Exploiting language models for protein discovery with latent walk-jump sampling

Sai Pooja Mahajan[1,] **Nathan C. Frey**[1,] **Daniel Berenberg**[1,3] **Joseph Kleinhenz**[1]
Richard Bonneau[1] **Vladimir Gligorijević**[1] **Andrew Watkins**[1] **Saeed Saremi**[1]
[1]Prescient Design, Genentech [2]Antibody Engineering, Genentech
[3]Department of Computer Science, New York University
[4]Center for Data Science, New York University

Abstract

We introduce a single-step, score-based denoising framework for generative modeling of protein sequences from higher dimensional embeddings of pretrained language models. Our latent Walk-Jump Sampler (or L-WJS) framework learns the manifold of a smoothed latent space of a pretrained protein language model. New sequences are generated by score-based exploration using Langevin MCMC (walk) on the smoothed latent space and denoising back (jump) to the latent space. Our framework combines the attractive properties of the rich and semantically meaningful representations from pretrained protein language models and the improved sample quality of score-based modeling with the ease of training with a single-step denoising framework. We demonstrate that latent-WJS is data efficient, generates novel and diverse sequences that recapitulate biophysical properties of the underlying distribution, and opens-up avenues for sampling (both unguided and guided) from the latent space of various pretrained models.

1 Introduction

Score-based generative models have exhibited state-of-the-art performance in image generation (Ho et al., 2020). Latent diffusion models (LDMs) are score-based diffusion models that apply the diffusion framework in the latent space of pretrained autoencoders (Rombach et al., 2022), An important advantage of the LDM framework is that the pretrained autoencoders only need to be trained once and can therefore be reused for multiple trainings, different datasets and to explore a range of tasks.

Protein language models are pushing the boundaries of learning information at evolutionary scale from millions of protein sequences (Lin et al., 2023b; Rives et al., 2021). These models learn rich representations of protein sequences capturing residue-level biophysical properties to remote homology of proteins. Furthermore, language models encode secondary and tertiary structure information; a feature that has been exploited in the latest state-of-the-art protein structure prediction methods (Lin et al., 2023b; Jumper et al., 2021; Ruffolo et al., 2021, 2023).

In light of these advancements in score-based generative modeling and representation learning of protein and antibody sequences, we introduce the latent walk-jump sampler (latent-WJS or L-WJS) for the specific problem of generating protein sequences. L-WJS extends the discrete walk-jump sampler (dWJS) (Frey et al., 2023) built on the neural empirical Bayes (NEB) (Saremi & Hyvärinen, 2019) formalism to the smoothed latent space of pretrained protein language models. *With the L-WJS, we present a single-step denoising model at a single fixed noise-level to learn the manifold of the smoothed high-dimensional space of pretrained language models.* Similar to LDMs, such an approach allows the reuse of the embedded space of pretrained language models (encoder-decoder architectures) for multiple trainings and tasks. Unlike LDMs, the walk jump sampling formalism

Machine Learning for Structural Biology Workshop, NeurIPS 2023.

decouples generation via Langevin MCMC (walk) from the denoising (jump) steps. Our proposed L-WJS framework has the following unique features which distinguishes it from dWJS and existing generative models:

1. Data is "noised/corrupted" and "denoised/recovered" in the latent space instead of discrete input space akin to latent diffusion models such as Stable Diffusion.
2. Such a framework allows us to "walk" in a smoothed latent space that is semantically meaningful unlike the discrete one-hot input space.
3. The use of pretrained language models allows us to leverage the rich embedding space of general protein language models and leads to automatic fine-tuning in the sequence space of interest such as the observed functional space of antibody sequences against a specific target of interest.
4. As a by-product of the walk jump framework, the Langevin MCMC based walk provides a principled way to navigate a smoothed manifold of the rich-representations from pretrained protein language models.

1.1 Related Work

The discrete walk-jump sampler (dWJS) (Frey et al., 2023) introduced an extension of the NEB formalism to discrete data (one-hot encoded amino acid space) for antibody protein sequences. Here, we extend the dWJS formalism to the latent space. In this extension, we have been inspired by latent diffusion models (LDM), where the diffusion process occurs in the latent space of a pretrained autoencoder, enabling state-of-the-art results on text-conditioned image generation (Rombach et al., 2022). At a high level, this work combines these two distinct works into one.

Generative modeling of protein and antibody sequences has been dominated by autoregressive models such as ProGen (Madani et al., 2023) and ProGen2 (Nijkamp et al., 2022), IgLM (antibody-specific model trained with the infilling objective) (Shuai et al., 2021) and ProtGPT2 (decoder-only transformer model trained on proteins) (Ferruz & Höcker, 2022). At the same time, masked protein language models such as ESM (Rives et al., 2021), ESM2 (Lin et al., 2023b) and AntiBERTy (Ruffolo et al., 2021) have been trained on large corpuses of protein (Consortium, 2022) and antibody sequences (Olsen et al., 2022) to yield rich representations of protein and antibody sequences. Our work (latent-WJS) combines advancements in generative modeling (LDMs), representation learning (protein language models) with the NEB formalism to yield a generative model for sampling from the smoothed embedded/latent space of protein language models (see Figure 1 for a schematic).

