nature machine intelligence

Article

https://doi.org/10.1038/s42256-023-00787-2

Variational autoencoder for design of synthetic viral vector serotypes

Received: 11 May 2023

Accepted: 18 December 2023

Published online: 23 January 2024

Check for updates

Suyue Lyu, Shahin Sowlati-Hashjin & Michael Garton

Recent, rapid advances in deep generative models for protein design have focused on small proteins with lots of data. Such models perform poorly on large proteins with limited natural sequences, for instance, the capsid protein of adenoviruses and adeno-associated virus, which are common delivery vehicles for gene therapy. Generating synthetic viral vector serotypes could overcome the potent pre-existing immune responses that most gene therapy recipients exhibit—a consequence of previous environmental exposure. We present a variational autoencoder (ProteinVAE) that can generate synthetic viral vector serotypes without epitopes for pre-existing neutralizing antibodies. A pre-trained protein language model was incorporated into the encoder to improve data efficiency, and deconvolution-based upsampling was used for decoding to avoid degenerate repetition seen in long protein sequence generation. ProteinVAE is a compact generative model with just 12.4 million parameters and was efficiently trained on the limited natural sequences. Viral protein sequences generated were used to produce structures with thermodynamic stability and viral assembly capability indistinguishable from natural vector counterparts. ProteinVAE can be used to generate a broad range of synthetic serotype sequences without epitopes for pre-existing neutralizing antibodies in the human population, effectively addressing one of the major challenges of gene therapy. It could be used more broadly to generate different types of viral vector, and any large, therapeutically valuable proteins, where available data are sparse.

Gene therapy is a powerful approach to treating—even curing—genetic disease, by the introduction of new genetic material into a patient that modifies their cell function. There are currently 13 United States Food and Drug Administration-approved gene therapies[1], perhaps most notably a treatment for spinal muscular atrophy[2], which cannot fail to impress for its dramatic effect on a horrific disease that ravages infants and devastates their families. These successes have generated much optimism that gene therapy can be applied to the many thousands of genetic diseases that afflict many tens of millions of people worldwide[3]. However, the delivery of therapeutically meaningful amounts of new genetic material to patient cells remains highly challenging. Chief among these challenges is the immunogenicity of effective delivery vectors[4–6]. Adenoviral (AdV) vectors are currently the most popular vector used for vaccines and cancer therapy delivery, with 575 currently in clinical trial[7]. AdVs have many advantages, including broad tropism profiles, lack of host genome integration, high transduction efficiency (60–80%) of most dividing and quiescent cells and large payload capacity (~38 kilobases)[8–10]. Despite this, substantial hurdles remain in deploying them to correct genetic disease, as the prevalence of pre-existing immunity against common human AdV serotypes is very high worldwide. This immunity is primarily driven by neutralizing antibodies, which have prevalence rates for the most widely used HAd5 vector ranging from 35% of the population in the United States to more than 90% in Cote d'Ivoire[11]. In a European study, 74% of children's sera

Institute of Biomedical Engineering, University of Toronto, Toronto, Ontario, Canada. e-mail: michael.garton@utoronto.ca

samples contained neutralizing antibodies for at least one serotype[12]. Another major related issue is that repeated vector administration is precluded even among patients without environmental exposure, as different serotypes are needed for each round[13]. Anti-AdV antibodies prevent AdV vectors from transducing their targets, leading to ineffective treatment[14]. Strategies to address this include using rare serotypes, non-human AdV vectors and AdV capsid engineering. These have met with limited success and do not adequately address the challenge of immunogenicity arising from repeated administration. As the major capsid protein, hexons were identified as the primary target of neutralizing antibodies[13]. Hexon modifications can assist evasion of the serotype-specific neutralizing antibodies[4]. However, changing the entire solvent-exposed surface, such that all potential neutralizing antibody epitopes are removed, involves introducing a large number of mutations. Introducing these many mutations randomly or using rational structure-based design cannot be done without catastrophic loss of protein function[15,16].

We hypothesized that machine learning could be used to generate dramatically different hexon proteins without impacting protein folding, particle assembly or cell transducing function. Here, we present a generative model capable of designing synthetic AdV vector serotypes that have never been surveilled by an animal or human immune system and therefore are predicted to avoid pre-existing AdV immunity. Deep learning models have been recently developed for de novo protein design[17–24]. However, these models were trained on much larger datasets. For example, ProteinGAN, the first Generative Adversarial Networks (GAN) model designed for protein sequence generation, comprised 60 million trainable parameters and was trained on 16,706 unique malate dehydrogenase sequences[21]. As only 88 serotypes of human adenovirus are currently known[25], the number of available unique hexon sequences is limited to 711 unique full-length sequences (UniprotKB database[26]). Due to limited available training data, models with a large number of parameters can be prone to overfitting, and a smaller model can be more appropriate[27]. Hexon sequences average 938 amino acids in length, suggesting a high likelihood of inter-residue dependency at longer distances. No previous work has reported generating sequences of comparable length. This, combined with the small dataset, required development of a small but expressive model that could be trained efficiently. To solve this, a pre-trained protein language model for amino-acid-level embedding was used, allowing transfer of knowledge learned by the pre-trained model on a large protein database. A variational autoencoder (VAE) framework was used to obtain an informative and structured latent space, and thereby convert the discrete protein sequence space to a continuous space for ease of sampling and manipulation. A special bottleneck attention module[28] was used in the encoder to map the high-quality amino-acid-level embedding—generated by the pre-trained model—to the latent space. A non-autoregressive deconvolution-based decoder was designed for sequence reconstruction from the latent variable. This model, which we are calling ProteinVAE, was able to generate high-quality, structurally stable hexon sequences with only 12.4 million parameters. Neutralizing antibody accessible hexon surfaces differed from natural hexons to the extent that they could be classified as new serotypes that are predicted to avoid pre-existing immunity.

Results

Generative VAE model for large proteins with limited data

To address the limited amount of hexon data available, a model that can be effectively trained on small datasets is required. A pre-trained protein language model[29] was incorporated in the encoder, as illustrated in Fig. 1. A convolutional neural network was used to extract a feature vector from the hidden amino-acid-level representation produced by the pre-trained protein language model (Extended Data Fig. 1a). Using the convolutional neural network-extracted feature vector as the query in the bottleneck attention module[28], global-level information was integrated to obtain a refined protein-level representation. Instead of training from scratch on the limited hexon data, the parameters in the pre-trained model were kept fixed. This permitted use of the high-quality amino-acid-level embeddings and more efficient training on a small dataset. ProtBert was chosen as the pre-trained protein language model, as it achieved the highest accuracy in prediction tasks among three models trained on sequence length ranges (2,048 amino acids) that cover hexons in the ProtTrans article[30]. ProtTrans[30] was one of the first efforts to apply language models on protein sequence data. ProtTrans[30] was highly cited, as its models were commonly used as benchmarks for both generation and predictions tasks. ProteinVAE can be easily adapted for alternative language models to extract embeddings.

For the decoder, a non-autoregressive processing method was used to avoid degenerate repetition commonly seen in generation of long sequences[31,32]. Inspired by HybridVAE[33], deconvolutional layers (Extended Data Fig. 1b) were designed to upsample the bottleneck representation, protein-level representation, to amino-acid-level representation M of size $L \times d_{de}$, where L means sequence length, and d_{de} means decoder hidden dimension. Specifically, a convolution layer with kernel size of 1×1 separated deconvolutional layers with a bigger 3×3 kernel[34]. This resulted in a reduction of the number of parameters needed. Next, a position-wise multi-head attention mechanism was used to capture the dependencies between amino acid usage at different positions, which allowed effective modelling of the long-distance interactions. A linear layer was used to convert the hidden representation M to the logits matrix P. Another multi-head attention module is designed to adjust for amino acid preference in different viruses[35], and, therefore, it is done across different amino acid channels. Some initial results of the model generated more helix sequences than strand (Extended Data Fig. 2). Combined with the consensus that strand proteins are harder to design, a reweighted cross-entropy loss that assigned higher penalty to the strand positions, predicted by SPOT-1D (ref. 36), was used in the final model.

ProteinVAE was trained on all hexon proteins in the UniprotKB[26,37] database that are annotated to be full-length. Since the desired output is full-length hexons that include all domains, sequences with incomplete domains were removed. Sequences with non-standard amino acids were also removed. The resultant dataset included 711 hexon sequences with length ranging from 893 to 992 amino acids.

Experimental setup for comparison with previous methods

To evaluate sequence generation, 7,000 samples were generated by sampling a Gaussian distribution with mean of 0 and standard deviation of 4 from the latent space and decoded by the ProteinVAE decoder. The top 1,000 sequences with highest average positional probability (APP) were selected. APP is calculated by averaging the token probability across all positions in a sequence, which reflects the model confidence level for the generated sequence. To benchmark hexon sequence generating capability against the current state-of-the-art, two recently published large transformer-based language models, ProtGPT2 (ref. 38) (738 million parameters) and ProGen2 (151 million parameters)[20], were selected and fine-tuned on the hexon dataset. ProtGPT2 was trained with a byte-pair encoding tokenizer, and ProGen2 was trained with single-amino-acid tokens. ESM-InverseFolding (ESM-IF1)[39], a competitive model, was used as a representative model for the fixed-backbone approach. ESM-IF1 was not fine-tuned due to the lack of structural data for most sequences in the training dataset. Other fixed-backbone methods required prohibitively large computational resources, for example, see refs. 40,41. Detailed fine-tuning and generation processes are described in the Methods. Sequence generation was done only once in each model. Another previously published multi-layer-perceptron-based VAE (MLP-VAE) model[42] was considered for a benchmark. The MLP-VAE model (12.7 million parameters) was trained on one-hot-encoded multiple sequence alignment (MSA) of

