Compositional Sculpting of Iterative Generative Processes

Timur Garipov[1] Sebastiaan De Peuter[2] Ge Yang[1,4] Vikas Garg[2,5] Samuel Kaski[2,3] Tommi Jaakkola[1]

[1]MIT CSAIL [2]Aalto University [3]University of Manchester [4]Institute for Artificial Intelligence and Fundamental Interactions [5]YaiYai Ltd

Overview

Focus: composition of **pre-trained** generative models (**Diffusion & GFlowNets)**

Motivation

- Large-scale general-purpose pre-training is becoming ubiquitous
- Need to **re-use and combine pre-trained models**
- Need tools for **control of sampling distribution**
- Applications in **multi-objective generation tasks**

Contributions

- New method for composing diffusion models or GFlowNets
- Operations on probability distributions
  - Binary: **"Harmonic Mean"** and **"Contrast"**
  - Generalized: N-ary, parameterized, chained
- **Controllable compositions** through inference: base models → (prior, observations) → posterior
- **Tractable sampling** from compositions using **classifier guidance on mixtures** of base models

Experimental Validation

- GFlowNets, 2D grid
- GFlowNets, molecule generation
- Diffusions models, colored MNIST

Paper **GitHub**

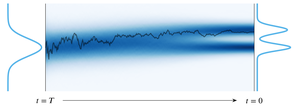
Background

	Models	Composition Operations	Sampling Algorithm
[Hinton, Neural Computation 2002] [Du et al, NeurIPS 2020]	Energy-based models (EBMs) $p_i(x) \propto \exp(-E_i(x;\theta))$	Principle: energy-function arithmetic Product: $\frac{1}{Z} p_1(x)\, p_2(x)$ Negation: $\frac{1}{Z} \frac{p_1(x)}{(p_2(x))^\gamma}$	MCMC Langevin dynamics
Challenge: iterative generative processes (Diffusion models & GFlowNets) impose delicate balance conditions			
[Liu et al, ECCV 2022] [Du et al, ICML 2023]	Diffusion models $p_i(x):\ s_{i,t}(x_t;\theta) \approx \nabla_{x_t} \log p_{i,t}(x_t)$	Principle: score-function arithmetic Product: $\frac{1}{Z} p_1(x)\, p_2(x)$ Negation: $\frac{1}{Z} \frac{p_1(x)}{(p_2(x))^\gamma}$	Diffusion sampling + annealed MCMC
Compositional sculpting (ours)	Diffusion models $p_i(x):\ s_{i,t}(x_t;\theta) \approx \nabla_{x_t} \log p_{i,t}(x_t)$ GFlowNets $p_i(x):\ p_{i,F}(s_{t+1}\mid s_t;\theta)$	Principle: mixture & conditional generative processes **Harmonic Mean**: $\frac{1}{Z} \frac{p_1(x)\, p_2(x)}{p_1(x)+p_2(x)}$ **Contrast**: $\frac{1}{Z} \frac{(p_1(x))^2}{p_1(x)+p_2(x)}$ (+) **other operations** in the paper	Diffusion mixture + classifier guidance GFlowNet mixture + classifier guidance

Diffusion model generates trajectories $\tau = (x_T \to \{x_t\}_{t=0}^T \to x_0)$ with terminal state distribution $p(x_0)$ by following a backward SDE

$$dx_t = \Big[f_t(x_t) - g_t^2 \underbrace{s_t(x_t;\theta)}_{\approx \nabla_{x_t} \log p_t(x_t)}\Big] dt + g_t\, d\overline{w}_t,$$

corresponding to a forward noising process $dx_t = f_t(x_t)\, dt + g_t\, dw_t$



GFlowNet generates trajectories $\tau = (s_0 \to \ldots \to s_{n-1} \to x)$ with terminal distribution $p(x) = \frac{R(x)}{Z}$ by following a forward policy

$$p_F(s_t \mid s_{t-1};\theta)$$

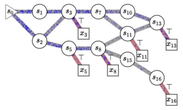
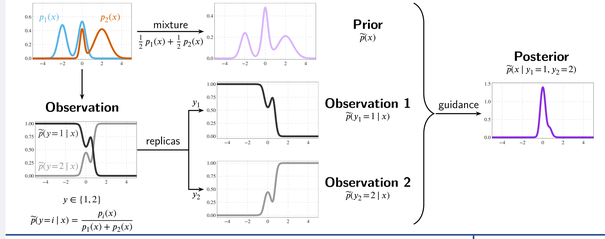


