From Points to Functions: Infinite-dimensional Representations in Diffusion Models

Sarthak Mittal[†,1,2] , **Guillaume Lajoie**[1,2], **Stefan Bauer**[5], **Arash Mehrjou**[3,4]
[1]Mila, [2]Université de Montréal, [3]MPI-IS, [4]ETH Zurich, [5]KTH Stockholm

Abstract

Diffusion-based generative models learn to iteratively transfer unstructured noise to a complex target distribution as opposed to Generative Adversarial Networks (GANs) or the decoder of Variational Autoencoders (VAEs) which produce samples from the target distribution in a single step. Thus, in diffusion models every sample is naturally connected to a random trajectory which is a solution to a learned stochastic differential equation (SDE). Generative models are only concerned with the final state of this trajectory that delivers samples from the desired distribution. Abstreiter et al. (2021) showed that these stochastic trajectories can be seen as continuous filters that wash out information along the way. Consequently, there is an intermediate time step at which the preserved information is optimal for a given downstream task. In this work, we show that a combination of information content from different time steps gives a strictly better representation for the downstream task. We introduce an attention and recurrence based modules that "learn to mix" information content of various time-steps such that the resultant representation leads to superior performance in downstream tasks.

1 Introduction

Diffusion-based models (Sohl-Dickstein et al., 2015; Song et al., 2020; 2021; Sajjadi et al., 2018; Niu et al., 2020; Cai et al., 2020; Chen et al., 2020a; Saremi et al., 2018; Dhariwal & Nichol, 2021; Luhman & Luhman, 2021; Ho et al., 2021; Mehrjou et al., 2017) are generative models that apply step-wise perturbations to the samples of the data distribution (eg. CIFAR10), modeled via a Stochastic Differential Equation (SDE), until convergence to some prior unstructured distribution (eg. $\mathcal{N}(\mathbf{0}, \mathbf{I})$). In contrast to this diffusion process, a score model is learned to approximate the reverse process that iteratively converges to the data distribution from the prior distribution mentioned before. In this work, we follow Abstreiter et al. (2021) which augments such diffusion-based systems with an encoder for learning a representation that can be used for downstream tasks.

2 Beyond Fixed Representations

We first outline how diffusion-based representation learning systems are trained. Given some example $\mathbf{x}_0 \in \mathbb{R}^d$ which is sampled from the target distribution p_0, the diffusion process constructs the trajectory $(\mathbf{x}_t)_{t\in[0,1]}$ through the application of an SDE. In this work, we consider the Variance Exploding SDE (Song et al., 2021) for this diffusion process, defined as

$$d\mathbf{x} = f(\mathbf{x}, t) + g(t)d\mathbf{w} := \sqrt{\frac{d[\sigma^2(t)]}{dt}}d\mathbf{w} \tag{1}$$

where $\mathbf{w}$ is the standard Wiener process and $\sigma^2(\cdot)$ the noise variance of the diffusion process. This leads to a closed form distribution of $\mathbf{x}_t$ conditional on $\mathbf{x}_0$ as $p_{0t}(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \mathbf{x}_0, [\sigma^2(t) - \sigma^2(0)]\mathbf{I})$. Given this diffusion process modeled through the Variance Exploding SDE, the reverse SDE takes a similar form but requires the knowledge about the score function, i.e. $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ for