https://doi.org/10.1038/s42256-023-00787-2

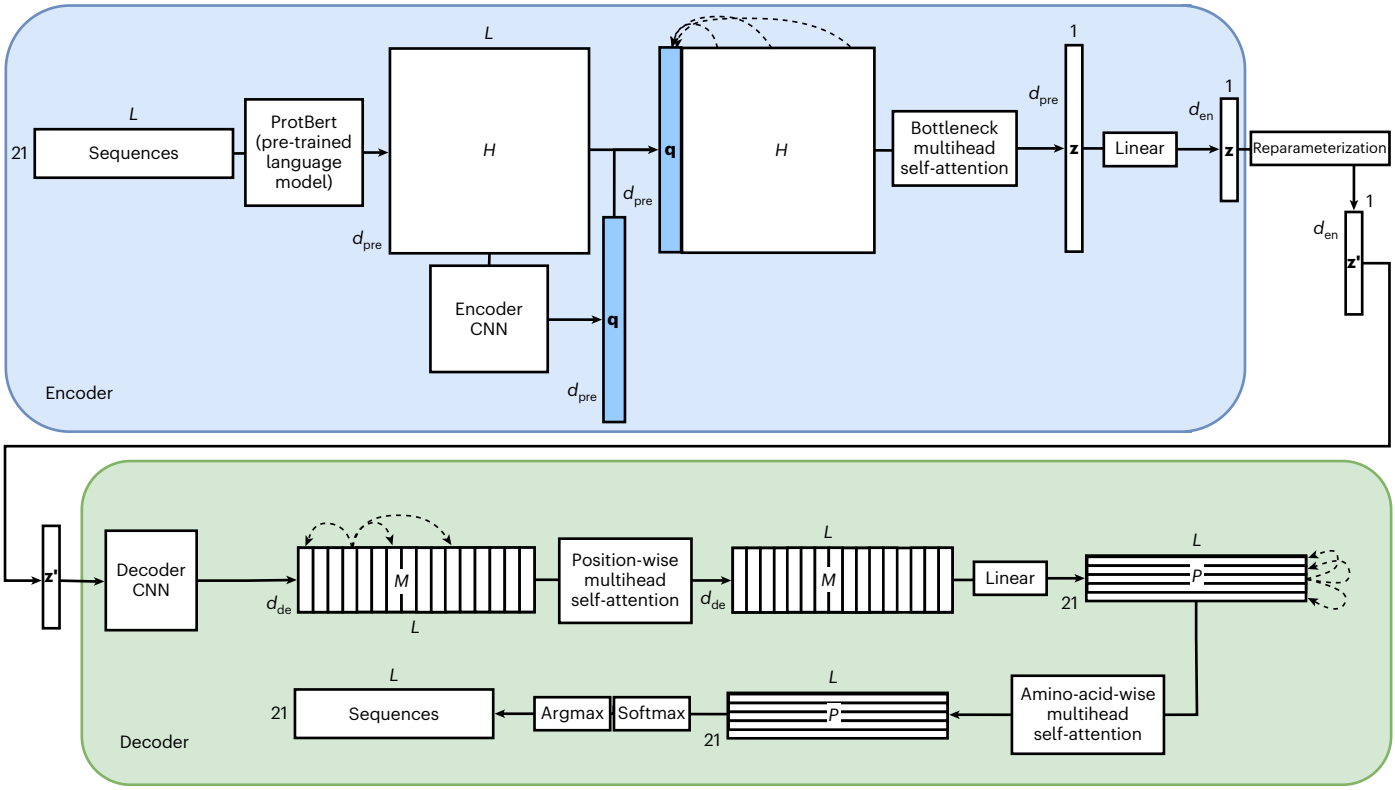


Fig. 1 | ProteinVAE architecture. Encoder: pre-trained language model (for example, ProtBert) converts the input one-hot-encoded sequences to amino-acid-level representation *H*. Encoder CNN extracts crude sequence-level representation **q**, which is used as the fixed query vector in the bottleneck attention mechanism to produce the refined global representation **z**. Dimensionality of **z** is adjusted with a linear layer. Reparameterization samples **z'** from the distribution defined by **z**. Decoder: deconvolutional networks are used for upsampling the latent vector **z'**. Self-attention along the length dimension is used to capture long-distance dependency. A linear layer converts hidden representation *M* to logits *P*, and another self-attention along the channel dimension is used to adjust for amino acid usage difference. The final logits are passed through a softmax layer to obtain probabilities, with the highest probability amino acid being selected as the prediction at each position. d_{en}, latent dimension; d_{pre}, hidden dimension of pre-trained language model; CNN, convolutional neural network; *L*, sequence length.

the hexon dataset, generated only 40 unique sequences with 1,000 randomly sampled latent vectors and was therefore not included in the comparison (Supplementary Note 1 and Supplementary Table 1). Benchmarking using the model developed by Ogden et al.[15] was considered, but was not possible due to the lack of publicly available hexon fitness landscape data. Generation with fine-tuned ProtBert[30] was also attempted; results can be found in Supplementary Note 2, Supplementary Table 2 and Supplementary Figs. 1 and 3–9.

ProteinVAE learns functional hexon-defining characteristics

To assess ProteinVAE's capacity to learn the distribution of natural hexon sequences, several metrics were computed comparing both local and whole-sequence-level patterns. Local amino acid pair association score[43] was calculated for all possible combinations in natural and model-generated sequences. (Fig. 2a). Similarity of ProteinVAE and ProtGPT2 scores to natural indicated that these models have learned the local amino acid patterns of natural hexons. In contrast, almost all amino acid pairs occur at a further distance than randomly shuffled sequences for ESM-IF1-generated sequences—substantially different from natural sequences. Lower level of association was seen in ProGen2-generated sequences, as indicated by the overall low absolute value, which also differs from the natural pattern. In addition, ProteinVAE-generated sequences also maintained a similar sequence profile in all seven hypervariable loops (Extended Data Fig. 3). Global sequence features of individual generated sequences were evaluated. Fine-tuned ProtGPT2 and ProGen2 perplexity levels were used to quantify the resemblance of generated to natural hexon sequence features (Fig. 2b), as a large language model (LLM) evaluator has demonstrated its potential in approximating human judgement in natural language processing tasks[44]. The number on the *x* axis is the average perplexity for each group. Lower perplexity means better fit in the training data of natural hexons. Natural sequences achieved lowest perplexity in both LLMs, while sequences with only 5% random mutations received considerably higher perplexity. This demonstrated that both models had learned the natural sequence profile and were sensitive enough to detect global sequence pattern changes. Sequences generated by fine-tuned ProtGPT2 and ProGen2 received low perplexity when evaluated by each respective LLM due to the known self-enhancement bias[44–46]. Excluding each model's self-evaluation, ProteinVAE-generated sequences received lower perplexity than the sequences with 5% random mutations. Another traditional metric, the HMMER score[47], was computed to assess the domain-level likelihood of each individual sequence containing the hexon domain (Fig. 2c). The percentage of hits and the average score are labelled next to the model name. Scores were normalized by the highest score seen in natural sequences. Higher score means higher likelihood of a sequence containing a domain. All natural sequences were identified as high-scoring hits in both hexon domains, whereas in sequences containing 5% random mutations, almost no hits were found. ProteinVAE- and ProtGPT2-generated sequences had the highest average likelihood to contain the hexon N-terminal domain, while ProtGPT2-generated sequences scored the highest for the C-terminal domain. ESM-IF1-generated sequences are likely to contain only the hexon N-terminal domain, as no hits were identified in the HMMER search for the hexon C-terminal domain. Next, the generated sequences were assessed for preservation of the natural evolutionary profile. Shannon entropies were computed for all valid

https://doi.org/10.1038/s42256-023-00787-2

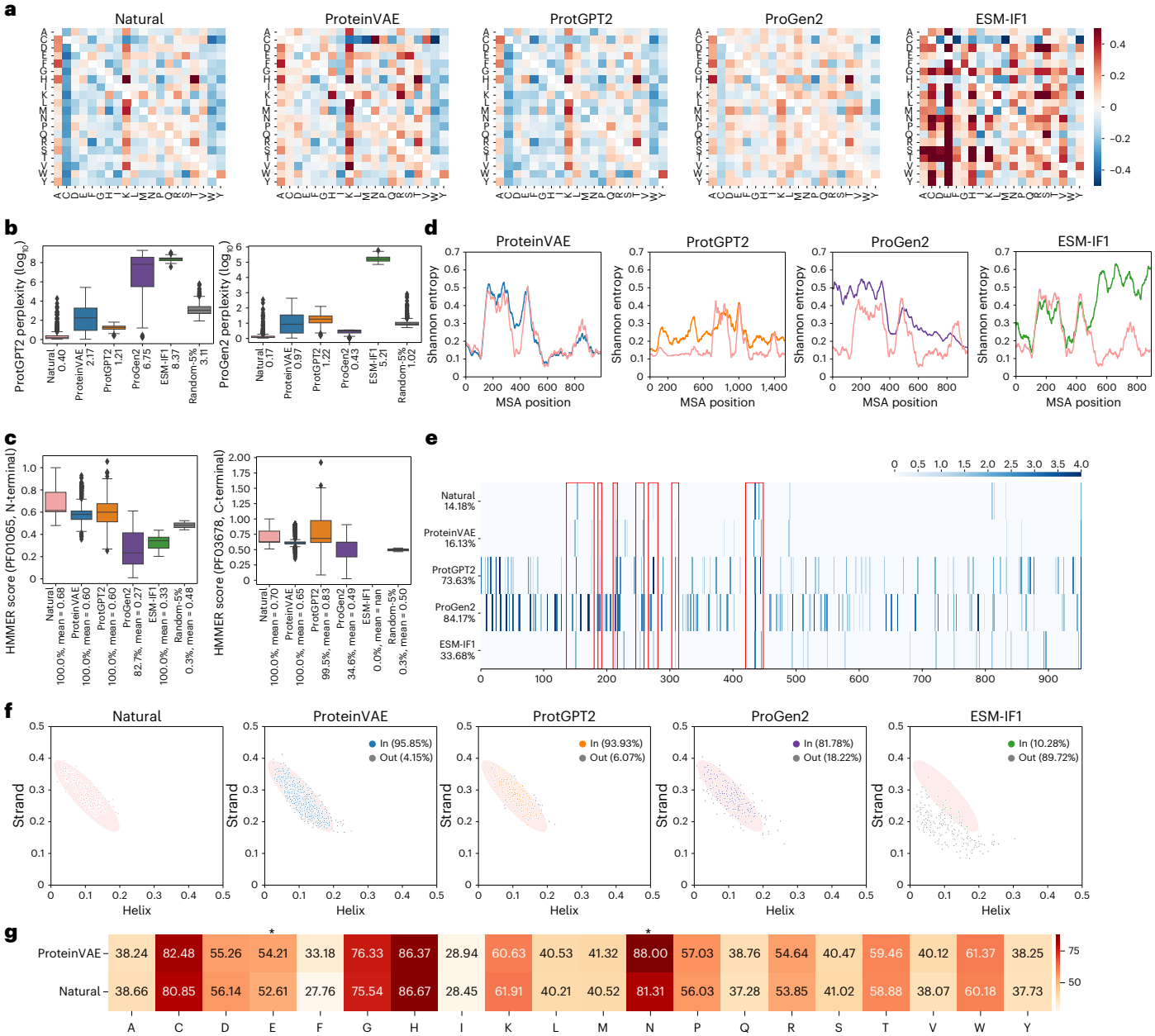


Fig. 2 | Comparing sequential and structural characteristics with natural hexons. a, Amino acid pair association scores for all ProtGPT2-generated, ProteinVAE-generated and natural sequences. Negative values (blue) indicate shorter distances compared with random shuffled sequences. **b**, Sequence perplexity ($\log_{10}$ transformed) from fine-tuned ProtGPT2 (left) and ProGen2 (right). **c**, HMMER score for the hexon N-terminal (left) and C-terminal (right) domains. In panels **b** and **c**, all natural sequences were used for analysis ($n = 711$). For all models, the same ratios of higher quality sequences were compared (ProteinVAE: $n = 1{,}000$; all other models: $n = 214$). Each box-plot shows the first and third quartiles, central line is median and whiskers show range of data with outliers displayed individually. **d**, Shannon entropy for natural hexons and sequences generated by all models in MSA columns with above 20% occupancy in each dataset. A higher value reflects higher sequence variability across samples. **e**, Positions of invalid columns in MSA (less than 80% occupancy) in the reference sequence of human adenovirus serotype 5 hexon (P04133). Colour indicates number of invalid columns (log transformed). Red squares show the location of HVRs. **f**, Helix and strand ratio in natural and generated hexons. Pink shade in all plots shows the area in the bi-variate normal distribution fitted on natural samples ($\alpha = 0.05$). In generated sequence plots, grey points represent outliers, while coloured points are sequences considered within the natural distribution. **g**, SASA for all amino acids in ProteinVAE-generated and natural proteins. Asterisk indicates amino acids with significantly different (unpaired two-sided Welch t-test, $\alpha = 0.05$; P value: Glu, 0.037; Asn, 1.13×10^{-7}) SASA values between two groups.