Image source: yoshuabengio.org/2022/03/05/generative-flow-networks

Composition of probability distributions: "Compositional Sculpting" operations

Base distributions



$$\widetilde{p}(y{=}i \mid x) = \frac{p_i(x)}{p_1(x)+p_2(x)}$$

Prior $\widetilde{p}(x) = \frac{1}{2}p_1(x) + \frac{1}{2}p_2(x)$

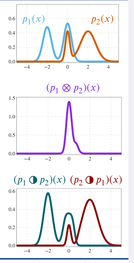
Observations $\widetilde{p}(y_k{=}i \mid x) = \frac{p_i(x)}{p_1(x)+p_2(x)}$

Posterior (composition) $\widetilde{p}(x \mid y_1{=}1, y_2{=}2) \propto \frac{p_1(x)p_2(x)}{p_1(x)+p_2(x)}$

"Harmonic Mean" $(p_1 \otimes p_2)(x) = \widetilde{p}(x \mid y_1{=}1, y_2{=}2) \propto \frac{p_1(x)p_2(x)}{p_1(x)+p_2(x)}$

"Contrast" $(p_1 \,\textcircled{-}\, p_2)(x) = \widetilde{p}(x \mid y_1{=}1, y_2{=}1) \propto \frac{(p_1(x))^2}{p_1(x)+p_2(x)}$

$(p_2 \,\textcircled{-}\, p_1)(x) = \widetilde{p}(x \mid y_1{=}2, y_2{=}2) \propto \frac{(p_2(x))^2}{p_1(x)+p_2(x)}$



Composing multiple distributions

Base distributions

$$p_1(x), \ldots, p_m(x)$$

Prior $\widetilde{p}(x) = \frac{1}{m}\sum_{j=1}^m p_j(x)$

Observations $y_k \in \{1, \ldots, m\}$, $\widetilde{p}(y_k{=}i \mid x) = \frac{p_i(x)}{\sum_{j=1}^m p_j(x)}$

Joint $\widetilde{p}(x, y_1, \ldots, y_n) = \widetilde{p}(x) \prod_{k=1}^n \widetilde{p}(y_k \mid x)$

Posterior (composition)

$$\widetilde{p}(x \mid y_1{=}i_1, \ldots, y_n{=}i_n) \propto \frac{\prod_{k=1}^n p_{i_k}(x)}{\left(\sum_{j=1}^m p_j(x)\right)^{n-1}}$$

Extension: parameterized operations

$$\widetilde{p}(x) = \tfrac{1}{2}p_1(x) + \tfrac{1}{2}p_2(x)$$

$$\widetilde{p}(x, y_1, y_2; a) = \widetilde{p}(x)\widetilde{p}(y_1 \mid x)\widetilde{p}(y_2 \mid x; a)$$

$$\widetilde{p}(y_1{=}1 \mid x) = \frac{p_1(x)}{p_1(x)+p_2(x)}$$

$$\widetilde{p}(y_1{=}2 \mid x) = \frac{p_2(x)}{p_1(x)+p_2(x)}$$

$$\widetilde{p}(y_2{=}1 \mid x; a) = \frac{a\,p_1(x)}{a\,p_1(x)+(1-a)\,p_2(x)}$$

$$\widetilde{p}(y_2{=}2 \mid x; a) = \frac{(1-a)\,p_2(x)}{a\,p_1(x)+(1-a)\,p_2(x)}$$

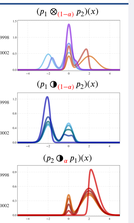
"Harmonic Interpolation"

$$(p_1 \otimes_{(1-a)} p_2)(x) \propto \frac{p_1(x)p_2(x)}{a\,p_1(x)+(1-a)\,p_2(x)}$$