[†]Correspondence author sarthmit@gmail.com

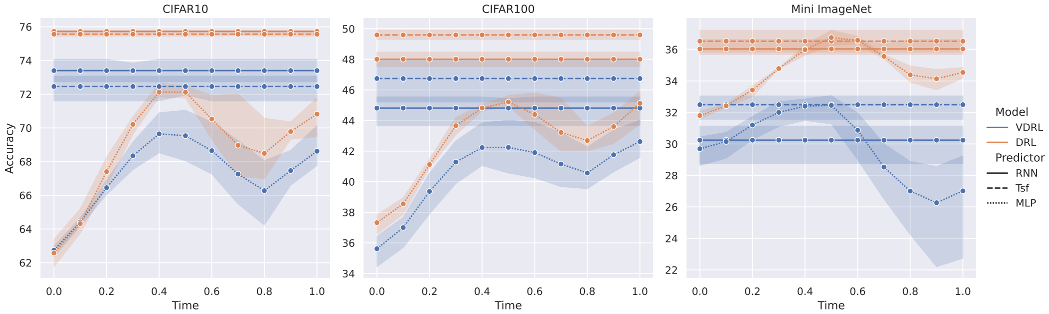


Figure 1: Downstream performance of single point based representations (MLP) and full trajectory based representations (RNN and Tsf) on different datasets for both types of learned encoders: probabilistic (VDRL) and deterministic (DRL).

all $t \in [0, 1]$. A common way to obtain this score function is through the Explicit Score Matching (Hyvärinen & Dayan, 2005) objective,

$$\mathbb{E}_{\mathbf{x}_t}\left[\|s_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)\|^2\right] \tag{2}$$

which suffers from just one hiccup, which is that data about the ground-truth score function is not available. To solve this problem, Denoising Score Matching (Vincent, 2011) was proposed,

$$\mathbb{E}_{\mathbf{x}_0}\left[\mathbb{E}_{\mathbf{x}_t|\mathbf{x}_0}\left[\|s_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p_{0t}(\mathbf{x}_t|\mathbf{x}_0)\|^2\right]\right] \tag{3}$$

where the term $\log p_{0t}(\mathbf{x}_t|\mathbf{x}_0)$ is available due to its closed-form structure. Given that the above objective cannot be reduced to 0, Abstreiter et al. (2021) proposes the objective

$$\mathbb{E}_{\mathbf{x}_0}\left[\mathbb{E}_{\mathbf{x}_t|\mathbf{x}_0}\left[\|s_\theta(\mathbf{x}_t, E_\phi(\mathbf{x}_0, t), t) - \nabla_{\mathbf{x}_t} \log p_{0t}(\mathbf{x}_t|\mathbf{x}_0)\|^2\right]\right] \tag{4}$$

where the additional input $E_\phi(\mathbf{x}_0, t)$ to the score function is obtained from a learned encoder and provides information about the unperturbed sample that might prove useful for denoising data at time step t in the diffusion trajectory. Training of this system can lead to the objective being reduced to 0, thereby providing incentive to the encoder $E_\phi(\cdot, t)$ to learn meaningful representations for each time t. From this, we obtain a trajectory-based representation $(E_\phi(\mathbf{x}_0, t))_{t\in[0,1]}$ from each sample $\mathbf{x}_0$ as opposed to just a finite sized representation obtained from typical Autoencoder (Bengio et al., 2013; Vinyals et al., 2016; Kingma & Welling, 2013; Rezende et al., 2014) and Contrastive Learning (Chen et al., 2020b; Grill et al., 2020; Caron et al., 2021; Bromley et al., 1993; Chen & He, 2020) based approaches.

