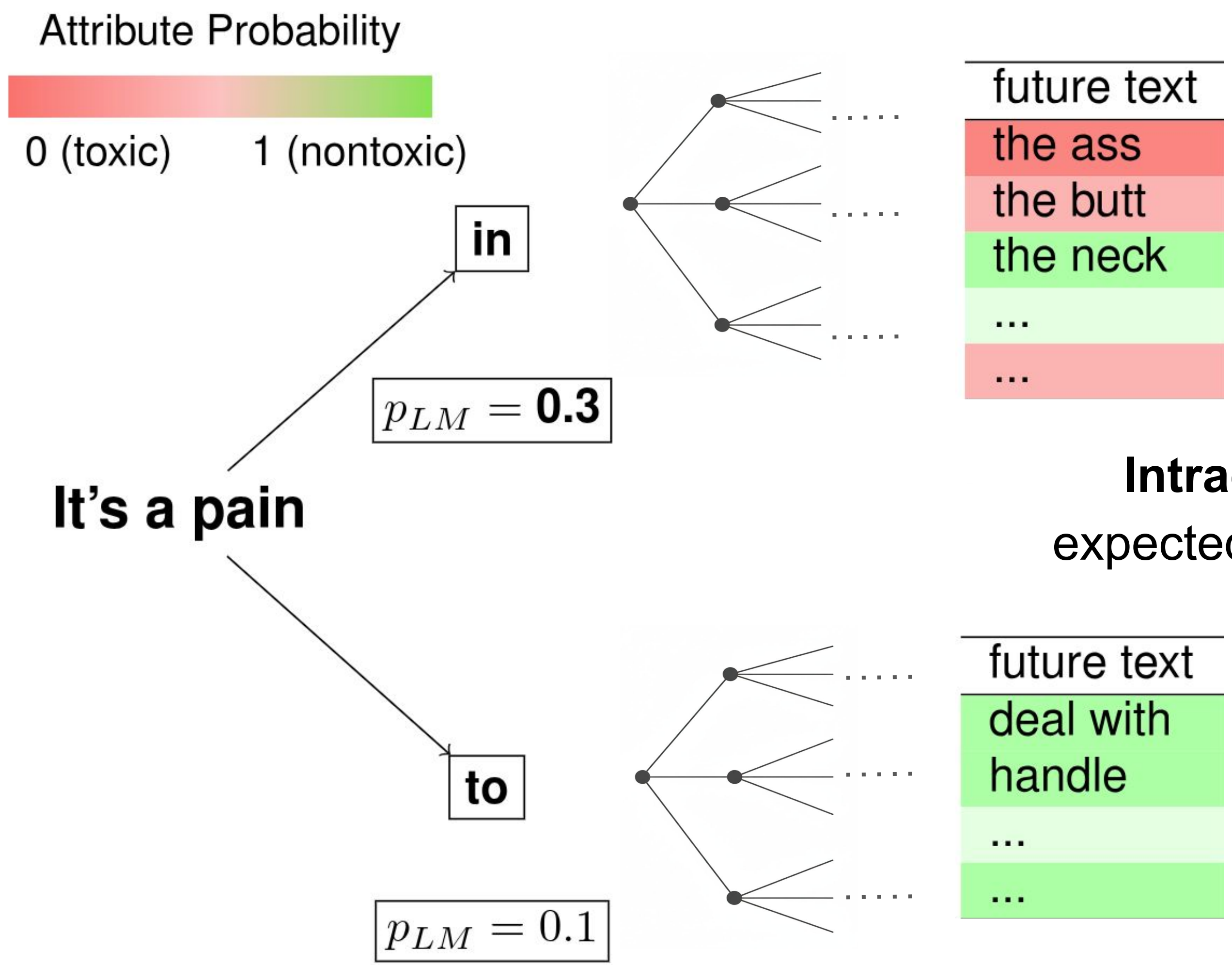
Prefix: It's a pain …

Constraint α: non-toxic

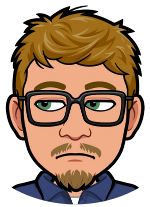
$\propto p(\text{next-token} \mid \text{prefix}) \cdot p(\alpha \mid \text{next-token, prefix})$