positions in the MSA of natural and generated sequences (Fig. 2d). The position of invalid columns was visualized with the location of hypervariable regions (HVRs) in Fig. 2e. Shannon entropy of ProteinVAE (Pearson's $r = 0.88$)-generated sequences presented peaks and valleys at similar locations to the natural sequences. Invalid columns in ProteinVAE MSA appeared mostly in the HVRs, which is similar to the natural MSA. Together, they suggested that ProteinVAE has learned the underlying sequence distribution. Both ProtGPT2 and ProGen2 have a high number of invalid columns, suggesting their generated sequences had more insertion or deletion mutations distributed throughout the sequence (Fig. 2e). Different sequence variability patterns were also present in ProtGPT2 (Pearson's $r = 0.7$)-generated and ProGen2 (Pearson's $r = 0.37$)-generated sequences. Notably, ESM-IF1-generated sequences exhibited different characteristics in the N-terminal and

https://doi.org/10.1038/s42256-023-00787-2

C-terminal domains, with poor resemblance to natural profile in the latter (probably because training length is only 500 amino acids[39]). Hexon sequence length added difficulties to its modelling. Generating long (~1,000s of tokens) and coherent texts in a specific small domain is challenging even for fine-tuned LLMs such as GPT2 (refs. 31,32), and generated texts typically suffer from degenerate repetition. To evaluate whether the generated sequences can avoid the degenerate repetition artifacts, while capturing certain local repetitive patterns observed in natural sequences[48], the number of repeated amino acids was calculated in a fixed-length window sliding across all possible positions in each sequence (Supplementary Fig. 1 and Supplementary Note 3). Regardless of the window size used, ProteinVAE (Pearson's $r = 0.92$) samples closely follow the repetitiveness trend of the natural, while ESM-IF1 showed a different repetition pattern (Pearson's $r = 0.13$). Notably, repetition did not increase as the generation progressed in fine-tuned ProtGPT2 (Pearson's $r = 0.54$) and ProGen2 (Pearson's $r = 0.56$) samples, but they did not maintain the local repetition patterns.

ProteinVAE learns hexon structural characteristics

To evaluate structural characteristics, first Q3 secondary structure was predicted for all natural and generated sequences with SPOT-1D (ref. 36). Figure 2f shows that strand and helix ratio is correlated (Pearson's $r = -0.86$) in natural hexons. A similar trend existed in sequences generated by all models, while a weaker correlation was observed in ESM-IF1 (Pearson's $r = -0.65$) (Fig. 2f). To further analyse secondary structure profile, a bi-variate normal distribution was fitted on the natural set, and out-of-distribution samples were identified in the generated sequences ($\alpha = 0.05$). ProteinVAE (95.85%) and ProtGPT2 (93.93%) samples share similar secondary structure composition with natural hexons (in-distribution samples) (Fig. 2f). Solvent accessible surface area (SASA) profiles were computed for all 20 amino acids in 100 randomly selected sequences from ProteinVAE samples and natural hexons (Fig. 2g). SASA was calculated from Alphafold2 (ref. 49)-predicted structures (Supplementary Note 4). SASA profiles for 18 amino acids are statistically indistinguishable in natural and ProteinVAE-generated sequences (Methods). Two amino acids (Glu, Asn) have significantly different SASA values, but their surface exposure character is retained. This comparison further supports that ProteinVAE-generated sequences are structurally similar to natural hexons; it also indicates that the ProteinVAE model has learned the physical–chemical properties of each amino acid to some extent. Only ProteinVAE-generated sequences were analysed due to limited computing resources.

ProteinVAE generates diverse hexon sequences

To visualize sequence diversity, an equal number of natural and generated sequences by all models were clustered at different thresholds. ProtGPT2- and ProGen2-generated samples produced many clusters even at extremely low identity threshold. Combining this trend with the high ratio of invalid columns in their MSA (Fig. 2e), it is likely that ProtGPT2 and ProGen2 inserted sequence fragments from the vast Uniref50 database that they were pre-trained on[38]. ProteinVAE-generated sequences closely resemble sequence patterns found in natural hexon populations. Notwithstanding, they consistently had more clusters and higher diversity than natural sequences at all thresholds (Fig. 3a). The ESM-IF1 model reached a higher level of diversity. This is due to the lower quality in the C-terminal half of the sequence, as this section of ESM-IF1-generated sequences were not likely to contain the hexon C-terminal domain and had higher entropy (Fig. 2c,d). Since the goal is to generate functional hexons that are biologically relevant, the ability of the model to diversify the sequences, while keeping high structure resemblance towards natural hexon protein, is critical. Sequence diversity was assessed for sequences with similar secondary structure ratios to natural sequences (in-distribution samples in Fig. 2f). Sequences with less than 80% target and query coverage when aligned to their closest natural sequence were removed. All in-distribution ProteinVAE samples

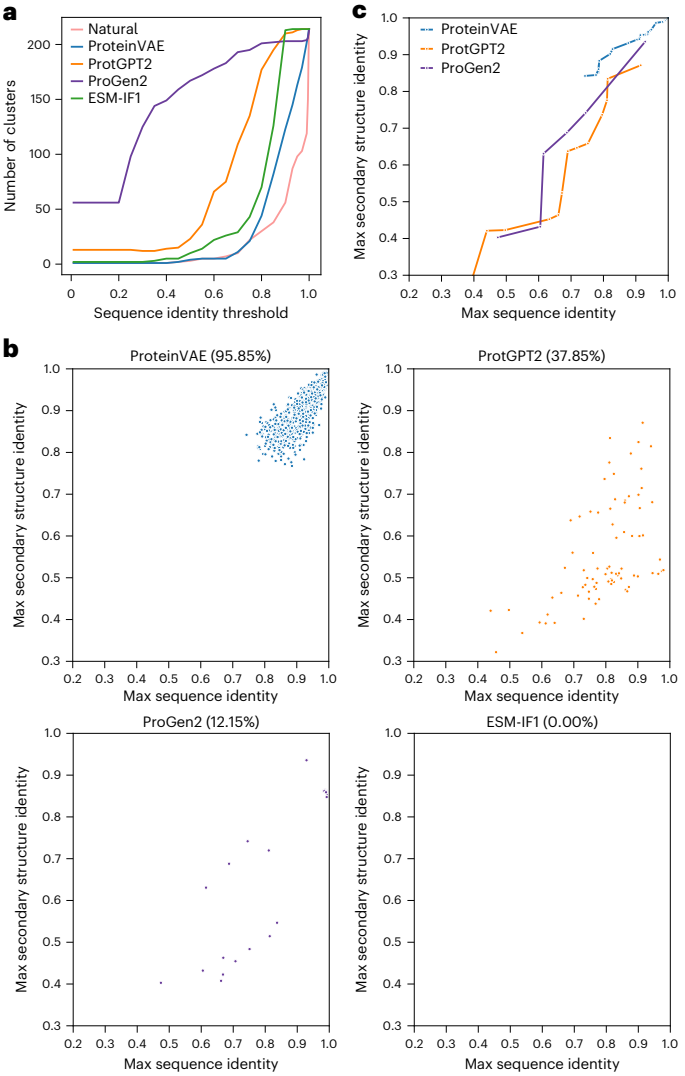


Fig. 3 | Comparing sequence diversity against sequence quality across models. a, Number of clusters at different identity thresholds. **b**, Scatter plot for sequence diversity and secondary structure similarity. The *x* axis is the maximum seqID on all aligned pairs. The *y* axis is the maximum percentage identity of three-state secondary structure on all aligned pairs of generated and natural sequences. Sequences closer to the top-left corner are ideal, as they are structurally similar to natural protein but more novel in sequence. **c**, Pareto frontiers: the optimal sequences designed by each model are highlighted along the respective frontier.

satisfy this sequence-level constraint, as they have been trained only on hexon sequences. In contrast, only 37.85% and 12.15% of samples are left in the ProtGPT2 and ProGen2 groups, respectively, which further supports the argument that ProtGPT2 and ProGen2 incorporate non-hexon fragments. While the absolute sequence identities (seqIDs) of ProteinVAE samples only cover the higher end of the range seen in other samples, all analysed ProteinVAE in-distribution samples have high structural similarity towards natural (Fig. 3b). Only the top 1,000 sequences with highest average positional probability were selected from 7,000 sequences generated by ProteinVAE, which ensures high sequence quality at the cost of lower diversity. There was no sequence in ESM-IF1-generated samples that satisfied the screening conditions, which is probably due to the low C-terminal sequence quality. To directly evaluate sequence diversity against structural similarity, we plotted the Pareto frontier of all generated samples, respectively, in Fig. 3c. In the comparable range, ProteinVAE produced samples more diverse without disruption of structural profile.

Molecular dynamics simulations confirm stable structure and interfaces

Molecular dynamics was used to assess structural stability. Natural hexon conformational sampling was obtained by clustering on natural sequences at 90% seqID and collecting the representative sequence from each cluster with more than ten sequences (13 clusters produced 13 representative sequences). A detailed generation process can be found in the Methods.

Three-dimensional structures for natural and model-generated sequences were predicted with Alphafold2. Hexons form homotrimers in the adenovirus capsid with extensive inter-subunit interactions (Extended Data Fig. 4). Thirteen representative natural hexon monomers consist of ~18% helix and 28% β-strand, with turns and coils making up the remaining 54% of the structure[50]. A comparison between the representative structure of natural hexon (A4ZKL6) and three ProteinVAE-designed hexons (with 91.5%, 85.6% and 75.4% identity to the closest natural hexon) indicated that they maintained the morphology and symmetry of the natural counterpart (Extended Data Fig. 4). All predicted structures were subject to 100 ns molecular dynamics, where root-mean-squared deviation (r.m.s.d.) has stabilized (Extended Data Fig. 5 and Supplementary Fig. 2). R.m.s.d. reveals that ProteinVAE-generated structures had a range similar to natural hexons (natural hexon: 1.14–4.53 Å; ProteinVAE samples: 1.21–6.58 Å), while the samples generated by ProtGPT2 (1.46–14.67 Å) and ProGen2 (1.56–8.67 Å) showed larger r.m.s.d. values (Extended Data Fig. 5). Root-mean-squared fluctuation (r.m.s.f.) to analyse local structural flexibility (Fig. 4) showed that ProGen2 introduced mutations that significantly increased flexibility in regions that were comparatively rigid in natural sequences, while those introduced by ProteinVAE did not. ProtGPT2 and ProGen2 also inserted long, highly flexible, potentially destabilizing fragments that are not homologous to natural hexons (Extended Data Fig. 6). As observed in natural MSA, the structurally exposed regions have higher evolutionary rate, and they are likely to be tolerant of mutations[51,52]. The same trend has been observed with artificially introduced mutations[15]. Mutations in ProteinVAE samples (Fig. 4d) are more likely to occur in these naturally exposed regions (Fig. 4e,f). ProteinVAE was able to generate diverse molecular dynamics stable sequences, with the most novel sequence containing 291 amino acids different from its closest natural sequence with 39.62% viral surface area changed. This degree of novelty illustrated ProteinVAE's generative capacity, while the considerably changed viral surface could increase chances of evading pre-existing serotype-specific antibodies.