2 Latent Walk-Jump Sampler

For a detailed background on walk-jump sampling and its application to antibody sequence generation in the discrete input space, we refer the readers to Saremi & Hyvärinen (2019) and Frey et al. (2023). We extend the discrete walk jump sampler to the latent space z of a pretrained autoencoder with an encoder (E) and decoder (D) such that encoder maps the discrete input x to the latent space z and the decoder maps z back to x. Following the NEB formalism, we transform the latent variable z with additive Gaussian noise, such that $Y = Z + N$, $N \sim \mathcal{N}(0, \sigma^2 I)$. The least-squares estimator of Z given $Y = y$ is the Bayes estimator given by (Robbins, 1956; Miyasawa, 1961)

$$\hat{z}(y) = \mathbb{E}[Z|y] = y + \sigma^2 \nabla \log p(y), \tag{1}$$

where $p(y) = \int p(y|z)p(z)dz$ is smoothed probability density. The estimator (1) is often expressed directly in terms of $g(y) = \nabla \log p(y)$ known as the score function (Hyvärinen, 2005) which is parameterized with a neural network denoted by $g_\phi : \mathbb{R}^{d_z} \to \mathbb{R}^{d_z}$. The estimator (1) then takes the following parametric form:

$$\hat{z}_\phi(y) = y + \sigma^2 g_\phi(y). \tag{2}$$

Putting this all together leads to the following least-squares denoising objective

$$\mathcal{L}(\phi) = \mathbb{E}_{z \sim p(z), \varepsilon \sim \mathcal{N}(0, \sigma^2 I)} \|z - \hat{z}_\phi(z + \varepsilon)\|^2,$$

which is optimized with stochastic gradient descent. Following training we sample from the smoothed distribution in the latent space with Langevin MCMC using the learned score function and then use the protein language model decoder to arrive at sequences $\hat{x}(y) = D(\hat{z}_\phi(y))$.

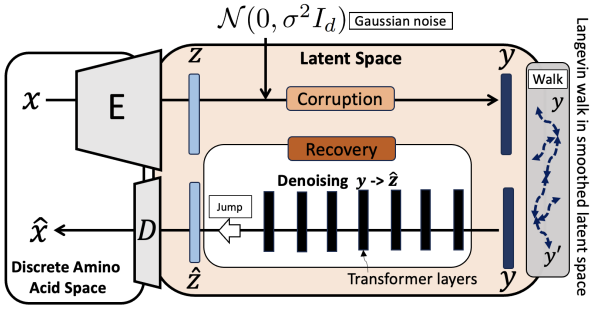

Figure 1: Latent Walk-Jump Sampler (Appendix A.1).

Table 1: Mean edit distance over the validation set (naive paired-OAS) between decoded ground truth (z) and denoised ($\hat{z}$) latents for latent-WJS models with the ESM2 pretrained encoder

# hidden layers	Latent size / Position	Noise Factor (σ)	E_{dist} (z, x)	E_{dist} $(\hat{z}, x) \downarrow$
	320	0.0	5.1	
1	320	1.0	5.1	3.8
1	320	2.5	5.1	15.7
2	640	2.5	5.1	3.9
2	640	5.0	5.1	4.7
2	640	7.0	5.1	6.6
2	640	10.0	5.1	10.7

3 Results

3.1 L-WJS achieves high fidelity reconstruction of input sequence from denoised latents

First we investigated the quality of the decoded sequences from the denoised latents in comparison to the clean latents. We report results for L-WJS models trained with the pretrained ESM2 (8M parameters) encoder and latent space constructed from either the last layer or the last and penultimate layer of the ESM2 model. Unlike Stable Diffusion (Rombach et al., 2022), our latent space is very high-dimensional (latent size between 320 to 640 per residue). For L-WJS, the choice of an appropriate noise factor for training is determined by the variance of the embedded space (Appendix A.1). Consequently, for comparison, we train models at a range of noise levels (σ) representing different amounts of corruption to the latent space.

In Table 1, we report the edit distance (E_{dist}) between the input sequence and the decoded sequence from denoised ($\hat{z}$) and clean (z) latents. As increasing levels of noise is added to the latents, it becomes more challenging for the model to reconstruct the input from the denoised latents. Nevertheless, models trained at fairly high noise levels (7.0 and 10.0 for ESM2 with two hidden layers) recover input sequences from denoised latents with sufficiently low edit distances, comparable to those of the sequences decoded from clean latents. Remarkably, at low to medium noise levels (2.5 to 5.0), sequences decoded from denoised latents are closer to the input sequence (lower edit distance) than those decoded from the clean latents. Thus, the denoising task improves the latent representation insofar as recovering the input sequence is concerned even though the model is only trained with a reconstruction loss (mean-square error or MSE loss) on the denoised latents (not the decoded sequence).