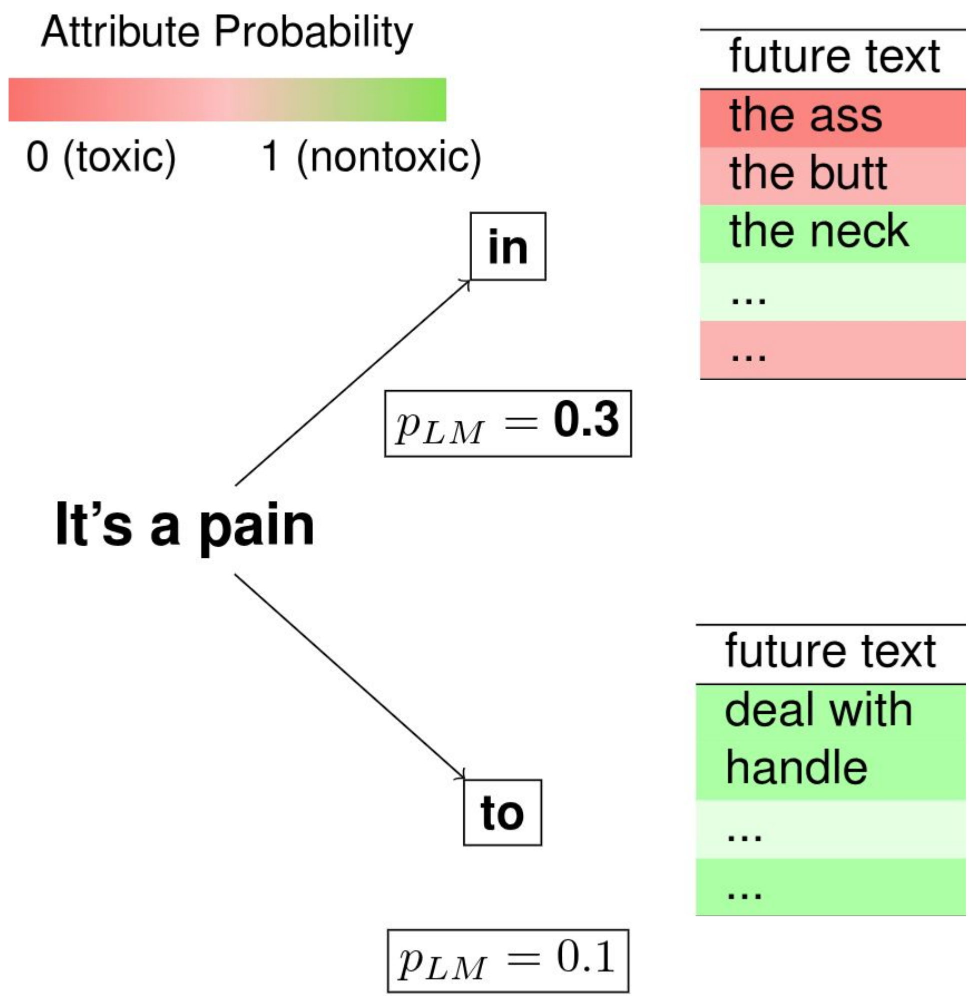
in	0.3
to	0.1

in	0.1
to	0.8

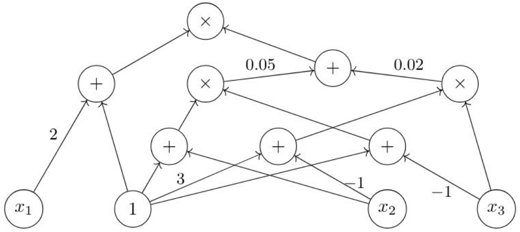


Intractable to know expected future toxicity





Model LLM continuations with *tractable probabilistic circuit*

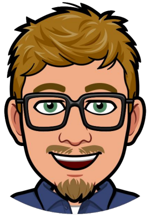


+

Model goal attribute with *log-linear classifier*

=

Efficient Expected Attribute Probability!



Gwen Yidou Weng, Benjie Wang and Guy Van den Broeck.
TRACE Back from the Future: A Probabilistic Reasoning Approach to Controllable Language Generation, 2025

Attribute Probability

0 (toxic) 1 (nontoxic)

It’s a pain

in

$p_{LM} = \mathbf{0.3}$

future text
the ass
the butt
the neck
...
...

$EAP = 0.1$

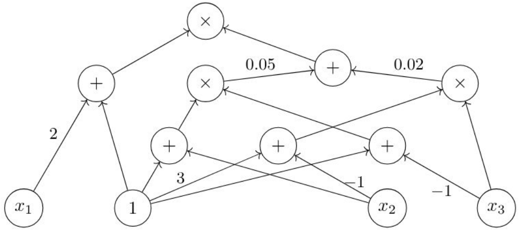
to

$p_{LM} = 0.1$

future text
deal with handle
...
...

$EAP = 0.8$

Model LLM continuations with *tractable probabilistic circuit*

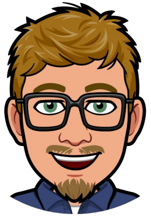


+

Model goal attribute with *log-linear classifier*

=

Efficient Expected Attribute Probability!



Gwen Yidou Weng, Benjie Wang and Guy Van den Broeck.
TRACE Back from the Future: A Probabilistic Reasoning Approach to Controllable Language Generation, 2025

Attribute Probability

0 (toxic) 1 (nontoxic)

It's a pain

in

$p_{LM} = \mathbf{0.3}$

future text
the ass
the butt
the neck
...
...

$\mathit{EAP} = 0.1 \quad = \boxed{p_{\mathit{TRACE}} \propto 0.03}$

to

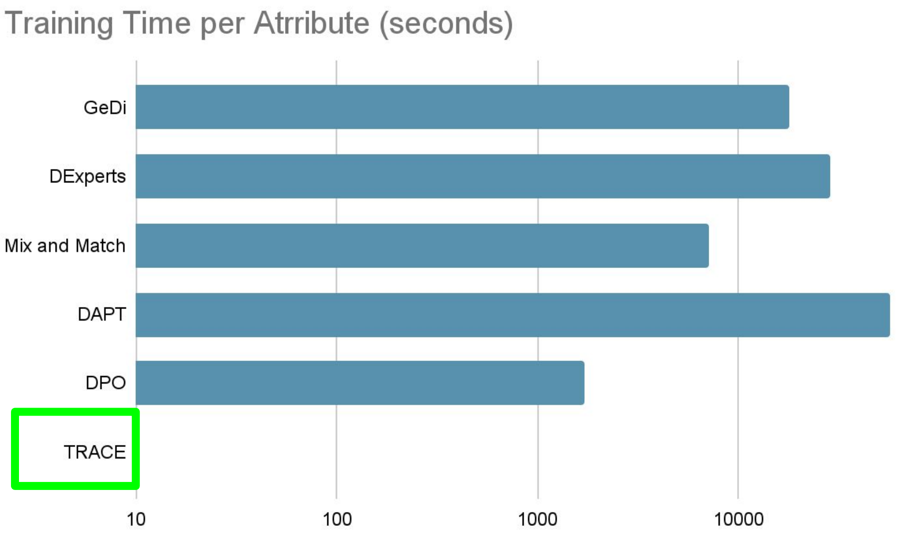
$p_{LM} = 0.1$

future text
deal with
handle
...
...