"Parameterized Contrast"

$$(p_1 \,\textcircled{-}_{(1-a)}\, p_2)(x) \propto \frac{(p_1(x))^2}{a\,p_1(x)+(1-a)\,p_2(x)}$$

$$(p_2 \,\textcircled{-}_{a}\, p_1)(x) \propto \frac{(p_2(x))^2}{a\,p_1(x)+(1-a)\,p_2(x)}$$



Composition of iterative processes: sampling via mixtures and classifier guidance

Diffusion. Given base models $p_1(x_0), \ldots, p_m(x_0)$ with score networks $s_{1,t}(x_t), \ldots, s_{m,t}(x_t)$ and a shared forward process, a prior $\widetilde{p}(x) = \frac{1}{m}\sum_{j=1}^m p_j(x)$, and observations $y_1, \ldots, y_n$ defined as $\widetilde{p}(y_k{=}i \mid x_0) = \frac{p_i(x_0)}{\sum_{j=1}^m p_j(x_0)}$, the conditional $\widetilde{p}(x_0 \mid y_1, \ldots, y_n)$ is realized by the diffusion with the score

$$s_t(x_t, y_1, \ldots, y_n) = \underbrace{\left(\sum_{i=1}^m \widetilde{p}(y{=}i \mid x_t)\, s_{i,t}(x_t)\right)}_{\text{mixture}} + \underbrace{\nabla_{x_t} \log \widetilde{p}(y_1, \ldots, y_n \mid x_t)}_{\text{guidance}}$$

Note: diffusion mixture and conditional (classifier-guided) scores were derived in prior work.

GFlowNets. Given base models $p_1(x), \ldots, p_m(x)$ with forward policies $p_{1,F}(s' \mid s), \ldots, p_{m,F}(s' \mid s)$, a prior $\widetilde{p}(x) = \frac{1}{m}\sum_{j=1}^m p_j(x)$, and observations $y_1, \ldots, y_n$ defined as $\widetilde{p}(y_k{=}i \mid x) = \frac{p_i(x)}{\sum_{j=1}^m p_j(x)}$, the conditional $\widetilde{p}(x \mid y_1, \ldots, y_n)$ is realized by the forward policy

$$p_F(s' \mid s, y_1, \ldots y_n) = \underbrace{\left(\sum_{i=1}^m \widetilde{p}(y{=}i \mid s)\, p_{i,F}(s' \mid s)\right)}_{\text{mixture}} \cdot \underbrace{\frac{\widetilde{p}(y_1, \ldots, y_n \mid s')}{\widetilde{p}(y_1, \ldots, y_n \mid s)}}_{\text{guidance}}$$

Note*: GFlowNet mixture and conditional (classifier-guided) policies are original contributions.

*to the best of our knowledge

Classifier training

Full probabilistic model $\widetilde{p}(\{x_t\}_{t=0}^T)\widetilde{p}(y_1|x_0)\widetilde{p}(y_2|x_0)$

Compositions $\widetilde{p}(x|y_1{=}i, y_2{=}j)$

Mixtures & guidance require $\widetilde{p}(y|x_t)$, $\nabla_{x_t} \log \widetilde{p}(y_1, y_2|x_t)$

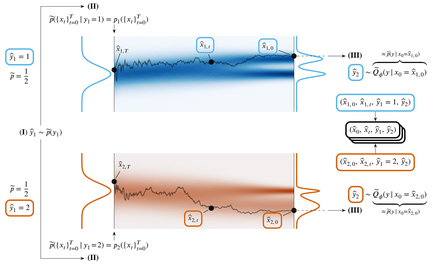
Method: train classifier $\widetilde{Q}_\phi(y_1, y_2|x_t) \approx \widetilde{p}(y_1, y_2|x_t)$

Terminal classifier $(t = 0)$ $\widetilde{Q}_\phi(y|x_0) \approx \widetilde{p}(y|x_0)$