3.2 L-WJS generates diverse samples that recapitulate biophysical properties of natural antibodies

For the antibody sequence generation problem, we seek to generate sequences that not only match the properties of natural antibody sequences but also exhibit novelty and diversity. Keeping in line with previous work, in Table 2, we report the Wasserstein distance (W_{property}) for 15 biophysical properties between the sampled sequences and the reference distribution, the uniqueness (number of

Table 2: Metrics for heavy chain from 2000 samples generated with 20 *denovo* seed sequences (reported for heavy only in original paper). For L-WJS, we report metrics for the least number of Langevin steps needed to achieve the lowest Wasserstein distance (W_{property}). Step sizes and number of steps are reported in Table 5. *Note that achieving high diversity (E_{dist}, IntDiv) without matching the reference distribution (W_{property}) is not desirable.

Model	$W_{\text{property}} \downarrow$	Uniqueness ↑	E_{dist} $(\hat{z}, x)$ ↑	IntDiv ↑
dWJS (energy-based)	0.056	**1.0**	58.4	55.3
dWJS (score-based)	0.065	0.97	**62.7**	**65.1**
IgLM	0.087	**1.0**	48.6	34.6
ESM2	0.15	**1.0**	**70.99***	**77.56***
L-WJS-ESM2 (score-based) σ=2.5	0.053	**1.0**	56.6	54.1
L-WJS-ESM2 (score-based) σ=5.0	0.054	**1.0**	51.9	46.1
L-WJS-ESM2 (score-based) σ=7.0	0.052	**1.0**	54.2	49.5
L-WJS-ESM2 (score-based) σ=10.0	**0.051**	**1.0**	55.2	42.4

unique sequences sampled), the mean of the edit distance of sampled sequences from the reference distribution (E_{dist}) and the mean of the edit distance within the generated sequences (internal diversity or IntDiv). We compare our results to dWJS, IgLM (Shuai et al., 2021) and ESM2 (Lin et al., 2023b) as reported by Frey et al. (2023) (Appendix A.3). Sampled sequences are shown in Figure 2.

L-WJS generated sequences exhibit low Wasserstein distances to the biophysical property distributions of sequences sampled from paired OAS. While L-WJS exhibits comparable E_{dist} to dWJS, the internal diversity of the generated sequences is consistently lower than those generated with dWJS. We attribute this lower internal diversity to the latent space being "stickier" than the discrete space i.e. sequences starting close by (say from the same seed) in latent space collapse to similar sequences.

3.3 L-WJS enables high sample efficiency and extrapolation beyond observed data

One of the expected advantages of using a pretrained sequence encoder and decoder framework is to expand the sequence space of a small dataset in a semantically meaningful manner. For example, a recent study fine-tuned an LDM on a small set of extremely visually appealing images to generate images that were visually very appealing (Dai et al., 2023). Such fine-tuning is relevant for antibody sequence generation as well. For example, while the paired sequences in the OAS range in the hundreds of thousands, only a few hundred thousand antibody sequences have crystal structures and even fewer sequences have crystal structures for the antibody-antigen complex.

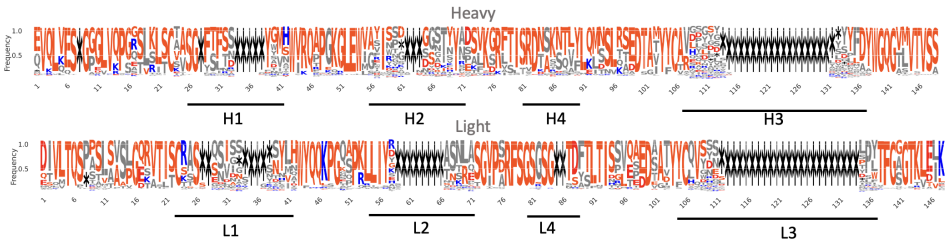

Figure 2: Example of generated sequences (AHo numbering scheme (Honegger & PluÈckthun, 2001)) from L-WJS-ESM2 (sigma=7.0) model trained on pOAS. Generated sequences exhibit length variability, resemble natural antibodies and show high diversity in high entropy regions (CDRs) and lower diversity in framework regions. See Figure 4 for coloring scheme.

In Table 3, we report the performance of the L-WJS on smaller datasets (1000 samples) derived from SAbDAb (Dunbar & Deane, 2015) (Appendix A.1). While both dWJS and L-WJS have impressive W_{property} metrics on the SAbDAb dataset, L-WJS generates samples with considerably higher mean internal diversity and edit distance from the reference set.