2.1 Infinite-dimensional representation of finite-dimensional data

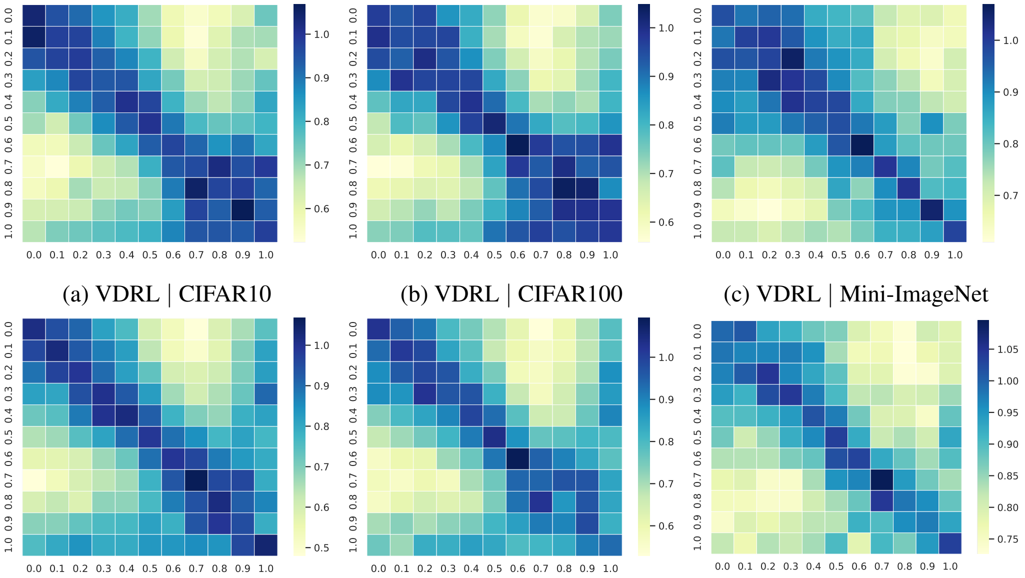
Normally in autoencoders or other *static* representation learning methods, the input data $\mathbf{x}_0 \in \mathbb{R}^d$ is mapped to a single point $\mathbf{z} \in \mathbb{R}^c$ in the code space. However, our proposed algorithm learns a richer representation where the input $\mathbf{x}_0$ is mapped to a curve in $\mathbb{R}^c$ instead of a single point through the encoder $E_\phi(\cdot, t)$. Hence, the learned code is produced by the map $\mathbf{x}_0 \to (E_\phi(\mathbf{x}_0, t))_{t\in[0,1]}$ where the infinite-dimensional object $(E_\phi(\mathbf{x}_0, t))_{t\in[0,1]}$ is the encoding for $\mathbf{x}_0$.

The learned code is at least as good as static codes in terms of separation induced among the codes. Consider two input samples $\mathbf{x}_0$ and $\mathbf{x}'_0$, hence we have:

$$\|E_\phi(\mathbf{x}_0, 0) - E_\phi(\mathbf{x}'_0, 0)\| \leq \sup_{t\in[0,1]} \|E_\phi(\mathbf{x}_0, t) - E_\phi(\mathbf{x}'_0, t)\| \tag{5}$$

which implies that the downstream task can at least recover the separation provided by finite-dimensional codes from the infinite-dimensional code by looking for the maximum separation along the representation trajectory.

A downstream task can leverage this rich encoding in various ways. Consider the classification task where we want to find a mapping $f : \mathbb{R}^d \to \{0, 1\}$ from input data to the label space. Instead of giving $\mathbf{x}_0$ as the input to f, we define $f : \mathcal{H} \to \{0, 1\}$ where the input to the classifier is the whole trajectory $(E_\phi(\mathbf{x}_0, t))_{t\in[0,1]}$. Thus, the classifier can now use RNN and Transformer models to make use of the information content of the entire trajectories.



(a) VDRL | CIFAR10

(b) VDRL | CIFAR100

(c) VDRL | Mini-ImageNet

(d) DRL | CIFAR10

(e) DRL | CIFAR100

(f) DRL | Mini-ImageNet

Figure 2: Normalized Mutual Information between different points on the trajectory. Cell (i, j) demonstrates the normalized mutual information, estimated with the MINE algorithm, between the representations at time $t = i$ and $t = j$.

3 Experiments

We first train two kinds of diffusion-based generative model as outlined in Abstreiter et al. (2021), based on probabilistic (VDRL) and deterministic (DRL) encoders respectively. After training, the encoder model is kept fixed. For all our downstream experiments, we use this trained encoder to obtain the trajectory based representation for each of the samples. While the trajectories lie in a continuous domain $[0, 1]$, we sample it at a regular intervals with length 0.1. This leads to a discretization of the trajectory, which is then used for various analysis as outlined below.