ProteinVAE produces novel synthetic human AdV serotypes

To distinguish human adenovirus serotypes from generated sequences, a simple logistic regression classifier was trained from the encoder embeddings of all training data (364 human adenovirus hexon sequences and 347 non-human adenovirus hexon sequences). The validation area under the receiver operating characteristic curve of the trained classifier is 0.97 (Extended Data Fig. 7a and Supplementary Note 5), and the validation F1 score is 0.94. Sequences generated from each cluster were encoded and classified (Extended Data Fig. 7b). The percentage of generated sequences classified as human adenovirus hexon correlated with that of natural sequences in each cluster (Pearson's $r = 0.81$). Phylogenetic relationships were analysed between the 46 predicted human adenovirus hexon sequences, 65 hexons from unique human adenovirus serotypes in the training set and 20 randomly selected hexons with non-human host (Fig. 5a and Supplementary Note 6). A majority of the generated hexon sequences reside within the phylogenetic clades of human adenovirus species B and D, while preserving a substantial evolutionary divergence in relation to known serotypes. In addition, six generated sequences are separated from clades of known human species, but they still reside in the primate adenovirus clade (highlighted with a red curve in Fig. 5a). This suggested that they might represent novel human adenovirus species, or they might also be primate adenovirus hexons similar to human adenovirus.

Generated sequences were then aligned with every natural human adenovirus hexon. Amino acid divergence in loop 1 and loop 2 was calculated for each pair of sequences (Fig. 5b,c). These loops are considered to be the primary serotype determining hexon regions[53]. Generated sequences diverged more than 4.2% in loop 1 and more than 1.2% in loop 2 from any known serotypes, which defines them as hexons from new human adenovirus serotypes (Supplementary Note 7)[53].

ProteinVAE latent space allows interpolation

One benefit of using the VAE-based model is the ease of sampling provided by the structured VAE latent space ('Discussion'). To validate that evolutionary relationships and sequence similarities have been captured in the latent vectors, the ten largest clusters (at 90% seqID) were plotted in dimension-reduced hidden space (principal component analysis obtained) (Fig. 6a). Multiple clusters can be found in the hidden space distinctly separated. ProteinVAE hidden space also appears around the mean of 0 with no obvious hole. Next, interpolation was done between hexons from two interchangeable adenovirus serotypes, AdV2 and AdV5 (ref. 54). In total, 1,000 vectors were linearly interpolated between AdV5 and AdV2 hexon hidden vectors in ProteinVAE latent space, since this is a common approach to utilize VAE structure latent space[55,56]. These were decoded to sequences. As a control, another method[55] was implemented to sample between AdV5 and AdV2 hexon sequences directly. Both the control and latent interpolation achieved monotonic changes in Hamming distance (Fig. 6c,d). However, ProteinVAE latent interpolation allowed for generation of natural-resembling sequences, as indicated by higher average positional probability (Fig. 6b).

Discussion

ProteinVAE can learn the intrinsic relationships of long protein sequences from a limited number of samples and generated sequences which could be used to generate molecular dynamics stable structures. In addition, generated sequences are more diverse than natural sequences, capable of forming more clusters at the same identity threshold. Some ProteinVAE-generated hexons can be classified as new human adenovirus serotypes with imputed serotyping, providing meaningful candidate sequences for therapeutic applications.

Considerable efforts have been made toward computationally expanding known protein families with novel sequences. In conventional bioinformatics, hidden Markov models[21,57] were used with limited success due to their inability to learn the higher-order relationships in natural protein families.

More recently, deep learning models, including GANs[21,58], VAEs[19,22,23,42] and large generative protein language models[20,38,59], have been implemented to learn the complex constraints in biological sequence design. These methods have mostly focused on shorter protein sequences with many members from the same family. Because of this, they tend to perform poorly on large proteins with few members, such as AdV hexons. Challenges associated with generating diverse hexon sequences were demonstrated with the unsatisfactory performance of a fixed-backbone design model (ESM-IF1) and two recently published LLMs (ProtGPT2 and ProGen2) fine-tuned on the hexon dataset. Although the competitive fixed-backbone design method (ESM-IF1) showed more promising results on smaller proteins, with the current training sequence length range (500 amino acids) ESM-IF1 cannot generate high-quality hexon sequences at the C terminus. In addition, these models require significantly higher GPU memory for training and generating long sequences, as the inter-residue distance information requires a quadruple amount of memory for processing as the length increases. For instance, 24 GB of GPU memory is needed to generate one hexon sequence using ESM-IF1, while only 21 GB of GPU memory is needed to generate 1,000 hexon sequences in parallel

https://doi.org/10.1038/s42256-023-00787-2

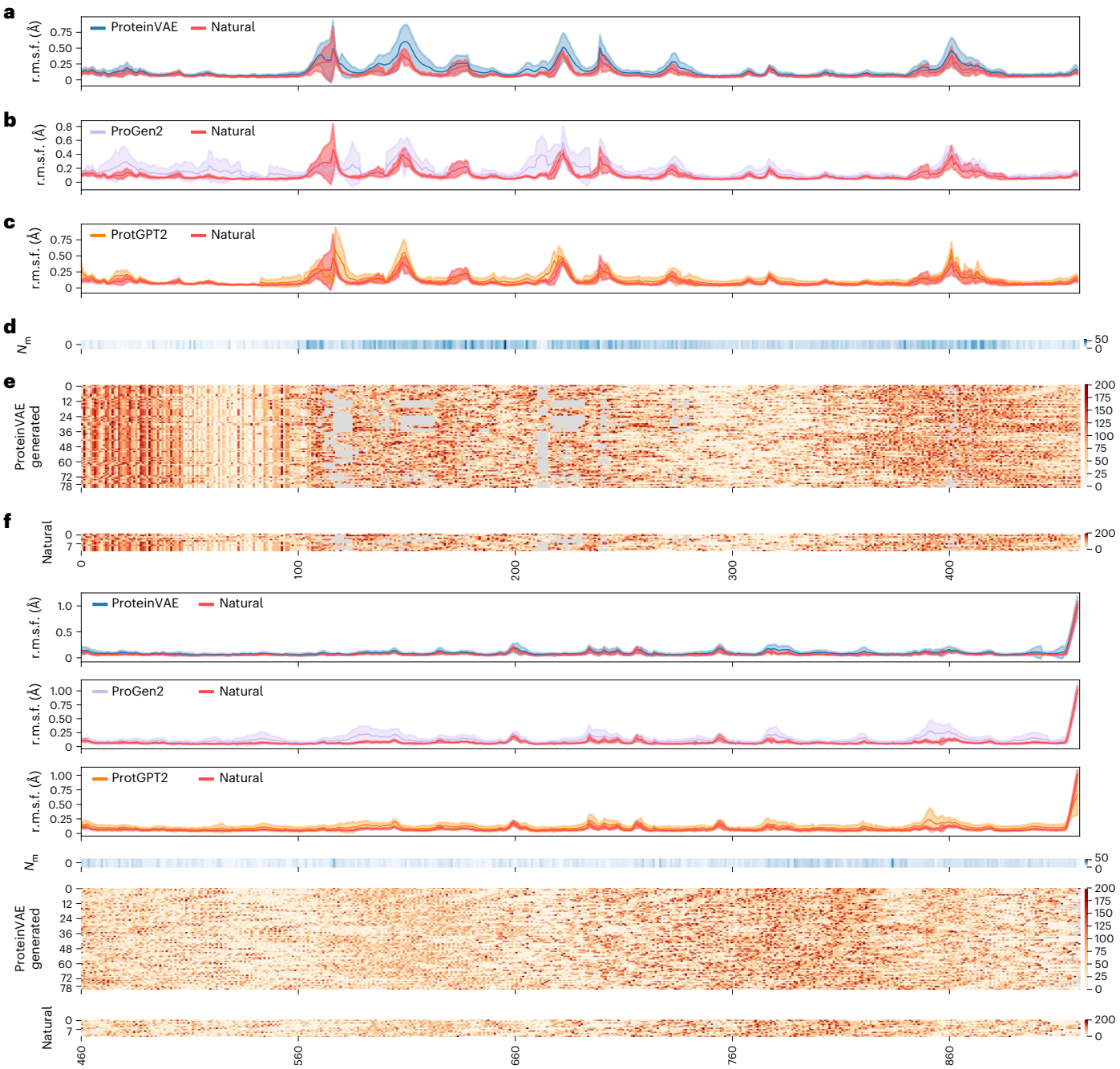


Fig. 4 | Molecular dynamics simulations. Top panels are the first half of the sequence length. Bottom panels are the second half of the sequence length. **a**, r.m.s.f. for all wild-type cluster representative sequences and stable ProteinVAE-generated sequences. **b**, r.m.s.f. for all wild-type cluster representative sequences and stable ProGen2-generated sequences. **c**, r.m.s.f. for all wild-type cluster representative sequences and stable ProtGPT2-generated sequences. Data in **a**–**c** are presented as mean ± s.d. **d**, Heatmap of positions where mutations were introduced in stable ProteinVAE-generated sequences compared with their closest natural sequence, respectively. **e**, Heatmap of solvent accessible area across all positions in each stable ProteinVAE sample. **f**, Heatmap of solvent accessible area across all positions in each natural sequence. N_m, number of mutations.

using ProteinVAE. For the LLMs, the fine-tuned ProtGPT2 performed better than the fine-tuned ProGen2 model. This is probably due to the higher number of parameters in the ProtGPT2 model. Larger ProGen2 models might generate better sequences, but even fine-tuning them is unfeasible with a standard 32 GB GPU on a dataset with long sequences (Methods). Improving ProtGPT2 and ProGen2 generation quality would require extensive efforts to be made in fixing the insertion of random sequences that the model retained from pre-training on a large database, since this is still happening in fine-tuned models. Instead, ProteinVAE distilled knowledge from a pre-trained protein language model and leveraged it to facilitate efficient learning of the complex sequence patterns from limited data. Moreover, the ProteinVAE model was able to generate 1,000 sequences in less than 1 min, while the generation of 1,000 sequences of similar length took ~12.5 min and 13.5 h for the ProtGPT2 and the ProGen2 models. Overall, ProteinVAE generated a higher ratio of diverse sequences that are structurally similar to natural hexons (Fig. 3). Lastly, the ProteinVAE-generated sequences analysed above were selected with an emphasis on sequence quality, which limited the range of diversity to some extent. In the future, less stringent selection criteria could be used to obtain more diverse sequences.