Table 3: Metrics on SAbDAb dataset (Dunbar et al., 2013). dWJS was trained with the score-based objective. L-WJS refers to L-WJS-ESM2 model trained with σ=7.0. Metrics are reported for the heavy chain for 2000 samples generated with 20 seeds with 100 samples per seed.

Dataset	Model	$W_{\text{property}} \downarrow$	Uniqueness ↑	$E_{\text{dist}}\ (\hat{z}, x) \uparrow$	IntDiv ↑
SabDAb	L-WJS	0.055	**1.0**	**54.3**	**48.2**
SabDAb	dWJS	**0.052**	**1.0**	50.2	36.7

Table 4: Predicted affinity for sampled CDR H3 sequences for Trastuzumab antibody for models trained on binders from Mason et al. (2021) dataset. *Models trained on paired OAS only.

Model	$p_{bind} \uparrow$
dWJS (energy-based) (Frey et al., 2023)	**0.96**
dWJS (score-based) (Frey et al., 2023)	0.91
L-WJS	0.91
L-WJS*	0.76

3.4 Langevin MCMC in smoothed latent space conserves structure of starting seed antibody

We further investigated the characteristics of the sequences sampled in a single trajectory by "walking" on the smoothed latent space. To this end, we tracked the sequences generated at every step along a long (200 steps) Langevin MCMC trajectory. We folded these sequenceswith ImmuneBuilder (Abanades et al., 2023) and characterized their germlines with ANARCI (Dunbar & Deane, 2016). We notice three important characteristics. First, the sequences along a single trajectory explore a structural space in the vicinity of the seed (Figure 5). Second, the generated samples show higher variability in CDR regions and lower variability in the framework regions (as expected). And lastly, the germline of the seed sequence is either preserved or constrained to structure-preserving germlines in a significant number of samples along an MCMC trajectory (Figure 6). We surmise that the Langevin MCMC walk enables exploration of the local structural neighborhood of a sequence reminiscent of structure-conditioned generation (Hsu et al., 2022; Dauparas et al., 2022; Mahajan et al., 2022, 2023).

3.5 L-WJS captures the distribution of HER2 binders

A common objective of antibody design or sequence generation is affinity maturation. To this end, we trained the L-WJS model on a set of 9,000 unique binders from Mason et al. (2021) dataset. We then generated 2000 sequences starting from the Trastuzumab sequence as the seed and predicted the fraction of sequences classified as binders by an affinity classifier trained on the full dataset (binders and non-binders) by Frey et al. (2023). The fraction of predicted binders for the L-WJS model fine-tuned on the binders is comparable to that of the score-based dWJS (Table 4). Furthermore, 76% of the sequences generated from L-WJS model trained on paired OAS sequences (without fine-tuning on binders) are predicted to bind HER2.

4 Conclusion

In this work, we introduced L-WJS, a single-step denoising score-based model that learns the manifold of the embedded space of pretrained protein language models with the NEB formalism (Saremi & Hyvärinen, 2019). L-WJS extends the dWJS framework Frey et al. (2023) to generate protein sequences from the latent space of pretrained protein language models. Applied to the task of protein sequence generation, L-WJS is both sample efficient and generates diverse, high-quality antibody-like sequences. In future work, we aim to explore guided sampling and conditional generation akin to latent diffusion models (Rombach et al., 2022; Jiang et al., 2023).

References

Brennan Abanades, Wing Ki Wong, Fergus Boyles, Guy Georges, Alexander Bujotzek, and Charlotte M. Deane. ImmuneBuilder: Deep-learning models for predicting the structures of immune proteins. *Communications Biology*, 6(1), May 2023. doi: 10.1038/s42003-023-04927-7. URL https://doi.org/10.1038/s42003-023-04927-7.

The UniProt Consortium. UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Research*, 51(D1):D523–D531, 11 2022. ISSN 0305-1048. doi: 10.1093/nar/gkac1052. URL https://doi.org/10.1093/nar/gkac1052.

Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, Matthew Yu, Abhishek Kadian, Filip Radenovic, Dhruv Mahajan, Kunpeng Li, Yue Zhao, Vladan Petrovic, Mitesh Kumar Singh, Simran Motwani, Yi Wen, Yiwen Song, Roshan Sumbaly, Vignesh Ramanathan, Zijian He, Peter Vajda, and Devi Parikh. Emu: Enhancing image generation models using photogenic needles in a haystack, 2023.

J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. de Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King, and D. Baker. Robust deep learning–based protein sequence design using ProteinMPNN. *Science*, 378(6615):49–56, October 2022. doi: 10.1126/science.add2187. URL https://doi.org/10.1126/science.add2187.

James Dunbar and Charlotte M. Deane. ANARCI: antigen receptor numbering and receptor classification. *Bioinformatics*, 32(2):298–300, September 2015. doi: 10.1093/bioinformatics/btv552. URL https://doi.org/10.1093/bioinformatics/btv552.