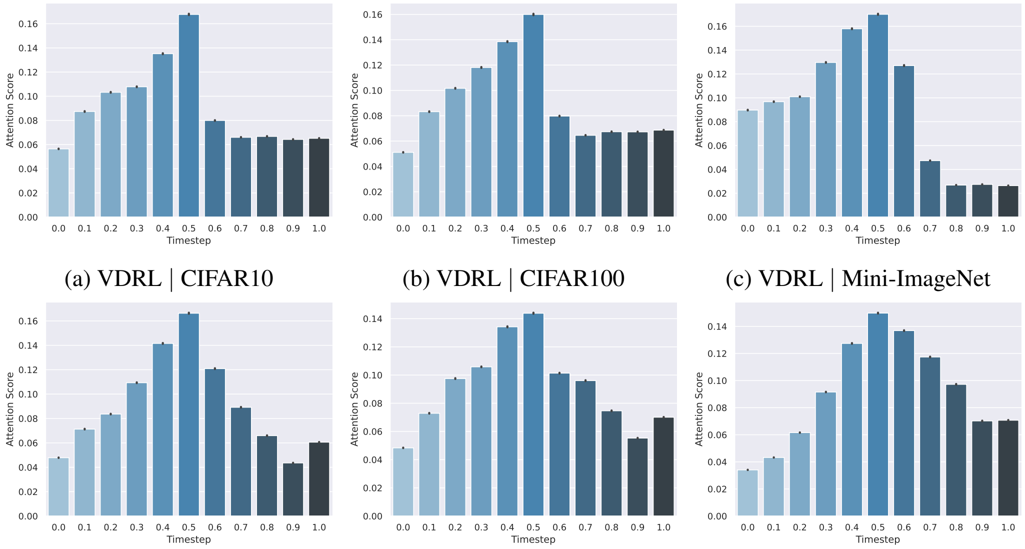
3.1 Downstream Performance Reveals Benefits of Trajectory Information

To understand the benefits of utilizing the trajectory-based representations, we train standard Multi-Layer Perceptron (MLP) models at different points on the trajectory and compare it with Recurrent Neural Network (RNN) (Hochreiter & Schmidhuber, 1997; Cho et al., 2014) and Transformer (Vaswani et al., 2017) based models that are able to aggregate information from different parts of the trajectory.

We evaluate the MLP, RNN and Transformer based downstream models on diffusion systems with both probabilistic encoders (VDRL) and also non-probabilistic ones (DRL). In Figure 1, we see the performance of these different setups for the following datasets: CIFAR10 (Krizhevsky et al., a), CIFAR100 (Krizhevsky et al., b) and Mini-ImageNet (Vinyals et al., 2016). We typically see that RNN and Transformer based models perform better than even the peaks obtained by the MLP systems. This shows that there is no single point on the trajectory that encapsulates all the information stored in the trajectory, and thus utilizing the whole trajectory as opposed to individual points leads to an improvement in performance.

3.2 Mutual Information Reveals Differences Along the Trajectory

In an effort to understand whether different parts of the trajectory based representation actually contain different types of information about the sample, we evaluate the mutual information between the representations at various points in the trajectory. We use the MINE algorithm (Belghazi et al., 2018) to estimate the mutual information between the representations at any two different points in the trajectory. Through this algorithm, we compute and analyse a normalized version of the



(a) VDRL | CIFAR10 (b) VDRL | CIFAR100 (c) VDRL | Mini-ImageNet

(d) DRL | CIFAR10 (e) DRL | CIFAR100 (f) DRL | Mini-ImageNet

Figure 3: Attention Scores provided to different points on the trajectories, which are obtained from diffusion based representation learning systems with probabilistic encoders (VDRL; top row) and with deterministic encoders (DRL; bottom row) across the following datasets (a) Left: CIFAR10, (b) Middle: CIFAR100, and (c) Right: Mini-Imagenet.

mutual information, defined as $\mathrm{NMI}(\mathbf{X};\mathbf{Y}) := \mathrm{I}(\mathbf{X};\mathbf{Y})/\sqrt{\mathrm{H}(\mathbf{X})\mathrm{H}(\mathbf{Y})}$ where $\mathrm{I}(\cdot\,;\,\cdot)$ is the standard Mutual Information function (Cover, 1999) and $\mathrm{H}(\cdot)$ is the entropy function.