$\mathit{EAP} = 0.8 \quad = \boxed{p_{\mathit{TRACE}} \propto \mathbf{0.08}}$

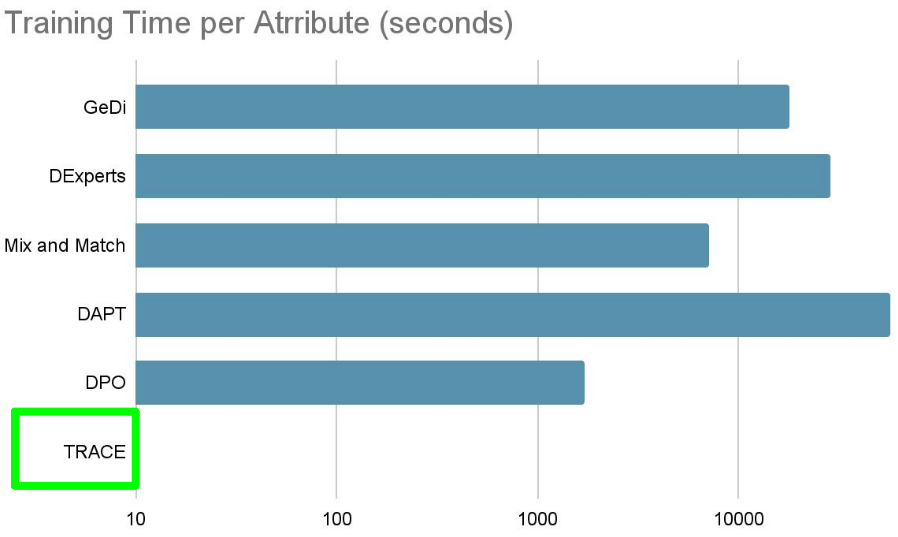
TRACE is Blazingly Fast

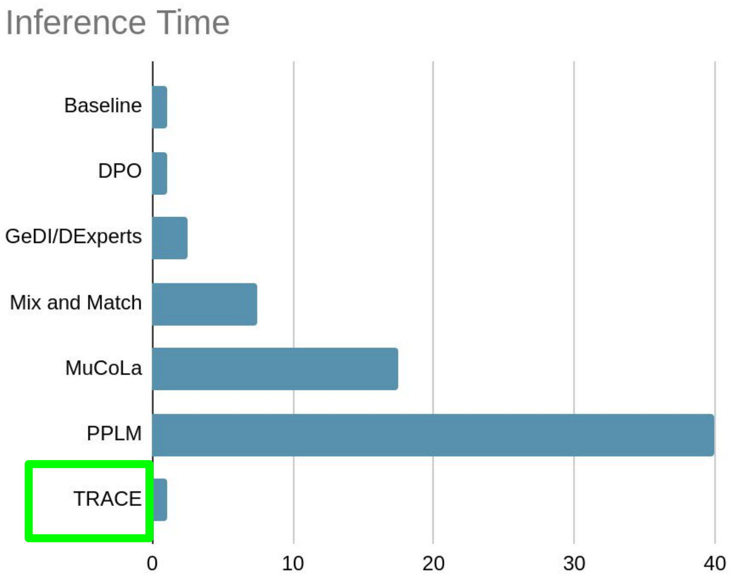
Given a language model, and its tractable twin,
train log-linear attribute classifier



TRACE is Blazingly Fast

Given a language model, and its tractable twin,
train log-linear attribute classifier,
then use Bayesian logits at decoding time





Gwen Yidou Weng, Benjie Wang and Guy Van den Broeck. TRACE Back from the Future: A Probabilistic Reasoning Approach to Controllable Language Generation, ICML 2025

State-of-the-art LLM Detoxification

Model	Toxicity (↓) avg. max.	Toxicity (↓) prob.	Approach Type
GPT-2 Large Results			
GPT2	0.385	0.254	Baseline
DAPT[(1)]	0.428	0.360	Finetuning
GeDi[(2)]	0.363	0.217	Decoding (Trained Guide)
FUDGE[(3)]	0.302	0.371	Decoding (Trained Guide)
DExperts[(4)]	0.314	0.128	Decoding (Trained Guide)
PPLM[(5)]	0.520	0.518	Decoding (Logit Control)
MuCoLa[(6)]	0.308	0.088	Decoding (Sampling)
PPO[(7)]	0.218	0.044	RL
Quark[(8)]	0.196	0.035	RL
DPO[(9)]	0.180	0.026	RL
TRACE	**0.163**	**0.016**	Decoding (HMM Reasoning)
Gemma-2B Results			
Gemma-2B	0.359	0.23	Baseline
DPO[(9)]	0.222	0.06	RL
TRACE	**0.189**	**0.02**	Decoding (HMM Reasoning)