James Dunbar and Charlotte M Deane. ANARCI: antigen receptor numbering and receptor classification. *Bioinformatics*, 32(2):298–300, 2016.

James Dunbar, Konrad Krawczyk, Jinwoo Leem, Terry Baker, Angelika Fuchs, Guy Georges, Jiye Shi, and Charlotte M. Deane. SAbDab: the structural antibody database. *Nucleic Acids Research*, 42(D1):D1140–D1146, November 2013. doi: 10.1093/nar/gkt1043. URL https://doi.org/10.1093/nar/gkt1043.

Noelia Ferruz and Birte Höcker. Controllable protein design with language models. *Nature Machine Intelligence*, 4(6):521–532, 2022.

Nathan C. Frey, Daniel Berenberg, Karina Zadorozhny, Joseph Kleinhenz, Julien Lafrance-Vanasse, Isidro Hotzel, Yan Wu, Stephen Ra, Richard Bonneau, Kyunghyun Cho, Andreas Loukas, Vladimir Gligorijevic, and Saeed Saremi. Protein discovery with discrete walk-jump sampling, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Annemarie Honegger and Andreas PluÈckthun. Yet another numbering scheme for immunoglobulin variable domains: an automatic modeling and analysis tool. *Journal of molecular biology*, 309(3): 657–670, 2001.

Chloe Hsu, Robert Verkuil, Jason Liu, Zeming Lin, Brian Hie, Tom Sercu, Adam Lerer, and Alexander Rives. Learning inverse folding from millions of predicted structures. *ICML*, 2022. doi: 10.1101/2022.04.10.487779. URL https://www.biorxiv.org/content/early/2022/04/10/2022.04.10.487779.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(Apr):695–709, 2005.

Zixuan Jiang, Sitao Zhang, Rundong Huang, Shaoxun Mo, Letao Zhu, Peiheng Li, Ziyi Zhang, Xi Chen, Yunfei Long, Renjing Xu, and Rui Qing. PRO-LDM: Protein sequence generation with conditional latent diffusion models. August 2023. doi: 10.1101/2023.08.22.554145. URL https://doi.org/10.1101/2023.08.22.554145.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, July 2021. doi: 10.1038/s41586-021-03819-2. URL https://doi.org/10.1038/s41586-021-03819-2.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, March 2023a. doi: 10.1126/science.ade2574. URL https://doi.org/10.1126/science.ade2574.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023b.

Ali Madani, Ben Krause, Eric R. Greene, Subu Subramanian, Benjamin P. Mohr, James M. Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z. Sun, Richard Socher, James S. Fraser, and Nikhil Naik. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*, 41(8):1099–1106, January 2023. doi: 10.1038/s41587-022-01618-2. URL https://doi.org/10.1038/s41587-022-01618-2.

Sai Pooja Mahajan, Jeffrey A. Ruffolo, Rahel Frick, and Jeffrey J. Gray. Hallucinating structure-conditioned antibody libraries for target-specific binders. *Frontiers in Immunology*, 13, October 2022. doi: 10.3389/fimmu.2022.999034. URL https://doi.org/10.3389/fimmu.2022.999034.

Sai Pooja Mahajan, Jeffrey A. Ruffolo, and Jeffrey J. Gray. Contextual protein and antibody encodings from equivariant graph transformers. July 2023. doi: 10.1101/2023.07.15.549154. URL https://doi.org/10.1101/2023.07.15.549154.

Derek M Mason, Simon Friedensohn, Cédric R Weber, Christian Jordi, Bastian Wagner, Simon M Meng, Roy A Ehling, Lucia Bonati, Jan Dahinden, Pablo Gainza, et al. Optimization of therapeutic antibodies by predicting antigen specificity from antibody sequence via deep learning. *Nature Biomedical Engineering*, 5(6):600–612, 2021.

Koichi Miyasawa. An empirical Bayes estimator of the mean of a normal population. *Bulletin of the International Statistical Institute*, 38(4):181–188, 1961.

Erik Nijkamp, Jeffrey Ruffolo, Eli N. Weinstein, Nikhil Naik, and Ali Madani. Progen2: Exploring the boundaries of protein language models, 2022.

Tobias H Olsen, Fergus Boyles, and Charlotte M Deane. Observed antibody space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences. *Protein Science*, 31 (1):141–146, 2022.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15), April 2021. doi: 10.1073/pnas.2016239118. URL https://doi.org/10.1073/pnas.2016239118.

Herbert Robbins. An empirical Bayes approach to statistics. In *Proc. Third Berkeley Symp.*, volume 1, pp. 157–163, 1956.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

Jeffrey A Ruffolo, Jeffrey J Gray, and Jeremias Sulam. Deciphering antibody affinity maturation with language models and weakly supervised learning. *arXiv preprint arXiv:2112.07782*, 2021.