https://doi.org/10.1038/s42256-023-00787-2

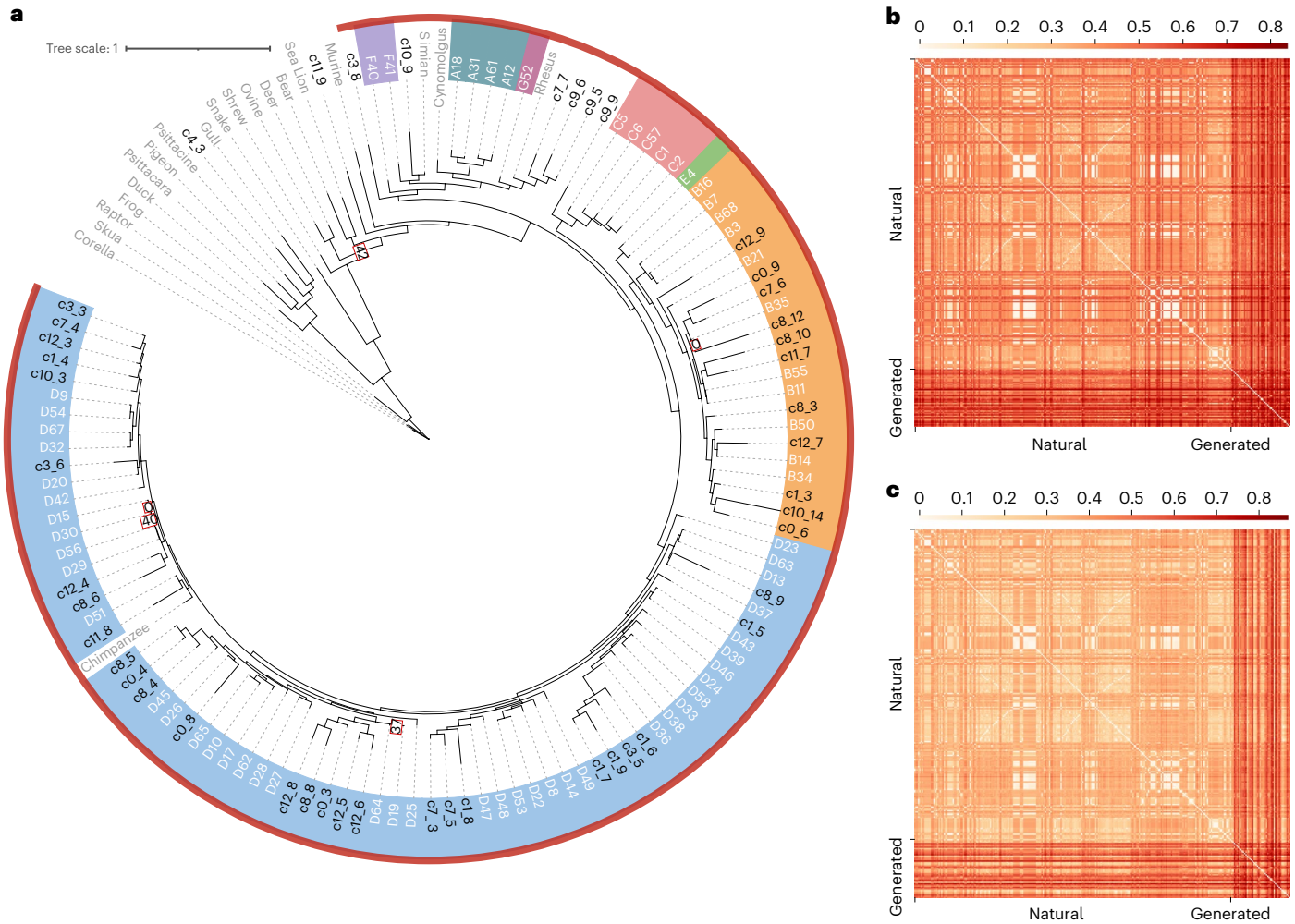


Fig. 5 | Phylogenetic analysis and imputed serotyping for generated human adenovirus hexon. a, Phylogenetic tree illustrating relationships among hexon protein sequences. The tree was constructed using the maximum likelihood method with BLOSUM62 substitutions model (see the Methods for details). Natural human adenovirus serotypes are denoted in white, with species identified by letters and serotypes by numbers. Generated sequences that are predicted to be human adenovirus hexon are displayed in black. Twenty randomly selected non-human adenovirus sequences are labelled in grey. Clades are coloured by human adenovirus species, while clades lacking known human adenoviruses remain uncoloured. Support values below 60 are shown in red boxes. The clade with primate adenovirus hexons is highlighted with a red curve on the outside. **b**, Pairwise amino acid divergence for the loop 1 region in hexons (columns 129–390 in MSA). **c**, Pairwise amino acid divergence for the loop 2 region in hexons (columns 447–540 in MSA). Darker shade means more difference between the compared sequences.

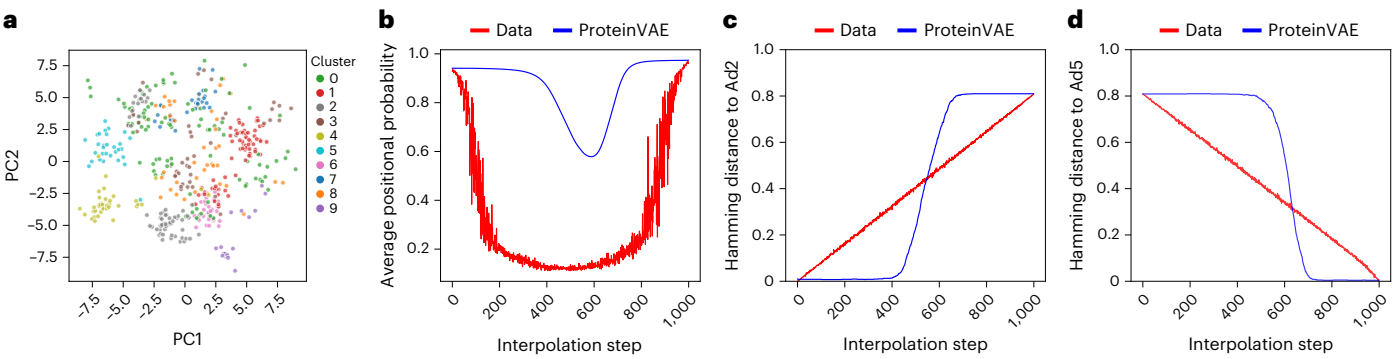


Fig. 6 | ProteinVAE latent space allows interpolation. a, Latent space clustering of sequences from the ten largest clusters at 90% identity. **b**, Average positional probability for latent space interpolated sequences and direct interpolated sequences. **c**, Hamming distance to AdV2 hexon. **d**, Hamming distance to AdV5 hexon. PC1, the first principal component; PC2, the second principal component.

Concurrent works, ProT-VAE[23] and ReLSO[17], both involved autoencoders and the use of a language model, but (1) neither presented results on designing proteins at the same length range as hexons; (2) both models were trained for a different objective of exploring fitness landscape and generating functionally improved sequences; and (3) both used larger labelled datasets (ProT-VAE: 6,447 and

20,000 sequences; ReLSO: 10^{10}, 20^4 and 51,175 sequences). ProT-VAE was trained to reconstruct the hidden state of a pre-trained protein language model. Since the ProT-VAE model has not been released, we simulated the reconstruction F1 in the ProT-VAE model by introducing a small Gaussian noise to the language model hidden state before decoding and found that ProT-VAE generation performance quickly worsened even at a low noise level (Supplementary Note 8 and Supplementary Fig. 10). The ReLSO model was designed with an autoencoder instead of a VAE architecture, and fitness information is jointly trained to be encoded in the latent space. It was trained on both positive and negative samples with a specially designed interpolation loss. The language model in the encoder was not pre-trained in ReLSO. The decoder is a deep convolutional network. To train ReLSO on the hexon dataset without the fitness label, the regression-related term was removed from the loss formulation. ReLSO-generated sequences are repetitive, and they suffer from low sequence and structural similarity to the training data as shown in various metrics (Supplementary Figs. 1 and 3–9, Supplementary Note 9 and Supplementary Table 3).

The capacity of ProteinVAE to learn the complex protein sequence distribution from limited samples could potentially be applied in a variety of different sequence design problems. To guide future model development, an ablation study was conducted to assess the impact of individual modules within ProteinVAE (Supplementary Note 10 and Supplementary Table 4). In the future, another model could be trained to map the ProteinVAE latent space to the protein fitness landscape and apply the ProteinVAE model to conditionally generate sequences with functional improvement[18]. Such computational exploration may facilitate exploration of distant regions of the fitness landscape where significant functional enhancement might be achieved.

Methods

Dataset

Hexon protein is the major capsid protein in adenovirus with a length spanning from 893 to 992 amino acids (average length: 938). To increase the chances of generating complete sequences covering all domains, only full-length hexon proteins annotated in the UniprotKB database[26,37] were collected. These sequences were then filtered for those shorter than 800 amino acids for quality purposes, and, for ease of downstream application, sequences with non-standard amino acids (U, J, Z, O, B, X) were removed. In total, 711 hexon sequences were collected. The same training/validation/test set splits ratio of 7/2/1 was used for all models. The same random seed was used for splitting (done with scikit-learn 1.2.2 (ref. 60)) in each replicate group, respectively.

VAE

The VAE[61] is composed of an encoder and a decoder. The encoder $q_\phi(z|x)$, a neural network parameterized by ϕ, maps the input data samples x into a latent variable z, assumed to follow a Gaussian distribution as its prior. The decoder $p_\theta(x|z)$, another neural network parameterized by θ, reconstructs the sample x from the latent variable z. The VAE is trained by maximizing the evidence lower bound ELBO, where:

$$\mathrm{ELBO} = E_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] - D_{\mathrm{KL}}\left(q_\phi(z|x)||p(z)\right)$$

$E_{q_\phi(z|x)}[\log p_\theta(x|z)]$ is the expected conditional log-likelihood. $p(z)$ is the prior Gaussian distribution; $D_{\mathrm{KL}}(q_\phi(z|x)||p(z))$ is the Kullback–Leibler (KL)-divergence. The details are described in the original publication[61].

In common text generation tasks, it has been demonstrated before that when KL-divergence decreases too much, the generated samples are likely to suffer from low diversity[62]. To prevent KL-vanishing and to allow effective manipulation of the impact of KL-divergence, a nonlinear proportional-integral-derivative controller was implemented to automatically tune the weight of KL-divergence in the VAE objectives throughout training. The KL-divergence weight $\gamma(t)$ is calculated through a feedback control defined as:

$$\gamma(t) = K_\mathrm{p}e(t) + K_\mathrm{i}\int_0^t e(\tau)\mathrm{d}\tau + K_\mathrm{d}\frac{\mathrm{d}e(t)}{\mathrm{d}t}$$

$e(t)$ is the error between actual and expected value at time t. K_p, K_i and K_d are the coefficients for the proportional, integral and derivative terms, respectively. See details in ref. 63.

