Journal Club: Mean Flows for One-step Generative Modeling

WU SHENGYE

Institute of Science Tokyo

June 4, 2025

Notation

- Here are some notations used in today's slides.
- The instantaneous velocity field is denoted as $v(z_t, t)$
- The average velocity field is denoted as $u(z_t, t)$
- In the MeanFlow part, we will define that our time range t is in $[0, 1]$. When $t = 0$, $z_0 = x$ means the target data; When $t = 1$, $z_1 = \epsilon$ means the noise.
- Thus our sampling process will be started from $t = 1$ to $t = 0$.

Content

- Background of Flow matching
- Proposed Method: Mean Flow
- Experiment
- Conclusion

- Background of Flow matching
- Proposed Method: Mean Flow
- Experiment
- Conclusion

Flow matching

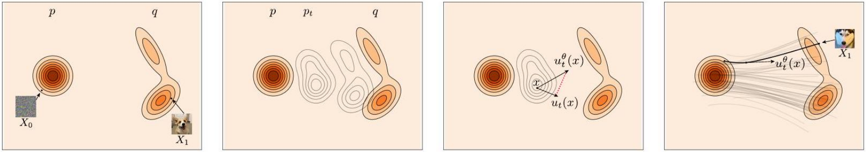
- Given access to a training dataset of samples from some target distribution q over $\mathbb{R}^d$, our goal is to build a model capable of generating new samples from it.
- Formally, we want to generate a novel sample $X_1 \sim q$ from the source distribution $X_0 \sim p$.
- There are several ways of generating the samples, such as Flow matching and Diffusion.
- Flow matching is a stimulation-free methods, which leads to the faster generation.

- FM builds a probability path $(p_t)_{0 \le t \le 1}$, from a known source distribution $p_0 = p$ to the data target distribution $p_1 = q$, where each p_t is a distribution over $\mathbb{R}^d$.
- FM is a regression objective to train the velocity field neural network which convert p_0 into p_1 along the probability path p_t.
- This velocity field determines a time-dependent flow $\psi : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$, defined as

$$\frac{\mathrm{d}}{\mathrm{d}t}\psi_t(x) = v_t\left(\psi_t(x)\right)$$

where $\psi_t := \psi(t,x)$ and $\psi_0(x) = x$. The velocity field v_t generates the probability path p_t if its flow ψ_t satisfies

$$X_t := \psi_t\left(X_0\right) \sim p_t \text{ for } X_0 \sim p_0$$



(a) Data. **(b)** Path design. **(c)** Training. **(d)** Sampling.

- Tips: Please notice that here we use v_t instead of u_t since in next part two velocity field will have different meanings.
- Our training objective Flow Matching Loss is as follows:

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{t,X_t} \left\| v_t^\theta (X_t) - v_t (X_t) \right\|^2, \text{ where } t \sim \mathcal{U}[0,1] \text{ and } X_t \sim p_t$$

- During the sampling process, we will try to deal with:

$$X_1 = X_0 + \int_0^1 v(X_\tau, \tau)\, d\tau$$

- The ground truth v_t is intractable, but we can use a conditional velocity field instead of it.
- e.g. If we define random variables X_t as follows:

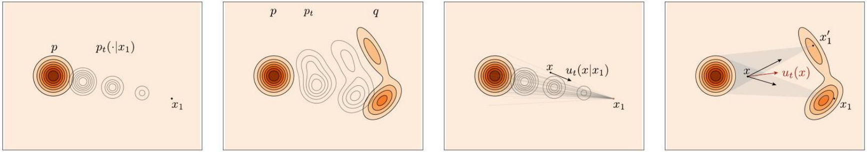
$$X_t = tX_1 + (1-t)X_0 \sim p_t$$

Then given the instance $X_1 = x_1$ we can write the conditional random variables:

$$X_{t|1} = tx_1 + (1-t)X_0 \quad \sim \quad p_{t|1}(\cdot \mid x_1) = \mathcal{N}\left(\cdot \mid tx_1, (1-t)^2 I\right)$$

In that case, the conditional velocity field will be defined as:

$$v_t(x \mid x_1) = \frac{x_1 - x}{1 - t}$$



(a) Conditional probability path $p_t(x|x_1)$. **(b)** (Marginal) Probability path $p_t(x)$. **(c)** Conditional velocity field $u_t(x|x_1)$. **(d)** (Marginal) Velocity field $u_t(x)$.