Jeffrey A Ruffolo, Lee-Shin Chu, Sai Pooja Mahajan, and Jeffrey J Gray. Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies. *Nature communications*, 14(1):2389, 2023.

Matthias Sachs, Benedict Leimkuhler, and Vincent Danos. Langevin dynamics with variable coefficients and nonconservative forces: from stationary states to numerical methods. *Entropy*, 19(12): 647, 2017.

Saeed Saremi and Aapo Hyvärinen. Neural empirical Bayes. *Journal of Machine Learning Research*, 20(181):1–23, 2019.

Richard W Shuai, Jeffrey A Ruffolo, and Jeffrey J Gray. Generative language modeling for antibody design. *bioRxiv*, 2021.

Martin Steinegger and Johannes Söding. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, 35(11):1026–1028, oct 2017. doi: 10.1038/nbt.3988. URL `https://doi.org/10.1038%2Fnbt.3988`.

A Appendix

A.1 Model architecture, training and datasets

Denoising network architecture: For all L-WJS models, we implemented a transformer-based architecture for the denoising network. For the models trained on paired OAS, we trained a transformer with 6 hidden layers, 8 attention heads and a feed forward dimension of 1152, and 144 features in the encoder/decoder inputs, and SiLU activations. Larger models did not improve performance. For models trained on the SAbDAb datasets, we implemented a smaller decoder-only transformer with 3 hidden layers, 8 attention heads and a feed forward dimension of 576, and 72 features in the decoder inputs with SiLU activations.

Pretrained ESM2 model: For all results reported in this work, we use the embeddings from the last and second last hidden layers of the pretrained ESM2 model with 8M parameters (Lin et al., 2023a). For decoding, we use the ESM2 decoder from HuggingFace that takes the last hidden representation and returns the logits per token/position on the sequence. To obtain a sequence from the logits, we simply take the argmax (amino acid token corresponding to the logit with the largest value) along the amino acid dimension.

Full model architecture: The full model consists of three main components (Figure 1): the pretrained encoder from ESM2 model, the denoising network that parameterizes the score function and the decoder for ESM2 model. A one-hot encoded antibody sequence is inputted to the pretrained ESM2 encoder that outputs an embedding z of size $L * D * N_{\text{hidden}}$ where L is the length of the chain (heavy or light), D is the size of the embedded space (320) and N_{hidden} is the number of hidden layers. Thus the output for a single chain from the pretrained encoder is $L * 320$ when only last hidden layer is used and $L * 640$ when the last two layers are used. The embedded latent is concatenated for the heavy and light chains such that the resulting embedding z with a size of $(L_{\text{heavy}} + L_{\text{light}} + L_{\text{mtoken}}) * D * N_{\text{hidden}}$, where L_{chain} is the length of the chain and L_{mtokens} are the sequence start and end tokens, transformed with Gaussian noise y and fed to the denoising transformer network. The denoising network outputs a tensor of the same length as the smoothed latent y. We then use (2) to predict $\hat{z}$ from $g_\phi(y)$. This follows separating the $\hat{z}$ into embeddings corresponding to heavy and light chains. $\hat{z}$ corresponding to each chain is then decoded by the ESM2 decoder. Either the full $\hat{z}$ (N_{hidden}=1) or partial $\hat{z}$ corresponding to the last layer only (N_{hidden}=2) is decoded with the pretrained ESM2 model's decoder followed by the argmax operation to obtain $\hat{x}$.

Code will be made available at https://github.com/Genentech/latent-walk-jump.

Datasets Paired sequences were padded to a maximum length of 297 (149 for the heavy chain and 148 for the light chain).

pOAS dataset preparation: Datasets were prepared by clustering the paired sequences (containing heavy and light chain pairs) from the Observed Antibody Space (OAS) database (Olsen et al., 2022). Sequences were clustered at 95% sequence identity with 80% coverage using MMseqs2 (Steinegger & Söding, 2017) and divided into training and validation sets. We did not observe a significant change in performance at 85% clustering. All models reported in Tables 1 and 2 were trained on naive paired OAS sequences.

SAbDAb dataset preparation: SAbDAb datasets consisted of entries from the PDB with the structure of the antibody-antigen complex. The antibody-antigen sequences were clustered on the antigen sequences with MMseqs2 (Steinegger & Söding, 2017) at 40% sequence identity and 80% coverage (after removing any non-protein letters).

A.2 Noise Factor

In Figure 3, we show the mean and variance of the embedded space (for each layer) of the pretrained ESM2 (8M parameters) for antibody sequences sampled from the Observed Antibody Space. Since the embedded space (for penultimate layer of ESM2) lies between ± 10.0, we train models with latents derived from hidden noise factors ranging from 2.5 (low-noise) to 10.0 (very-high noise).