Bottleneck encoder

To refine the global representation, the bottleneck attention module was used. This special attention module is defined as:

$$\beta(H;\delta) = \mathrm{MultiHead}(q, K, V)$$

where the keys K (size $T \times d$) and values V (size $T \times d$) are transformed from the output hidden representations of the pre-trained language model H ($T \times d$). The parameter δ includes the weights for transformation of the query, keys and values. During training, the pre-trained language model stays frozen, and only the parameter δ is learned.

Restricted by the length of hexon, only limited pre-trained language models are available. ProtBert (420 million parameters)[30] was chosen as the pre-trained protein language model, as (1) it is trained with a length limit of 2,048 amino acids and (2) it achieved better results on downstream tasks than two other models trained on long protein sequences. In brief, the ProtBert used in ProteinVAE contains 30 layers, and it was trained for 300,000 steps on sequences shorter than 512 amino acids, then for an additional 100,000 steps on sequences with a maximum length of 2,000 (ref. 30).

Non-autoregressive decoder

As mentioned in the introduction, the hypothesis was that the protein sequence can be better modelled with a non-autoregressive processing approach. Thus, inspired by the HybridVAE model[33], deconvolution networks were used to perform upsampling. The deconvolutional network increases the spatial size of the input, while decreasing the number of hidden dimensions. Specifically, the deconvolutional networks consist of eight UpBlocks (Extended Data Fig. 1). In each UpBlock, a 1×1 convolutional layer transforms the input to have a lower number of channels, which reduces the number of parameters needed; in the next layer, a 3×3 deconvolutional layer upsamples the low-channel input. To maintain the gradient, the output of each previous UpBlock was concatenated as the input for the next block. Unlike HybridVAE, the deconvolutional networks output M was not passed to a recurrent neural network. Instead, a multi-head attention module was used to capture both short- and long-range relationships. Next, the output was converted to logits using a linear layer. Lastly, another amino-acid-wise attention module was added to capture the amino acid usage preference among different viruses[35]. As a comparison, classic autoregressive processing was tested by replacing the deconvolutional networks with a multi-layer long short-term memory (LSTM) recurrent neural network[64]. Both single direction and bidirectional LSTM models are tested. LSTM hidden sizes are divided by 2 when testing bidirectional LSTM. The hidden dimensions of the bottleneck representation z and the upsampled decoder hidden representation M were kept the same.

ProteinVAE training

The ProteinVAE model was implemented with Pytorch v.1.12.1 (ref. 65) and Pytorch-lightning v.1.6.5 (ref. 66). Training was monitored with wandb v.0.15.0 (ref. 67). The ProteinVAE model was trained on the hexon dataset using negative ELBO loss. KL-divergence was dynamically weighted using a proportional-integral-derivative controller with expected KL-divergence of 0.5, K_p of 0.01, K_i of 0.0001 and K_d of 0.001. Strand position was weighted to have 1.2× cross-entropy loss.

The ProteinVAE model was optimized using the Adam optimizer[68] with a learning rate of 0.0005 and weight decay of 0.0001. Dropout rate of 0.3 was used. A one-cycle learning rate scheduler[69] was used with total steps 8,000, percentage of the cycle (in number of steps) spent increasing the learning rate was set to 0.4 and the initial learning rate was set to 1/20 of peak learning rate. Decoder and encoder latent sizes were set to 128. In each of the eight decoder UpBlocks, upsampling was done after input was transformed to 16 channels. Encoder bottleneck attention has four heads. Decoder position-wise attention has two heads, and the decoder amino-acid-wise attention also has two heads. To prevent overfitting, the training was stopped with early stopping when validation cross-entropy loss had not improved in the last 250 epochs, and the checkpoint with the highest test F1 (calculated with torchmetrics v.0.8.1 (ref. 70)) was used for generation. The ProteinVAE model was trained on an NVIDIA V100 GPU with 32 GB of memory.

ProteinVAE sequence generation

Generate samples for MD analysis and synthetic human AdV analysis. For each of the top 13 clusters (90% identity, size > 10), 50,000 sequences were generated with the mean of the respective cluster and standard deviation of 3. Within each cluster, sequences more repetitive than the most repetitive natural sequence in that cluster were filtered out. For the rest of the sequences, each one is aligned against the whole natural hexon dataset to get the percentage identity towards the closest natural protein. A bin width of 2% was used to separate sequences of different novelty. For the 17 bins with percentage identity from 60% to 92% (some bins are empty), the sequence with the highest APP was selected for Alphafold2 structure prediction. APP was calculated as:

$$\mathrm{APP} = \frac{1}{L}\sum_{i=1}^{L}\max_{1\le j\le 21} p_{ij}$$

where p_{ij} is the predicted probability of amino acid j (including a special token '−' representing a gap in sequence) at the ith position. L is the maximum length of training sequences. Predicted local-distance difference test (pLDDT) threshold was not benchmarked, because there are experimental structures for only five human adenovirus hexons, and four non-human adenovirus hexons. Instead, the threshold was set at 85% (higher than the general recommended value of 70%) to obtain more accurate structures. In total, 102 structures with an average pLDDT score of higher than 85% were selected for molecular dynamic analysis. Constructs are named as c_{i_j}, where i represents the ith cluster and j signifies the jth bin.

All other samples are generated following this procedure. To achieve a balance between diversity and quality, a larger batch of vectors were sampled with a higher standard deviation, and only high quality sequences were selected. Seven thousand vectors were sampled from a normal distribution with mean of 0 and standard deviation of 4 and decoded to new sequences. To maintain high sequence quality, sequences were ranked according to APP, and we selected the top $\frac{1}{7}$ sequences for downstream analysis.

MSA entropy

Clustal Omega v.1.2.4 (ref. 71) was used to calculate MSA for the entire natural hexon dataset mixed with an equal number of generated sequences. Columns with more than 80% gaps in either the natural or generated dataset were removed. Shannon entropy within each column was calculated as:

$$\mathrm{SE} = -\sum_{i=1}^{20} p(x_i)\log_{20} p(x_i)$$

where $p(x_i)$ is the frequency of amino acid i in each column. Pearson correlation was calculated between valid entropy values of natural and generated sequences.

Association measure for amino acid pairs

For any pair of amino acids a and b, the minimal proximity score and the pair association metrics were calculated as described in the original publication[43]. Distance between each occurrence of a at position x_i with its nearest occurrence of b at position y_i was computed, and then averaged across all occurrences of a:

$$P_{\mathrm{m}}(a,b) = \frac{1}{n}\sum_{i=1}^{n}\min_{j=1,\ldots,m}\{|x_i - y_i|\}$$

To remove dependency on the number of occurrences for different amino acids, the position of a was fixed and b was randomly shuffled (Rand(b) stands for a random array of positions for b). Mean ($\overline{P_{\mathrm{m}}(a,\mathrm{Rand}(b))}$) and standard deviation ($\sigma_{P_{\mathrm{m}}(a,\mathrm{Rand}(b))}$) of the randomly shuffled sequences (where position of b is shuffled, but position of a is fixed) were calculated, and the minimal proximity score was normalized to obtain the association score as described in the original publication[43].

$$Z_{\mathrm{m}}(a,b) = \frac{P_{\mathrm{m}}(a,b) - \overline{P_{\mathrm{m}}(a,\mathrm{Rand}(b))}}{\sigma_{P_{\mathrm{m}}(a,\mathrm{Rand}(b))}}$$

For the score shown in Fig. 2, averaged association scores were plotted for each group of sequences. A null value was assigned if a pair of amino acids did not exist in a sequence.

Sequence clustering and secondary structure analysis

Sequence clustering was done at different identity thresholds using MMSeqs2 (release 24 February 2021)[72]. For sequence homology detection, default MMseq2 settings were used to perform pairwise alignment between all possible pairs of generated and natural sequences in training data. To assess the structural similarity, SPOT-1D (original release)[36] was used to predict three-state (helix, strand, coil) secondary structure for all amino acids. For outlier detection in secondary structure ratio, given the sum of all three ratios is 1, a bi-variate Gaussian distribution was fitted only on the helix and strand percentage of natural hexons. Mahalanobis distance (d)[73,74] was calculated between all generated samples and the centre of bi-variate Gaussian distribution. Since d^2 follows Chi-squared distribution, a critical value $\alpha = 0.05$ was used to determine the cut-off distance. Samples with the smallest Mahalanobis distance and the smallest maximum seqID were identified to form the Pareto front[75].

SASA analysis

Due to the high computational resource requirement, only a portion of sequences were selected to calculate the SASA profile. In total, 100 natural sequences were randomly selected and folded to get the SASA of each amino acid in natural hexons. The 100 sequences were randomly selected from the 1,000 ProteinVAE-generated sequences used in previous sequence and structural pattern analysis as representatives. Structures were predicted with Alphafold v.2.0.0, and SASA was calculated with FreeSASA v.2.1.0 (ref. 76). Since the number of an amino acid might differ between the natural and generated samples, unpaired two-sided Welch t-test was used to compare the mean SASA for each amino acid[77]. A P value smaller than $\alpha = 0.05$ is classified as statistically different. The t-test was implemented with Pingouin python library (v.0.5.2)[78].

Latent space clustering and interpolation

To visualize the latent space, principal component analysis[79] was used to reduce the number of dimensions to 2. The ten biggest clusters (457 sequences in total) at 90% seqID in natural hexon sequences were used for plotting.

For interpolation, 1,000 points were linearly sampled between the hidden vectors of adenovirus 2 hexon and adenovirus 5 hexon. Hidden codes were passed through a decoder to get the predicted probabilities.

At each position, the token with the highest logit was chosen, and average positional probability was calculated as previously described.

As a control, direct interpolation between two sequences was done, by sampling a Bernoulli random variable with α probability to choose amino acid from adenovirus 2 hexon ($1-\alpha$ probability from adenovirus 5 hexon) at each position. In total, 1,000 different α values were linearly selected from 0 to 1. Ten sequences were sampled at each α, resulting in 10,000 sequences in total. ProteinVAE was run on the directly interpolated samples, and the predicted probabilities were collected. The APP is calculated on probabilities for amino acids in the input sequence, instead of the amino acids with highest probability at each position.

ProtGPT2 fine-tuning and sampling

The ProtGPT2 model (original release) was fine-tuned on our training data. Due to GPU memory limitation, eight AMD MI50-32GB GPUs were used in parallel, with a total effective batch size of 16. Learning rate from 10^{-1} to 10^{-6} was tested with weight decay from 0 to 10^{-6} (Supplementary Table 5). To prevent overfitting, the training was stopped with early stopping when validation loss had not improved in the last ten epochs, and the checkpoint with the lowest validation perplexity was selected for evaluation. The model with the lowest test perplexity was used for generation (learning rate: 10^{-3}; weight decay: 0). The fine-tuned model was prompted with 'M' at the start of the sentence, and top_k sampling was used with the parameters as suggested in the original publication (top_k: 950; repetition penalty: 1.2)[38]. It was observed that generation performance drastically worsened with inclusion of any token with 'X', and all such tokens were removed. For the language models, even after fine-tuning, ProtGPT2 tends to generate shorter sequences without the minimal token criteria. To prevent generation of short sequences, the range of tokens allowed in a sequence was set to 300–350, as seen in tokenization of natural hexon sequences. Inference of 25 sequences was repeated for 66 batches (1,650 sequences in total; ~12.5 min inference time), until 1,500 sequences within the length range of hexon were accumulated. Sequences were ranked according to their perplexity[80], and we kept only the top $\frac{1}{7}$ for comparison. To accommodate downstream analysis, 'Z' and 'B' found in ProtGPT2-generated sequences were replaced with appropriate standard amino acids.