- Given a fixed target sample $X = x_1$, its conditional velocity field $v_t(x \mid x_1)$ generates the conditional probability path $p_t(x \mid x_1)$.
- The conditional Flow Matching Loss:

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \mathbb{E}_{t,X_t,X_1} \left\| v_t^\theta(X_t) - v_t(X_t \mid X_1) \right\|^2$$

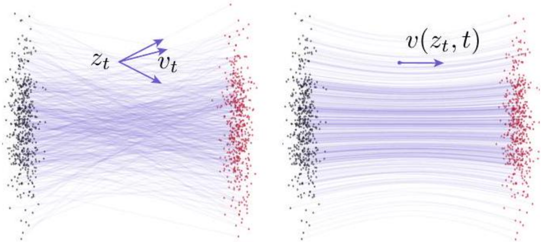
where $t \sim U[0,1], X_0 \sim p, X_1 \sim q$

- It can be proved that the FM loss and CFM loss provide the same gradients to learn v_t^θ:

$$\nabla_\theta \mathcal{L}_{\mathrm{FM}}(\theta) = \nabla_\theta \mathcal{L}_{\mathrm{CFM}}(\theta)$$

The limitations

- For the integration $X_1 = X_0 + \int_0^1 v\left(X_\tau, \tau\right) d\tau$, we should use some numerical method to approximate it in discrete time steps.
- e.g. Euler Method: $X_{t_{i+1}} = X_{t_i} + \left(t_{i+1} - t_i\right) v\left(X_{t_i}, t_i\right)$
- Some higher-order solvers might be adopted.
- It will lead to some inaccurate result while applying coarse dicertizations over curved trajectories.

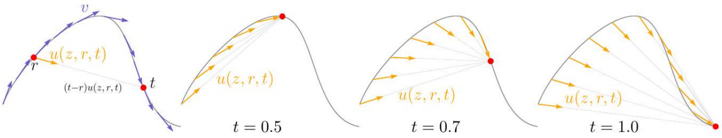


- Recently, the most popular way of defining the conditional velocity field is to use a linear interpolation assumption.
- However, even when the conditional flows are designed to be straight("rectified"), the marginal velocity field typically induces a curved trajectory.
- To improve the accuracy, higher NFE (Number of Function Evaluations) is needed.

Proposed Method: MeanFlow

- Motivation: We want to implement some algorithms which needs less steps of generating.
- We may refer to some physics process.
- E.g. Consider that we have two different velocity v_1 and v_2 for two time intervals $[a, b]$ and $[b, c]$. We would like to get the average velocity $\bar{v}$ of the total interval $[a, c]$
- Then we can easily calculate: $\bar{v} = \frac{(b-a)v_1+(c-b)v_2}{c-a}$
- We can use the average velocity to describe the velocity over a certain period of time

MeanFlow



- We define the average velocity field u_t between two time steps r and t as follows:

$$u(z_t, r, t) \triangleq \frac{1}{t-r} \int_r^t v(z_\tau, \tau)\, d\tau$$

- Obviously, the definition of u_t can satisfy the certain boundary conditions and "consistency" constraints.
- But in this definition, we still need to evaluate the function v for a lot of steps, can we do better?

MeanFlow

MeanFlow Indentity

From the definition of average velocity field:

$$(t-r)u\left(z_t,r,t\right)=\int_r^t v\left(z_\tau,\tau\right)d\tau$$

We differentiate both sides with respect to t, treating r as independent of t.

$$u\left(z_t,r,t\right)=v\left(z_t,t\right)-(t-r)\frac{d}{dt}u\left(z_t,r,t\right)$$

- Now we can see that we don't need the instantaneous velocity field v_t at every time step t.
- During sampling process, we can use: $z_0=z_1-u_\theta\left(z_1,0,1\right)$

Training Process

- For the training objective, it is nature to consider about using u_θ to approximate the average velocity u_t
- However, for $u\left(z_t, r, t\right) = v\left(z_t, t\right) - (t-r)\frac{d}{dt}u\left(z_t, r, t\right)$, we still need to know the marginal velocity field v. Besides, we need to calculate the derivative of average velocity field u.
- To deal with this, we define our target field u_{tgt} as:

$$u_{\mathrm{tgt}} = v_t - (t-r)\left(v_t \frac{\partial u_\theta}{\partial_z} + \frac{\partial u_\theta}{\partial_t}\right)$$

where $v_t = a_t' x + b_t' \epsilon$ is the conditional velocity, and by default, $v_t = \epsilon - x$. And we use the parameters of the network t_θ itself to get the derivative, which is called Bootstrapping strategy.