Gwen Yidou Weng, Benjie Wang and Guy Van den Broeck. TRACE Back from the Future: A Probabilistic Reasoning Approach to Controllable Language Generation, ICML 2025

….but…
it’s easy to be non-toxic
by reusing
the same bland response…

State-of-the-art LLM Detoxi

Model	Toxicity (↓)		Diversity (↑)		
	avg. max.	prob.	dist-2	dist-3	
GPT-2 Large Results					
GPT2	0.385	0.254	0.87	0.86	
DAPT[1]	0.428	0.360	0.84	0.84	
GeDi[2]	0.363	0.217	0.84	0.83	Decoding (Trained Guide)
FUDGE[3]	0.302	0.371	0.78	0.82	Decoding (Trained Guide)
DExperts[4]	0.314	0.128	0.84	0.84	Decoding (Trained Guide)
PPLM[5]	0.520	0.518	0.86	0.86	Decoding (Logit Control)
MuCoLa[6]	0.308	0.088	0.82	0.83	Decoding (Sampling)
PPO[7]	0.218	0.044	0.80	0.84	RL
Quark[8]	0.196	0.035	0.80	0.84	RL
DPO[9]	0.180	0.026	0.76	0.78	RL
TRACE	**0.163**	**0.016**	0.85	0.85	Decoding (HMM Reasoning)
Gemma-2B Results					
Gemma-2B	0.359	0.23	0.86	0.85	Baseline
DPO[9]	0.222	0.06	0.74	0.77	RL
TRACE	**0.189**	**0.02**	**0.86**	**0.85**	Decoding (HMM Reasoning)

Method	**Entropy (↑)**
GPT2-large	52.06
DPO	39.52
TRACE	52.54

Gwen Yidou Weng, Benjie Wang and Guy Van den Broeck. TRACE Back from the Future: A Probabilistic Reasoning Approach to Controllable Language Generation, ICML 2025

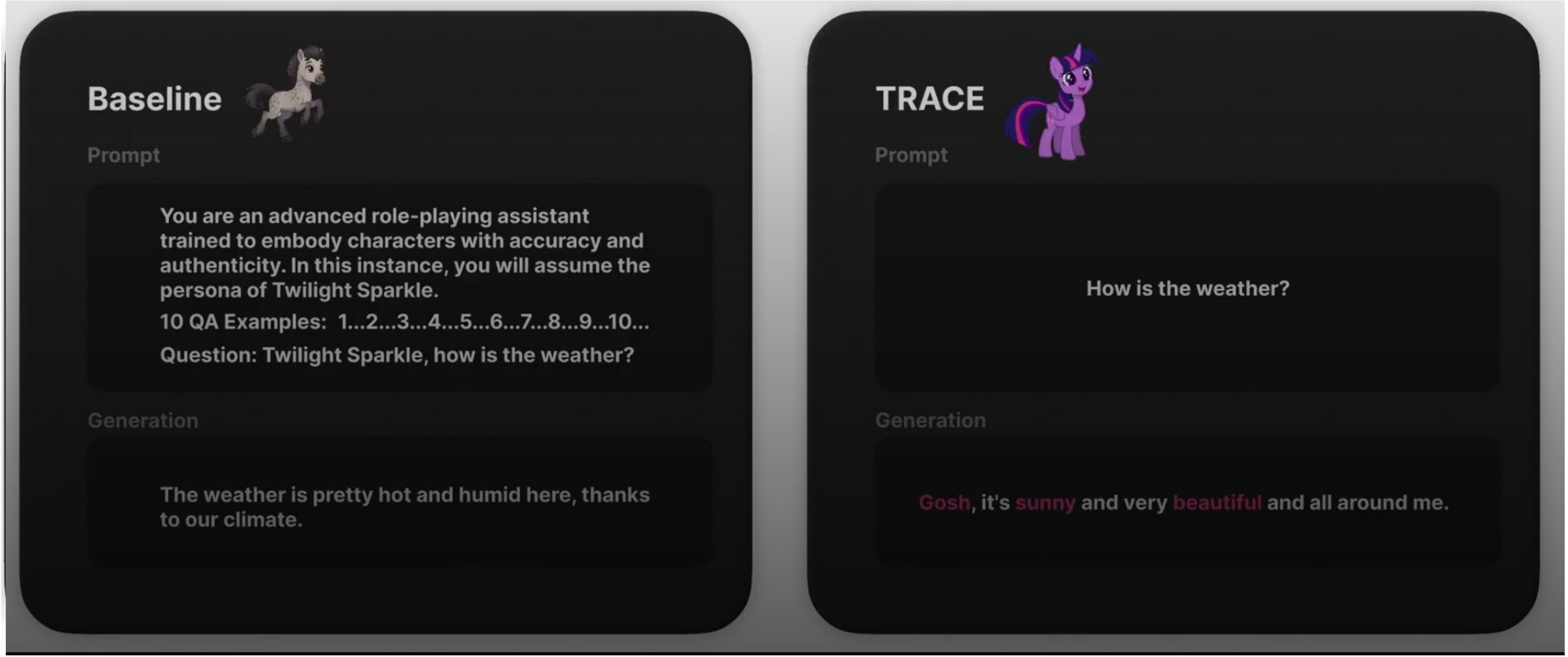
….but…
it’s easy to be non-toxic
by responding gibberish…