ProGen2 fine-tuning and sampling

The ProGen2-small model was fine-tuned on our training data. Other larger ProGen2 models have drastically high GPU memory requirement; out of memory error would occur even with a batch size smaller than 2 on a 32 GB GPU. Eight AMD MI50-32GB GPUs were used in parallel for fine-tuning, with a total effective batch size of 48. Learning rate from 6×10^{-3} to 6×10^{-6} was tested with weight decay from 0 to 10^{-6} (Supplementary Table 6). To prevent overfitting, the training was stopped with early stopping when validation loss had not improved in the last ten epochs, and the checkpoint with the lowest validation perplexity was selected for evaluation. The model with the lowest test perplexity was used for generation (learning rate: 6×10^{-4}; weight decay: 10^{-4}). The fine-tuned model was prompted with 'M' at the start of the sentence; nucleus sampling was used. Since no generation parameter was suggested as optimal in the original publication[20], top_p (0.7–1.0) and temperature (0.2–1.0) were optimized according to the log-likelihood of generated sequences (Supplementary Table 7). Top_p of 0.7 and a temperature of 0.4 were used for the final generation. To generate sequences within the length range, the maximum number of tokens allowed in a sequence was set to 992. Inference of 20 sequences was repeated for 90 batches (1,800 sequences in total; ~13.5 h inference time), until 1,500 sequences within the length range of hexon were accumulated. Sequences were ranked according to their perplexity[80], and we kept only the top $\frac{1}{7}$ for comparison.

ESM-IF1 generation

In total, 100 natural hexons were randomly selected from the training data, and their structures were predicted with Alphafold2 as described above. There are only 20 hexon structures in the PDB database (collected on 30 August 2023), and many of the structures are for the same reference human AdV5 hexon. The nine non-redundant sequence–structure pairs were not sufficient for fine-tuning of ESM-IF1. Generation was also attempted, but compared with using the predicted structures, sequence likelihood decreased. For both types of templates, sampling temperatures were optimized (Supplementary Tables 8 and 9). Temperature of 0.1 was used in the final generation, as only insubstantial improvements in likelihood were observed with lower temperatures, while the diversity decreased drastically. For each computationally obtained structure template, 15 sequences were generated with ESM-IF (original release). The 1,500 generated sequences were ranked according to log-likelihood. As long repetition (for example, EEEEEE) is a known failure mode[39], sequences with single-amino-acid repetition longer than six amino acids were filtered out. Repetition of bi-gram or tri-gram (for example, KDKDKD) was also seen, and affected sequences were removed. The top $\frac{1}{7}$ sequences were used for comparison.

Molecular dynamics simulation setup

ProteinVAE samples were selected as described in previous sections. For comparison, sequences generated by ProtGPT2 and ProGen2 were randomly chosen if they exceeded 80% in both query and target coverage when aligned with the closest natural sequence and were within the natural distribution of helix-to-strand ratio. To increase diversity, samples were selected from all 1,500 generated sequences. From each model, one generated sequence was randomly selected in each 3% identity range from 70% to 100% (10 ranges; 8 ProGen2 sequences, 2 ranges were empty; 10 ProtGPT2 sequences).

Input structures were used to build the protein representation using CHARMM-GUI v.3.8 online server[81]. Systems were solvated in an explicit TIP3P (ref. 82) water box. Charge neutrality was maintained by addition of counter ions, and physiological condition was mimicked using 0.15 M KCl.

All systems were energy minimized using the steepest descent before pre-equilibration phase, which was conducted for 500 ps under the constant number of particles, volume and temperature condition. Production phase was carried out for 100 ns. To ensure the stability, five randomly selected systems including four variants and one wild type were simulated for another 200 ns (total of 300 ns) and their r.m.s.d. and r.m.s.f. values were compared with their 100 ns simulated structures. All five systems showed negligible changes in r.m.s.d. and r.m.s.f. upon simulation time extension (Supplementary Fig. 2). The particle-mesh Ewald[83,84] method was used with a cut-off radius of 1.2 nm for long-range electrostatic interactions. Heavy atom–hydrogen atom bonds were constrained using the parallel linear constraint solver (P-LINCS) algorithm[85]. The Nosé–Hoover thermostat[86] with a coupling time constant of 1 ps and the Parrinello–Rahman barostat[87] with a coupling time constant of 5 ps were used for the production phase. A reference coupling pressure of 1 bar and a compressibility of 4.5×10^{-5} bar^{-1} were used. For all simulations, periodic boundary conditions were applied in all directions. Simulations were carried out using CHARMM36m force field[88] by GROMACS/2021.3 (ref. 89). Structure visualization (Extended Data Fig. 4) was done using Protein Imager v.0.5.60 (ref. 90).

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

Sequences of all 711 natural hexons can be found at /data/hexon_711.fasta in the CodeOcean capsule (https://doi.org/10.24433/CO.2530457.

v2 (ref. 91)). All natural hexon sequences were downloaded from the UniprotKB[26,37] database. Source data are provided with this paper.

Code availability

The code is provided at https://doi.org/10.24433/CO.2530457.v2 (ref. 91). ProtBert is used for extracting embeddings, and its code can be accessed at https://huggingface.co/Rostlab/prot_bert.

References

1. Vokinger, K.N., Glaus, C.E.G. & Kesselheim, A.S. Approval and therapeutic value of gene therapies in the US and Europe. *Gene Ther.* **30**, 756–760 (2023).
2. Mendell, J. R. et al. Single-dose gene-replacement therapy for spinal muscular atrophy. *N. Engl. J. Med.* **377**, 1713–1722 (2017).
3. Claussnitzer, M. et al. A brief history of human disease genetics. *Nature* **577**, 179–189 (2020).
4. Seregin, S. S. & Amalfitano, A. Overcoming pre-existing adenovirus immunity by genetic engineering of adenovirus-based vectors. *Expert Opin. Biol. Ther.* **9**, 1521–1531 (2009).
5. Verdera, H. C., Kuranda, K. & Mingozzi, F. AAV vector immunogenicity in humans: a long journey to successful gene transfer. *Mol. Ther.* **28**, 723–746 (2020).
6. Zhao, Z., Anselmo, A. C. & Mitragotri, S. Viral vector-based gene therapies in the clinic. *Bioeng. Transl. Med.* **7**, e10258 (2022).
7. Bulcha, J. T., Wang, Y., Ma, H., Tai, P. W. & Gao, G. Viral vector platforms within the gene therapy landscape. *Signal Transduct. Target. Ther.* **6**, 1–24 (2021).
8. Bouvet, M. et al. Adenovirus-mediated wild-type p53 tumor suppressor gene therapy induces apoptosis and suppresses growth of human pancreatic cancer. *Ann. Surg. Oncol.* **5**, 681–688 (1998).
9. Chillon, M. et al. Group D adenoviruses infect primary central nervous system cells more efficiently than those from group C. *J. Virol.* **73**, 2537–2540 (1999).
10. Stevenson, S. C., Rollence, M., Marshall-Neff, J. & McClelland, A. Selective targeting of human cells by a chimeric adenovirus vector containing a modified fiber protein. *J. Virol.* **71**, 4782–4790 (1997).
11. Xiang, Z. et al. Chimpanzee adenovirus antibodies in humans, sub-Saharan Africa. *Emerg. Infect. Dis.* **12**, 1596 (2006).
12. D'ambrosio, E., Del Grosso, N., Chicca, A. & Midulla, M. Neutralizing antibodies against 33 human adenoviruses in normal children in Rome. *Epidemiol. Infect.* **89**, 155–161 (1982).
13. Sumida, S. M. et al. Neutralizing antibodies to adenovirus serotype 5 vaccine vectors are directed primarily against the adenovirus hexon protein. *J. Immunol.* **174**, 7179–7185 (2005).
14. Lee, C. S. et al. Adenovirus-mediated gene delivery: potential applications for gene and cell-based therapies in the new era of personalized medicine. *Genes Dis.* **4**, 43–63 (2017).
15. Ogden, P. J., Kelsic, E. D., Sinai, S. & Church, G. M. Comprehensive AAV capsid fitness landscape reveals a viral gene and enables machine-guided design. *Science* **366**, 1139–1143 (2019).
16. Sarkisyan, K. S. et al. Local fitness landscape of the green fluorescent protein. *Nature* **533**, 397–401 (2016).
17. Castro, E. et al. Transformer-based protein generation with regularized latent space optimization. *Nat. Mach. Intell.* **4**, 840–851 (2022).
18. Ding, X., Zou, Z. & Brooks, C. L. III. Deciphering protein evolution and fitness landscapes with latent space models. Nat. Commun. 10, 5644 (2019).
19. Hawkins-Hooker, A. et al. Generating functional protein variants with variational autoencoders. *PLoS Comput. Biol.* **17**, e1008736 (2021).
20. Nijkamp, E., Ruffolo, J. A., Weinstein, E. N., Naik, N. & Madani, A. ProGen2: exploring the boundaries of protein language models. *Cell Syst.* **14**, 968–978 (2023).
21. Repecka, D. et al. Expanding functional protein sequence spaces using generative adversarial networks. *Nat. Mach. Intell.* **3**, 324–333 (2021).
22. Riesselman, A. J., Ingraham, J. B. & Marks, D. S. Deep generative models of genetic variation capture the effects of mutations. *Nat. Methods* **15**, 816–822 (2018).
23. Sevgen, E. et al. ProT-VAE: Protein Transformer Variational AutoEncoder for functional protein design. Preprint at *bioRxiv* https://doi.org/10.1101/2023.01.23.525232 (2023).
24. Sinai, S., Jain, N., Church, G. M. & Kelsic, E. D. Generative AAV capsid diversification by latent interpolation. Preprint at *bioRxiv* https://doi.org/10.1101/2021.04.16.440236 (2021).
25. Dhingra, A. et al. Molecular evolution of human adenovirus (HAdV) species C. *Sci. Rep.* **9**, 1039 (2019).
26. Consortium, U. UniProt: a hub for protein information. *Nucleic Acids Res.* **43**, D204–D212 (2015).
27. Bejani, M. M. & Ghatee, M. A systematic review on overfitting control in shallow and deep neural networks. *Artif. Intell. Rev.* **54**, 6391–6438 (2021).
28. Montero, I., Pappas, N. & Smith, N. A. Sentence bottleneck autoencoders from transformer language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (2021).
29. Khandelwal, U., Clark, K., Jurafsky, D. & Kaiser, L. Sample efficient text summarization using a single pre-trained transformer. Preprint at https://arxiv.org/abs/1905.08836 (2019).
30. Elnaggar, A. et al. ProtTrans: towards cracking the language of life's code through self-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 7112–7127 (2022).
31. Holtzman, A., Buys, J., Du, L., Forbes, M. & Choi, Y. The curious case of neural text degeneration. In *International Conference on Learning Representations* (2019).
32. Tan, B., Yang, Z., AI-Shedivat, M., Xing, E. P. & Hu, Z. Progressive generation of long text with pretrained language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (2021).
33. Semeniuta, S., Severyn, A. & Barth, E. A hybrid convolutional variational autoencoder for text generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (2017).
34. Iandola, F. et al. DenseNet: implementing efficient ConvNet descriptor pyramids. Preprint at https://arxiv.org/abs/1404.1869 (2014).
35. Bahir, I., Fromer, M., Prat, Y. & Linial, M. Viral adaptation to host: a proteome-based analysis of codon usage and amino acid preferences. *Mol. Syst. Biol.* **5**, 311 (2009).
36. Hanson, J., Paliwal, K., Litfin, T., Yang, Y. & Zhou, Y. Improving prediction of protein secondary structure, backbone angles, solvent accessibility and contact numbers by using predicted contact maps and an ensemble of recurrent and residual convolutional neural networks. *Bioinformatics* **35**, 2403–2410 (2019).
37. Boutet, E. et al. UniProtKB/Swiss-Prot, the manually annotated section of the UniProt KnowledgeBase: how to use the entry view. In *Plant Bioinformatics: Methods and Protocols* Vol. 1374 (ed. Edwards, D) (Humana Press, 2016).
38. Ferruz, N., Schmidt, S. & Höcker, B. ProtGPT2 is a deep unsupervised language model for protein design. *Nat. Commun.* **13**, 4348 (2022).
39. Hsu, C. et al. Learning inverse folding from millions of predicted structures. In *Proc. Int. Conf. Mach. Learn.* (eds Chaudhuri, K. et al.) 8946–8970 (PMLR, 2022).
40. Jeliazkov, J. R., del Alamo, D. & Karpiak, J. D. ESMFold hallucinates native-like protein sequences. Preprint at *bioRxiv* https://doi.org/10.1101/2023.05.23.541774 (2023).