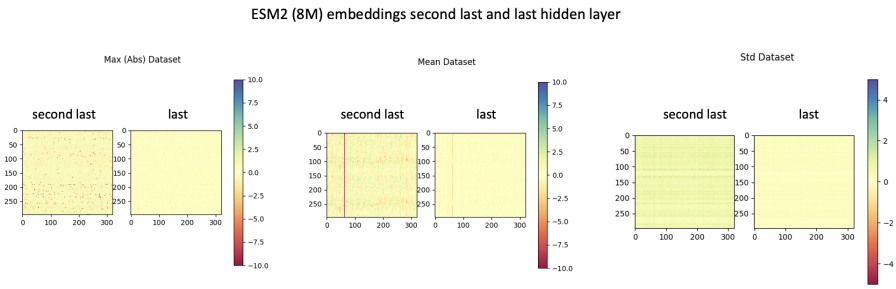

Figure 3: ESM2 embeddings

Table 5: Step-sizes and number of steps. *Similar performance.

Model	Step-size	Steps
L-WJS-ESM2 (score-based) σ=2.5	0.5	200
L-WJS-ESM2 (score-based) σ=5.0	0.5	50
L-WJS-ESM2 (score-based) σ=7.0	0.25	50
L-WJS-ESM2 (score-based) σ=7.0 *	0.5	20
L-WJS-ESM2 (score-based) σ=10.0	0.5	10

A.3 Sample generation and evaluation

Denovo generation For generating denovo sequences with L-WJS-ESM2 models, we initialized the seed sequence (fixed-length AHo numbering scheme (Honegger & PluÈckthun, 2001); 149 positions for heavy chain and 148 positions for light chain) by sampling each position from the corresponding position of an arbitrary sequence in the paired OAS.

Evaluation and comparison with ESM2, IgLM and dWJS We reuse evaluation metrics from Frey et al. (2023) in Tables 2 and 4. Briefly, Frey et al. (2023) generated samples from IgLM (Shuai et al., 2021) using the prompt shown below:

```
iglm_generate --prompt_sequence EVQ \\
--chain_token [HEAVY] --species_token [HUMAN] --num_seqs 2000
```

For ESM2 baseline (Lin et al., 2023b), they performed infilling at a high masking rate of (40%) to mimic ab initio/de novo generation. Frey et al. (2023) also noted that ESM2 (as expected) does not generate antibody-like sequences, and the high E_{dist} and IntDiv scores are therefore meaningless. For latent-WJS, ESM2 baseline serves two purposes. First, it is a powerful general protein language model baseline to show the gap in performance between a general, pre-trained protein MLM and latent-WJS that relies on ESM2 embeddings but is trained as a score-based single-step denoising model.

We use the underdamped Langevin MCMC algorithm (same as dWJS) from Sachs et al. (2017) also used by Frey et al. (2023) with the same hyperparameters. To generate samples for models trained at different noise factors, we tested a range of step-sizes and number of steps. We chose a combination of step-size and number of steps that resulted in the best W_{property} metrics. In Table 5, we report the number of steps and step-sizes used to report metrics in Table 2.

A.4 Example sequences from L-WJS

In Figure 4, we show the sequence logos for a denovo seed sequence.