Figure 2 illustrates the normalized mutual information between representations at different parts of the trajectory across three different datasets: CIFAR10, CIFAR100 and Mini-ImageNet as well as two different types of models: VDRL and DRL, where the former uses a probabilistic encoder and the latter doesn't. We see large normalized mutual information values near the principal diagonal and small values that are away from it, demonstrating that nearby representations on the trajectory are similar whereas distant points in the trajectory are considerably different. This shows that different parts of the trajectory learn to encode different kinds of information.

3.3 Attention Reveals Relevance of Different Parts of the Trajectory

To complement the analysis in Sections 3.1 and 3.2, we train a single-layered Transformer model for downstream prediction, which comes from a learned embedding that queries information from different parts of the trajectory. Through the analysis of the attention scores at different points in the trajectory, we realize that the middle parts of the trajectory are the most important, as illustrated in the high attention scores around $t = 0.5$ in Figure 3.

This is in line with the performance results in Figure 1 which also shows that amongst the single-point MLP-based systems, the best downstream performance is reached near the middle of the trajectory.

4 Conclusion

Through our analysis, we realize that the encoder $E_\phi(\cdot, t)$ actually learns different kinds of information at different time-steps t. Typically the mid-points of the trajectory are the most important for downstream classification tasks but we uncover that using the whole trajectory, which is a discretization of an infinite-dimensional object, is much better than just singular points on it. What kind of semantic information is encoded in the different parts of the trajectories? Can we leverage the full infinite-dimensional object without heuristic based discretizations? Our aim is to not only analyze the nature of information encoded in different parts of these trajectories but also leverage them for downstream tasks without heuristic discretizations.

ACKNOWLEDGEMENTS

SM would like to acknowledge the support of UNIQUE and IVADO towards his research. GL acknowledges the support from Canada CIFAR AI Chair Program, Samsung SAIT, and NSERC Discovery Grant [RGPIN-2018-04821].

References

Korbinian Abstreiter, Stefan Bauer, Bernhard Schölkopf, and Arash Mehrjou. Diffusion-based representation learning. *arXiv preprint arXiv:2105.14257*, 2021.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International conference on machine learning*, pp. 531–540. PMLR, 2018.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Jane Bromley, James Bentz, Leon Bottou, Isabelle Guyon, Yann Lecun, Cliff Moore, Eduard Sackinger, and Rookpak Shah. Signature verification using a ”siamese” time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7:25, 08 1993. doi: 10.1142/S0218001493000339.

Ruojin Cai, Guandao Yang, Hadar Averbuch-Elor, Zekun Hao, Serge Belongie, Noah Snavely, and Bharath Hariharan. Learning gradient fields for shape generation, 2020.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments, 2021.

Nanxin Chen, Yu Zhang, Heiga Zen, Ron J. Weiss, Mohammad Norouzi, and William Chan. Wavegrad: Estimating gradients for waveform generation, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. *CoRR*, abs/2006.10029, 2020b. URL https://arxiv.org/abs/2006.10029.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning, 2020.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020.

Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *arXiv preprint arXiv:2106.15282*, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). a. URL http://www.cs.toronto.edu/~kriz/cifar.html.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research). b. URL http://www.cs.toronto.edu/~kriz/cifar.html.

Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed, 2021.

Arash Mehrjou, Bernhard Schölkopf, and Saeed Saremi. Annealed generative adversarial networks. *arXiv preprint arXiv:1705.07505*, 2017.

Chenhao Niu, Yang Song, Jiaming Song, Shengjia Zhao, Aditya Grover, and Stefano Ermon. Permutation invariant graph generation via score-based generative modeling, 2020.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR, 2014.

Mehdi SM Sajjadi, Giambattista Parascandolo, Arash Mehrjou, and Bernhard Schölkopf. Tempered adversarial networks. In *International Conference on Machine Learning*, pp. 4451–4459. PMLR, 2018.

Saeed Saremi, Arash Mehrjou, Bernhard Schölkopf, and Aapo Hyvärinen. Deep energy estimator networks. *arXiv preprint arXiv:1805.08306*, 2018.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142.

Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *CoRR*, abs/1606.04080, 2016. URL http://arxiv.org/abs/1606.04080.