Training data $(\widetilde{x}_0, \widetilde{y}_1) \sim \widetilde{p}(x_0, y_1)$

Non-terminal classifier $(t > 0)$ $\widetilde{Q}_\phi(y_1, y_2|x_t) \approx \widetilde{p}(y_1, y_2|x_t)$

Training data $(\widetilde{x}_t, \widetilde{y}_1, \widetilde{y}_2) \sim \widetilde{p}(x_t, y_1, y_2)$



$$\mathcal{L}_T(\phi) = \mathop{\mathbb{E}}_{(\widetilde{x}_0, \widetilde{y}_1) \sim \widetilde{p}(x_0, y_1)} \left[-\log \widetilde{Q}_\phi(y{=}\widetilde{y}_1 \mid x_0{=}\widetilde{x}_0)\right] = \mathop{\mathbb{E}}_{\widetilde{y}_1 \sim \widetilde{p}(y_1)} \mathop{\mathbb{E}}_{\widetilde{x}_0 \sim \widetilde{p}(x_0 \mid y_1{=}\widetilde{y}_1)} \left[-\log \widetilde{Q}_\phi(y{=}\widetilde{y}_1 \mid x_0{=}\widetilde{x}_0)\right]$$

(I) (II)

$$\mathcal{L}_N(\phi) = \mathop{\mathbb{E}}_{(\widetilde{x}_t, \widetilde{y}_1, \widetilde{y}_2) \sim \widetilde{p}(x_t, y_1, y_2)} \left[-\log \widetilde{Q}_\phi(y_1{=}\widetilde{y}_1, y_2{=}\widetilde{y}_2 \mid x_t{=}\widetilde{x}_t)\right] = \mathop{\mathbb{E}}_{\widetilde{y}_1 \sim \widetilde{p}(y_1)} \mathop{\mathbb{E}}_{\substack{(\widetilde{x}_0, \widetilde{x}_t) \sim \\ \widetilde{p}(x_0, x_t \mid y_1{=}\widetilde{y}_1)}} \mathop{\mathbb{E}}_{\widetilde{y}_2 \sim \widetilde{p}(y_2 \mid x_0{=}\widetilde{x}_0)} \left[-\log \widetilde{Q}_\phi(y_1{=}\widetilde{y}_1, y_2{=}\widetilde{y}_2 \mid x_t{=}\widetilde{x}_t)\right]$$

(I) (II) (III)

Training step

$$\widehat{\mathcal{L}}_T(\phi) = -\tfrac{1}{2}\log \widetilde{Q}_\phi(y{=}1|x_0{=}\widetilde{x}_{1,0}) - \tfrac{1}{2}\log \widetilde{Q}_\phi(y{=}2|x_0{=}\widetilde{x}_{2,0})$$

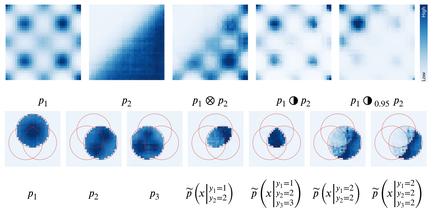
$$\widehat{\mathcal{L}}_N(\phi, \widetilde{\phi}) = -\tfrac{1}{2}\sum_{\widetilde{y}_1 \in \{1,2\}} \widetilde{Q}_\phi(y{=}\widetilde{y}_1 \mid x_0{=}\widetilde{x}_{1,0}) \cdot \log \widetilde{Q}_\phi(y_1{=}1, y_2{=}\widetilde{y}_2 \mid x_t{=}\widetilde{x}_{1,t}) - \tfrac{1}{2}\sum_{\widetilde{y}_2 \in \{1,2\}} \widetilde{Q}_\phi(y{=}\widetilde{y}_1 \mid x_0{=}\widetilde{x}_{2,0}) \cdot \log \widetilde{Q}_\phi(y_1{=}2, y_2{=}\widetilde{y}_2 \mid x_t{=}\widetilde{x}_{2,t})$$