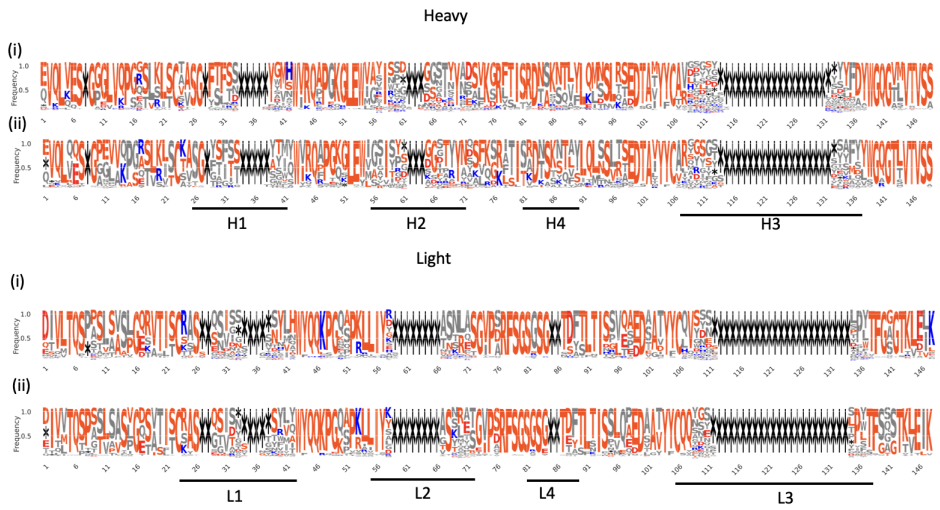

Figure 4: Example of generated sequences (AHo numbering scheme (Honegger & PluÈckthun, 2001)) from (i) L-WJS-ESM2 (sigma=7.0) model trained on pOAS and (ii) L-WJS-ESM2 (sigma=2.5) model trained on a subset of SAbDAb. Generated sequences exhibit length variability, resemble natural antibodies and show high diversity in high entropy regions (CDRs) and lower diversity in framework regions. Starting seed sequence is show in orange. Logos were generated with Logomaker with "charge" color scheme. Asterisk (*) represents a gap-token in AHo numbering scheme. CDRs are defined as: "L1": (23, 42), "L2": (56, 72), "L3": (106, 138), "L4": (81, 89), "H1": (23, 42), "H2": (56, 69), "H3": (106, 138), "H4": (81, 89).

A.5 Characterization of sequences sampled along a single trajectory

To characterize the sequences sampled along a single Langevin MCMC trajectory (or walk), we initialized the trajectory from a random sequence from the paired OAS and folded the generated sequences along a trajectory with Immunebuilder (Abanades et al., 2023). We further characterized the germline of the seed and sampled sequences with ANARCI (Dunbar & Deane, 2016). In Figure 5, we show the folded sequences and sequence logos from two trajectories. Mean RMSD stays under 1.0 angstrom at a step size of 0.5. In Figure 6, we show the germline of sampled and seed sequences.

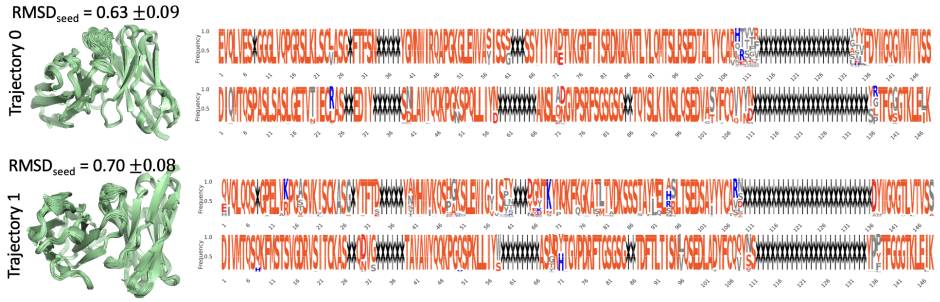

Figure 5: Folded structures and sequence logos (heavy-Top, light-Bottom) for each step along two 200 step Langevin MCMC trajectories at a step-size of 0.25 sampled with L-WJS-ESM2 (σ=7.0) model. For coloring and characters in sequence logos, see Figure 4.

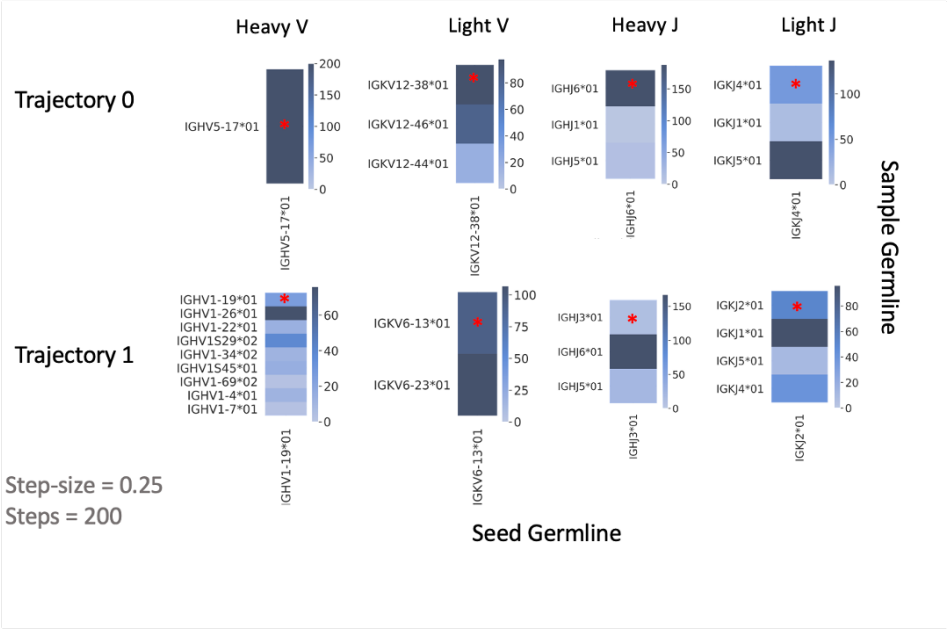

Figure 6: Germline of sampled sequences for each step along two 200 step Langevin MCMC trajectories at a step-size of 0.25 sampled with L-WJS-ESM2 (σ=7.0) model. Asterisk (*) denotes the cell with the same sampled germline as the seed.