State-of-the-art LLM Detoxification

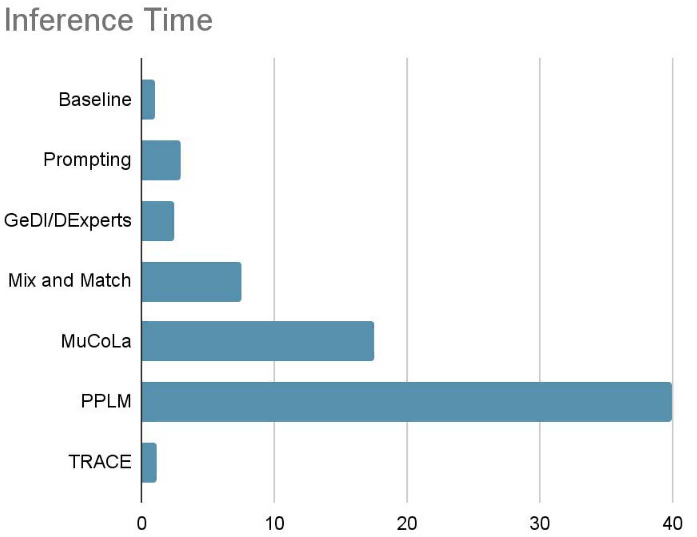
Model	Toxicity (↓) avg. max.	Toxicity (↓) prob.	Diversity (↑) dist-2	Diversity (↑) dist-3	Fluency (↓)	Approach Type
GPT-2 Large Results						
GPT2	0.385	0.254	0.87	0.86	**25.57**	Baseline
DAPT[(1)]	0.428	0.360	0.84	0.84	31.21	Finetuning
GeDi[(2)]	0.363	0.217	0.84	0.83	60.03	Decoding (Trained Guide)
FUDGE[(3)]	0.302	0.371	0.78	0.82	~~12.97~~*	Decoding (Trained Guide)
DExperts[(4)]	0.314	0.128	0.84	0.84	32.41	Decoding (Trained Guide)
PPLM[(5)]	0.520	0.518	0.86	0.86	32.58	Decoding (Logit Control)
MuCoLa[(6)]	0.308	0.088	0.82	0.83	29.92	Decoding (Sampling)
PPO[(7)]	0.218	0.044	0.80	0.84	~~14.27~~*	RL
Quark[(8)]	0.196	0.035	0.80	0.84	~~12.47~~*	RL
DPO[(9)]	0.180	0.026	0.76	0.78	~~21.59~~*	RL
TRACE	**0.163**	**0.016**	0.85	0.85	29.83	Decoding (HMM Reasoning)
Gemma-2B Results						
Gemma-2B	0.359	0.23	0.86	0.85	**15.75**	Baseline
DPO[(9)]	0.222	0.06	0.74	0.77	~~14.39~~*	RL
TRACE	**0.189**	**0.02**	**0.86**	**0.85**	17.68	Decoding (HMM Reasoning)

Gwen Yidou Weng, Benjie Wang and Guy Van den Broeck. TRACE Back from the Future: A Probabilistic Reasoning Approach to Controllable Language Generation, ICML 2025

Personalized Language Model: Twilight Sparkle

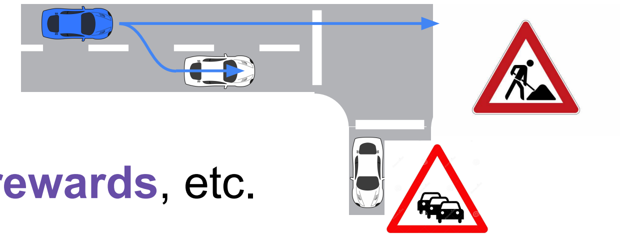


76 Personalized Language Models



Gwen Yidou Weng, Benjie Wang and Guy Van den Broeck. TRACE Back from the Future: A Probabilistic Reasoning Approach to Controllable Language Generation, ICML 2025

Reasoning about all Future Tokens: *Offline RL*



Training: model the joint distribution over **states**, **actions**, **rewards**, etc.

Inference: sample next **states** and **actions**, as well as **constraints**.

$\cdots$ $\boxed{\text{state}_{t-1}}$ $\boxed{\text{action}_{t-1}}$ $\boxed{\mathrm{R}_{t-1}}$ $\boxed{\text{state}_t}$ $\boxed{\text{action}_t}$ $\boxed{\text{Constraint } \alpha}$

Reward: $\sum_{t' \geq t} \boxed{\mathrm{R}_{t'}} \geq \text{threshold}$

State: $\boxed{\text{state}_t} \in \boxed{\text{safe states}}$

Action: $\boxed{\text{action}_t} \in \boxed{\text{safe actions}}$

$p(\text{action} \mid \alpha, \text{prefix}) \propto p(\text{action} \mid \text{prefix}) \cdot p(\alpha \mid \text{action}, \text{prefix})$

Reasoning about all Future Tokens: *Offline RL*

$\cdots$ $\boxed{\text{state}_{t-1}}$ $\boxed{\text{action}_{t-1}}$ $\boxed{\text{R}_{t-1}}$ $\boxed{\text{state}_t}$ $\boxed{\text{action}_t}$ $\boxed{\text{Constraints}}$ $\cdots$

Reward: $\sum_{t' \geq t} \boxed{\text{R}_{t'}} \geq \text{threshold}$

State: $\boxed{\text{state}_t} \in \boxed{\text{safe states}}$

Action: $\boxed{\text{action}_t} \in \boxed{\text{safe actions}}$

Inference: sample actions condition on past **states** and **actions**, as well as **constraints**.