Training Process

- The training object is:

$$L(\theta) = \mathbb{E} \left\| u_\theta \left(z_t, r, t \right) - \mathrm{sg}\left(u_{\mathrm{tgt}} \right) \right\|_2^2$$

$$\text{where} \quad u_{\mathrm{tgt}} = v_t - (t - r) \left(v_t \frac{\partial u_\theta}{\partial_z} + \frac{\partial u_\theta}{\partial_t} \right)$$

- The $\mathrm{sg}\left(u_{\mathrm{tgt}}\right)$ means the stop gradient operation.
- u_{tgt} has already contained $\frac{\partial u_\theta}{\partial_z}$ and $\frac{\partial u_\theta}{\partial_t}$. If we continue calculating its derivative during the process of minimizing the $L(\theta)$, it will lead to double backpropagation.
- Thus we just use u_{tgt} as a constant.

Training process

- The training process of the proposed method comprises the following steps:

1. Sample x from data distribution and sample ϵ from noise distribution.
2. According to the predefined trajectory (e.g. linear interpolation), calculate z_t and condition velocity v_t
3. Sample the paired time steps (r, t), where $0 \leq r \leq t \leq 1$
4. Evaluate $u_\theta\left(z_t, r, t\right)$ by the network
5. Calculate $\frac{d}{d\theta} u_\theta\left(z_t, r, t\right)$ throughout Jacobian-Vector Product.
6. Construct the target field $u_{\mathrm{tgt}} = v_t - (t - r)\left(v_t \frac{\partial u_\theta}{\partial_z} + \frac{\partial u_\theta}{\partial_t}\right)$
7. Minimize the loss function $L\left(\theta\right)$ and update the parameters θ

- Background of Flow matching
- Proposed Method: Mean Flow
- Experiment
- Conclusion

Experiment

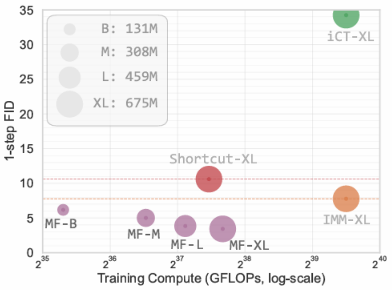


Figure 1: **One-step generation on ImageNet 256×256 from scratch**. Our *MeanFlow* (**MF**) model achieves significantly better generation quality than previous state-of-the-art one-step diffusion/flow methods. Here, iCT [43], Shortcut [13], and our MF are all **1-NFE** generation, while IMM's 1-step result [52] involves 2-NFE guidance. Detailed numbers are in Tab. 2. Images shown are generated by our 1-NFE model.

Experiment

method	params	NFE	FID
1-NFE diffusion/flow from scratch			
iCT-XL/2 [43]†	675M	1	34.24
Shortcut-XL/2 [13]	675M	1	10.60
MeanFlow-B/2	131M	1	6.17
MeanFlow-M/2	308M	1	5.01
MeanFlow-L/2	459M	1	3.84
MeanFlow-XL/2	676M	1	**3.43**
2-NFE diffusion/flow from scratch			
iCT-XL/2 [43]†	675M	2	20.30
iMM-XL/2 [52]	675M	1×2	7.77
MeanFlow-XL/2	676M	2	2.93
MeanFlow-XL/2+	676M	2	**2.20**

- Background of Flow matching
- Proposed Method: Mean Flow
- Experiment
- Conclusion

Conclusion

- Proposed a novel perspective on average velocity modeling: Shifted the core of generative flow modeling from instantaneous velocity to average velocity, providing a new theoretical foundation for few-step or one-step generation.
- Derived and utilized the MeanFlow identity: Starting from the definition of average velocity, rigorously derived a mathematical identity and adopted it as a principled training objective, avoiding reliance on heuristic constraints.
- Achieved state-of-the-art one-step generation performance: MeanFlow demonstrated leading generation quality under 1-NFE and 2-NFE settings across multiple standard datasets, particularly excelling in high-resolution generation on ImageNet 256$\times$256.