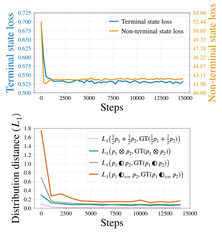
$$\phi \leftarrow \phi - \text{optimizer_step}\left(\nabla_\phi\left(\widehat{\mathcal{L}}_T(\phi) + \widehat{\mathcal{L}}_N(\phi, \widetilde{\phi})\right)\right)$$

$$\widetilde{\phi} \leftarrow \beta\,\widetilde{\phi} + (1-\beta)\,\phi$$

Experiments: GFlowNets

GFlowNet, 2D grid task





GFlowNet, molecule generation task

Molecule generation trajectories $\tau = (s_0 \to \ldots \to s_{n-1} \to x)$

- Terminal states: $x \in \mathcal{X}$ — molecules represented as 2D graphs
- Non-terminal states: $s \in \mathcal{S} \setminus \mathcal{X}$ — (incomplete) 2D molecular graphs
- Actions: $(s \to s') \in \mathcal{A}$:
  - a) attach one molecular fragment to the current molecular graph
  - b) "stop-action": terminate generation (only if current graph is valid)

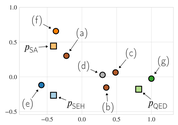
GFlowNet target distribution (at pre-training stage): $p(x) \propto (R(x))^\beta$

- $R(\cdot): \mathcal{X} \to [0, \infty)$ — reward function
- $\beta > 0$ — reward exponent

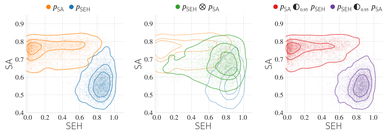
Rewards

- SEH — learned proxy of a protein binding score (soluble epoxide hydrolase)
- SA — synthetic availability score
- QED — quantitative estimate of drug-likeness

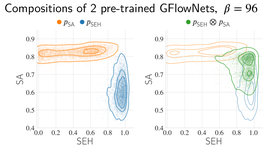
2D t-SNE embeddings of distributions computed based on pairwise earth mover's distances



Compositions of 2 pre-trained GFlowNets, $\beta = 32$



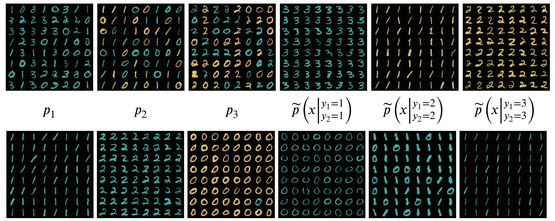
Compositions of 2 pre-trained GFlowNets, $\beta = 96$



Compositions of 3 pre-trained GFlowNets, $\beta = 32$

SEH	low				high			
SA	low		high		low		high	
QED	low	high	low	high	low	high	low	high
p_{SEH}	0	0	0	0	**62**	**9**	**24**	**5**
p_{SA}	0	0	**73**	**4**	0	0	**18**	**5**
p_{QED}	0	**40**	0	**26**	0	**21**	0	**13**
(a) $y = [\text{SEH}, \text{SA}]$	1	0	16	2	6	3	**54**	**18**
(b) $y = [\text{SEH}, \text{QED}]$	0	11	0	4	1	**48**	4	**32**
(c) $y = [\text{SA}, \text{QED}]$	0	15	1	**42**	0	8	2	**32**
(d) $y = [\text{SEH}, \text{SA}, \text{QED}]$	0	7	2	11	2	19	10	**49**
(e) $y = [\text{SEH}, \text{SEH}, \text{SEH}]$	0	0	0	0	**63**	9	24	4
(f) $y = [\text{SA}, \text{SA}, \text{SA}]$	0	0	**74**	5	0	0	17	4
(g) $y = [\text{QED}, \text{QED}, \text{QED}]$	0	**40**	0	23	0	23	0	14

Experiments: Diffusion models

Three diffusion models trained on ColorMNIST, sampling from p_1, p_2, p_3.
Iterative composition based on observations y_1, y_2, y_3:

$$\widetilde{p}(x|y_1, y_2, y_3) \propto \underbrace{\widetilde{p}(x)\widetilde{p}(y_1, y_2|x)}_{\propto \widetilde{p}(x|y_1, y_2)} \widetilde{p}(y_3|x, y_1, y_2)$$