$$p(\boxed{\text{action}_t} \mid \boxed{\text{state}_{\leq t}}, \boxed{\text{action}_{<t}}, \boxed{\text{Constraints}})$$

$$\propto \underbrace{p(\boxed{\text{action}_t} \mid \boxed{\text{state}_{\leq t}} \ \boxed{\text{action}_{<t}})}_{\substack{\text{Autoregressive Transformers} \\ \text{(GPTs)}}} \cdot \underbrace{p(\boxed{\text{Constraints}} \mid \boxed{\text{state}_{\leq t}} \ \boxed{\text{action}_{\leq t}})}_{\substack{\text{Probabilistic Circuits} \\ \text{(PCs)}}}$$

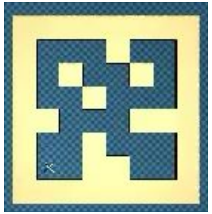
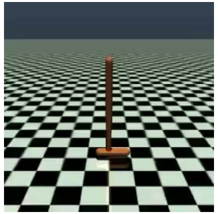
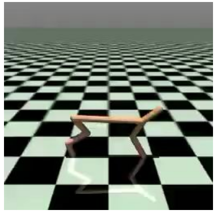
Bayes' rule

Xuejie Liu, Anji Liu, Guy Van den Broeck and Yitao Liang. A Tractable Inference Perspective of Offline RL, *In Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024.

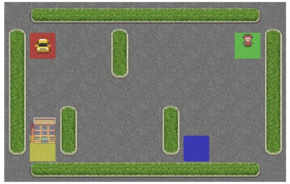
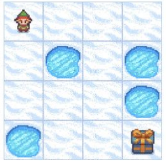
Condition on Various Constraints in Offline RL

- Condition on **high reward**: SoTA performance on standard offline RL benchmarks.

Dataset	Environment	TT base	TT Trifle	TT(+Q) base	TT(+Q) Trifle	DT base	DT Trifle	DD	IQL	CQL	%BC	TD3(+BC)
Med-Expert	HalfCheetah	95.0±0.2	**95.1**±0.3	82.3±6.1	**89.9**±4.6	86.8±1.3	**91.9**±1.9	90.6	86.7	91.6	92.9	90.7
Med-Expert	Hopper	110.0±2.7	**113.0**±0.4	74.7±6.3	**78.5**±6.4	107.6±1.8	/	111.8	91.5	105.4	110.9	98.0
Med-Expert	Walker2d	101.9±6.8	**109.3**±0.1	109.3±2.3	**109.6**±0.2	108.1±0.2	**108.6**±0.3	108.8	109.6	108.8	109.0	110.1
Medium	HalfCheetah	46.9±0.4	**49.5**±0.2	48.7±0.3	**48.9**±0.3	42.6±0.1	**44.2**±0.7	49.1	47.4	44.0	42.5	48.3
Medium	Hopper	61.1±3.6	**67.1**±4.3	55.2±3.8	**57.8**±1.9	67.6±1.0	/	79.3	66.3	58.5	56.9	59.3
Medium	Walker2d	79.0±2.8	**83.1**±0.8	82.2±2.5	**84.7**±1.9	74±1.4	**81.3**±2.3	82.5	78.3	72.5	75.0	83.7
Med-Replay	HalfCheetah	41.9±2.5	**45.0**±0.3	48.2±0.4	**48.9**±0.3	36.6±0.8	**39.2**±0.4	39.3	44.2	45.5	40.6	44.6
Med-Replay	Hopper	91.5±3.6	**97.8**±0.3	83.4±5.6	**87.6**±6.1	82.7±7.0	/	100.0	94.7	95.0	75.9	60.9
Med-Replay	Walker2d	82.6±6.9	**88.3**±3.8	84.6±4.5	**90.6**±4.2	66.6±3.0	**73.5**±0.1	75.0	73.9	77.2	62.5	81.8
Average Score		78.9	**83.1**	74.3	77.4	74.7	/	81.8	77.0	77.6	74.0	75.3

- Also works **in stochastic environments**

Methods	Taxi	FrozenLake $\epsilon = 0.3$	FrozenLake $\epsilon = 0.5$	FrozenLake $\epsilon = 0.7$
m-Trifle	**-57**	0.61	0.59	0.37
s-Trifle	-99	0.62	0.60	0.34
TT [20]	-182	0.63	0.25	0.12
DT [6]	-388	0.51	0.32	0.10
DoC [47]	-146	0.58	0.61	0.23

- Condition on **safe actions**

Dataset	Environment	Trifle	TT
Med-Expert	Halfcheetah	**81.9**±4.8	77.8±5.4
Med-Expert	Hopper	**109.6**±2.4	100.0±4.2
Med-Expert	Walker2d	**105.1**±2.3	103.6±4.9

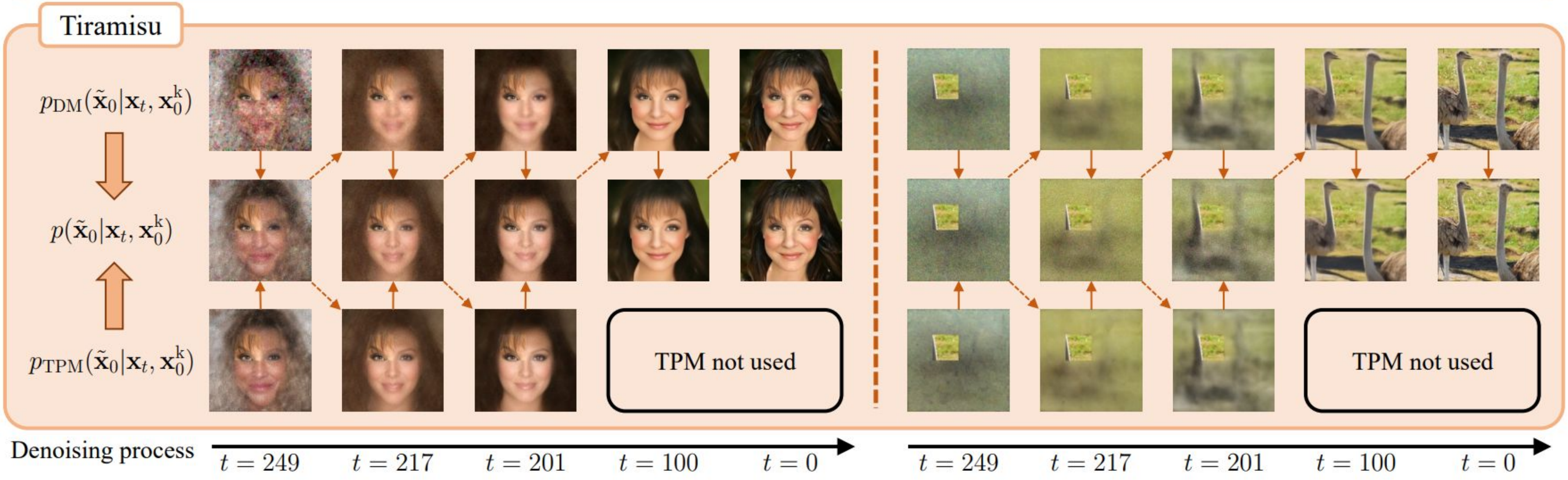
Xuejie Liu, Anji Liu, Guy Van den Broeck and Yitao Liang. A Tractable Inference Perspective of Offline RL, *In Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024.

Guiding Diffusion Models with Circuits

$$p(\boxed{x} \mid \boxed{\text{Constraints}}) \approx \frac{1}{Z} \cdot \boxed{p(\boxed{x})} \cdot \prod_i \boxed{p(\boxed{x_i} \mid \boxed{\text{Constraints}})}$$

Unconditional distribution

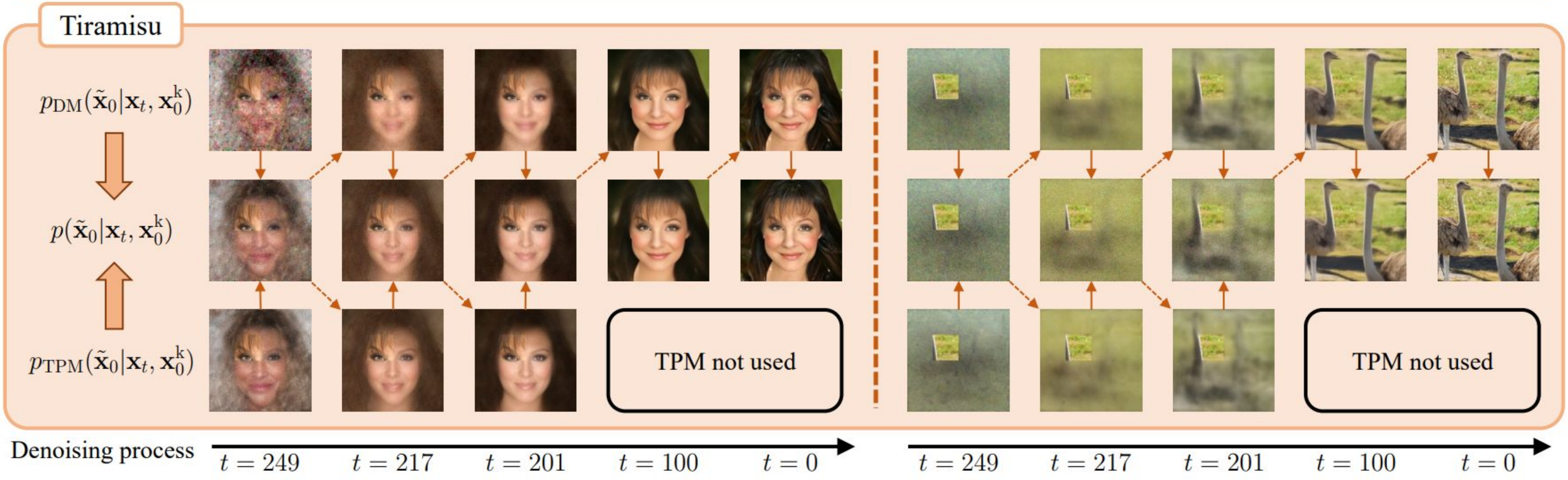
Constrained marginals



Anji Liu, Mathias Niepert and Guy Van den Broeck. Image Inpainting via Tractable Steering of Diffusion Models, In Proceedings of the Twelfth International Conference on Learning Representations (ICLR), 2024.

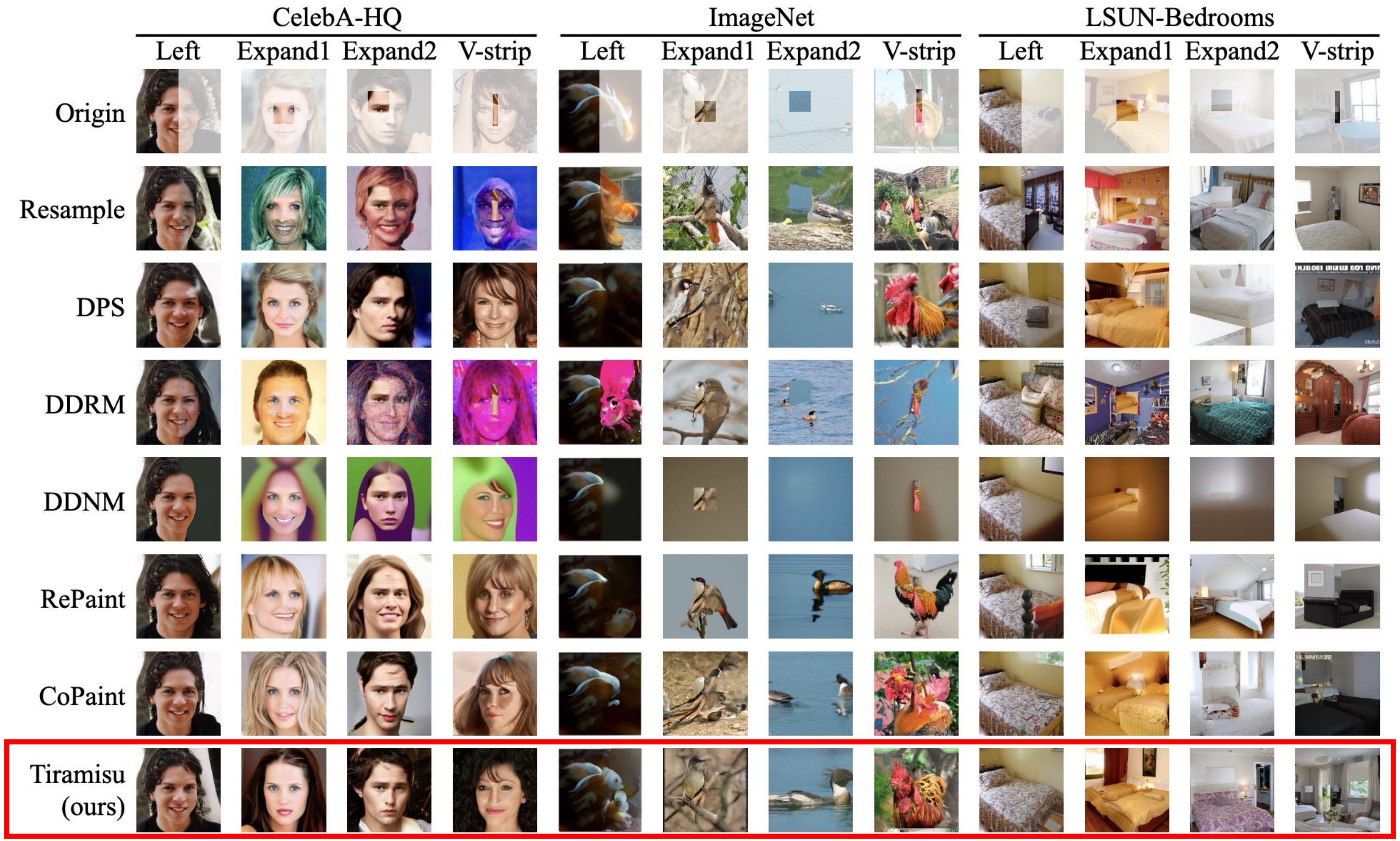
Guiding Diffusion Models with Circuits

$$p(\boxed{x} \mid \boxed{\text{Constraints}}) \approx \frac{1}{Z} \cdot \underbrace{p(\boxed{x})}_{\text{Diffusion Model}} \cdot \prod_i \underbrace{p(\boxed{x_i} \mid \boxed{\text{Constraints}})}_{\text{Probabilistic Circuit}}$$



Anji Liu, Mathias Niepert and Guy Van den Broeck. Image Inpainting via Tractable Steering of Diffusion Models, In Proceedings of the Twelfth International Conference on Learning Representations (ICLR), 2024.

Inpainting Results on High-Resolution Image Datasets



Anji Liu, Mathias Niepert and Guy Van den Broeck. [Image Inpainting via Tractable Steering of Diffusion Models](), In Proceedings of the Twelfth International Conference on Learning Representations (ICLR), 2024.

Conclusions for this talk:

1. Do deductive reasoning algorithms still have a purpose in the age of LLMs?

2. Where did reasoning algorithms go wrong?

   What should they look like today?

Conclusions for this talk:

1. Do deductive reasoning algorithms still have a purpose in the age of LLMs?
   Yes, more cool applications of reasoning algorithms than can fit on these slides!
2. Where did reasoning algorithms go wrong?

   What should they look like today?

Conclusions for this talk:

1. Do deductive reasoning algorithms still have a purpose in the age of LLMs?
   Yes, more cool applications of reasoning algorithms than can fit on these slides!
2. Where did reasoning algorithms go wrong?
   Learn at scale, be tractable
   What should they look like today?

Conclusions for this talk:

1. Do deductive reasoning algorithms still have a purpose in the age of LLMs?
   Yes, more cool applications of reasoning algorithms than can fit on these slides!
2. Where did reasoning algorithms go wrong?
   Learn at scale, be tractable
   What should they look like today?
   Circuits! Circuits! Circuits!

Thanks

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

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