41. Lin, Z. et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* **379**, 1123–1130 (2023).
42. Sinai, S., Kelsic, E., Church, G. M. & Nowak, M. A. Variational auto-encoding of protein sequences. Preprint at https://arxiv.org/abs/1712.03346 (2017).
43. Santoni, D., Felici, G. & Vergni, D. Natural vs. random protein sequences: discovering combinatorics properties on amino acid words. *J. Theor. Biol.* **391**, 13–20 (2016).
44. Zheng, L. et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. Preprint at https://arxiv.org/abs/2306.05685 (2023).
45. Wang, Y. et al. How far can camels go? Exploring the state of instruction tuning on open resources. Preprint at https://arxiv.org/abs/2306.04751 (2023).
46. Li, R., Patel, T. & Du, X. PRD: peer rank and discussion improve large language model based evaluations. Preprint at https://arxiv.org/abs/2307.02762 (2023).
47. Eddy, S. R. Accelerated profile HMM searches. *PLoS Comput. Biol.* **7**, e1002195 (2011).
48. Jorda, J., Xue, B., Uversky, V. N. & Kajava, A. V. Protein tandem repeats—the more perfect, the less structured. *FEBS J.* **277**, 2673–2682 (2010).
49. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
50. Drew, E. D. & Janes, R. W. PDBMD2CD: providing predicted protein circular dichroism spectra from multiple molecular dynamics-generated protein structures. *Nucleic Acids Res.* **48**, W17–W24 (2020).
51. Echave, J., Spielman, S. J. & Wilke, C. O. Causes of evolutionary rate variation among protein sites. *Nat. Rev. Genet.* **17**, 109–121 (2016).
52. Franzosa, E. A. & Xia, Y. Structural determinants of protein evolution are context-sensitive at the residue level. *Mol. Biol. Evol.* **26**, 2387–2395 (2009).
53. Madisch, I., Harste, G., Pommer, H. & Heim, A. Phylogenetic analysis of the main neutralization and hemagglutination determinants of all human adenovirus prototypes as a basis for molecular classification and taxonomy. *J. Virol.* **79**, 15265–15276 (2005).
54. Youil, R. et al. Hexon gene switch strategy for the generation of chimeric recombinant adenovirus. *Hum. Gene Ther.* **13**, 311–320 (2002).
55. Roberts, A., Engel, J., Raffel, C., Hawthorne, C. & Eck, D. A hierarchical latent vector model for learning long-term structure in music. In *Proc. Int. Conf. Mach. Learn.* (eds Dy, J. & Krause, A) 4364–4373 (PMLR, 2018).
56. Wang, R. E., Durmus, E., Goodman, N. &Hashimoto, T. Language modeling via stochastic processes. In *International Conference on Learning Representations* (2021).
57. Russ, W. P. et al. An evolution-based model for designing chorismate mutase enzymes. *Science* **369**, 440–445 (2020).
58. Goodfellow, I. et al. Generative adversarial networks. *Commun. ACM* **63**, 139–144 (2020).
59. Madani, A. et al. Large language models generate functional protein sequences across diverse families. *Nat. Biotechnol.* **41**, 1099–1106 (2023).
60. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
61. Kingma, D. P. & Welling, M. Auto-encoding variational bayes. Preprint at https://arxiv.org/abs/1312.6114 (2013).
62. Bowman, S. R. et al. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning* (ACL, 2016).
63. Shao, H. et al. Controlvae: controllable variational autoencoder. In *Proc. Int. Conf. Mach. Learn.* (eds Daumé, H. III & Singh, A) 8655–8664 (PMLR, 2020).
64. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997).
65. Paszke, A. et al. Pytorch: an imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **32**, 8024–8035 (2019).
66. Falcon, W. & The PyTorch Lightning team. PyTorch Lightning. *Zenodo* https://doi.org/10.5281/zenodo.3828935 (2019).
67. Biewald, L. Experiment tracking with weights and biases. *Weights & Biases* https://www.wandb.com/ (2020).
68. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations (ICLR'15)* (2015).
69. Smith, L. N. & Topin, N. Super-convergence: very fast training of neural networks using large learning rates. In *Proc. Vol. 11006. Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications* 369–386 (SPIE, 2019).
70. Detlefsen, N. S. et al. TorchMetrics—measuring reproducibility in PyTorch. *Journal of Open Source Software* **7**, 4101 (2022).
71. Sievers, F. & Higgins, D. G. Clustal Omega for making accurate alignments of many protein sequences. *Protein Sci.* **27**, 135–145 (2018).
72. Steinegger, M. & Söding, J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.* **35**, 1026–1028 (2017).
73. Etherington, T. R. Mahalanobis distances and ecological niche modelling: correcting a chi-squared probability error. *PeerJ* **7**, e6678 (2019).
74. Mahalanobis, P. C. On the generalized distance in statistics. *Proc. of the National Institute of Science of India* **2**, 4955 (1936).
75. Teich, J. Pareto-front exploration with uncertain objectives. In *International Conference on Evolutionary Multi-Criterion Optimization* (eds Zitzler, E., Thiele, L., Deb, K., Coello Coello, C.A., Corne, D.) 314–328 (Springer, 2001).
76. Mitternacht, S. FreeSASA: an open source C library for solvent accessible surface area calculations. *F1000Research* **5**, 189 (2016).
77. Zimmerman, D. W. A note on preliminary tests of equality of variances. *Br. J. Math. Stat. Psychol.* **57**, 173–181 (2004).
78. Vallat, R. Pingouin: statistics in Python. *J. Open Source Softw.* **3**, 1026 (2018).
79. Abdi, H. & Williams, L. J. Principal component analysis. *Wiley Interdiscip. Rev. Comput. Stat.* **2**, 433–459 (2010).
80. Jelinek, F., Mercer, R. L., Bahl, L. R. & Baker, J. K. Perplexity—a measure of the difficulty of speech recognition tasks. *J. Acoust. Soc. Am.* **62**, S63 (1977).
81. Lee, J. et al. CHARMM-GUI input generator for NAMD, GROMACS, AMBER, OpenMM, and CHARMM/OpenMM simulations using the CHARMM36 additive force field. *J. Chem. Theory Comput.* **12**, 405–413 (2016).
82. Jorgensen, W. L., Chandrasekhar, J., Madura, J. D., Impey, R. W. & Klein, M. L. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **79**, 926–935 (1983).
83. Darden, T., York, D. & Pedersen, L. Particle mesh Ewald: an N· log (N) method for Ewald sums in large systems. *J. Chem. Phys.* **98**, 10089–10092 (1993).
84. Essmann, U. et al. A smooth particle mesh Ewald method. *J. Chem. Phys.* **103**, 8577–8593 (1995).
85. Hess, B. P-LINCS: a parallel linear constraint solver for molecular simulation. *J. Chem. Theory Comput.* **4**, 116–122 (2008).
86. Hoover, W. G. Canonical dynamics: equilibrium phase-space distributions. *Phys. Rev. A* **31**, 1695 (1985).
87. Parrinello, M. & Rahman, A. Polymorphic transitions in single crystals: a new molecular dynamics method. *J. Appl. Phys.* **52**, 7182–7190 (1981).
88. Huang, J. et al. CHARMM36m: an improved force field for folded and intrinsically disordered proteins. *Nat. Methods* **14**, 71–73 (2017).

89. Lindahl, E., Abraham M. J., Hess, B. & van der Spoel, D. GROMACS 2021.3 Source code. *Zenodo* https://doi.org/10.5281/zenodo.5053201 (2021).
90. Tomasello, G., Armenia, I. & Molla, G. The Protein Imager: a full-featured online molecular viewer interface with server-side HQ-rendering capabilities. *Bioinformatics* **36**, 2909–2911 (2020).
91. Lyu, S., Sowlati-Hashjin, S. & Garton, M. ProteinVAE: variational autoencoder for design of synthetic viral vector serotypes. *Code Ocean* https://doi.org/10.24433/CO.2530457.v2 (2023).

Acknowledgements

We thank Z. Wen for engaging in discussions and sharing ideas pertaining to the application of protein language models in the context of this research. This work was supported by grants from the Canadian Institute of Health Research (CIHR) and the Natural Sciences and Engineering Research Council of Canada. We also thank SciNet and the Digital Research Alliance of Canada for providing essential computing resources, without which this study could not be conducted.

Author contributions

M.G. and S.L. conceived the project. S.L. designed the generative model, performed all experiments and analysed the results. S.S.-H. conducted molecular dynamics simulations, analysed simulation results with S.L.'s assistance and contributed to the corresponding section. S.L. and M.G wrote, revised and edited the paper. M.G. supervised the project.

Competing interests

The University of Toronto is in the process of filing for a patent on this method. All authors declare that there are no competing interests aside from the patent pending.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-023-00787-2.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-023-00787-2.

Correspondence and requests for materials should be addressed to Michael Garton.

Peer review information *Nature Machine Intelligence* thanks Jinwoo Leem and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

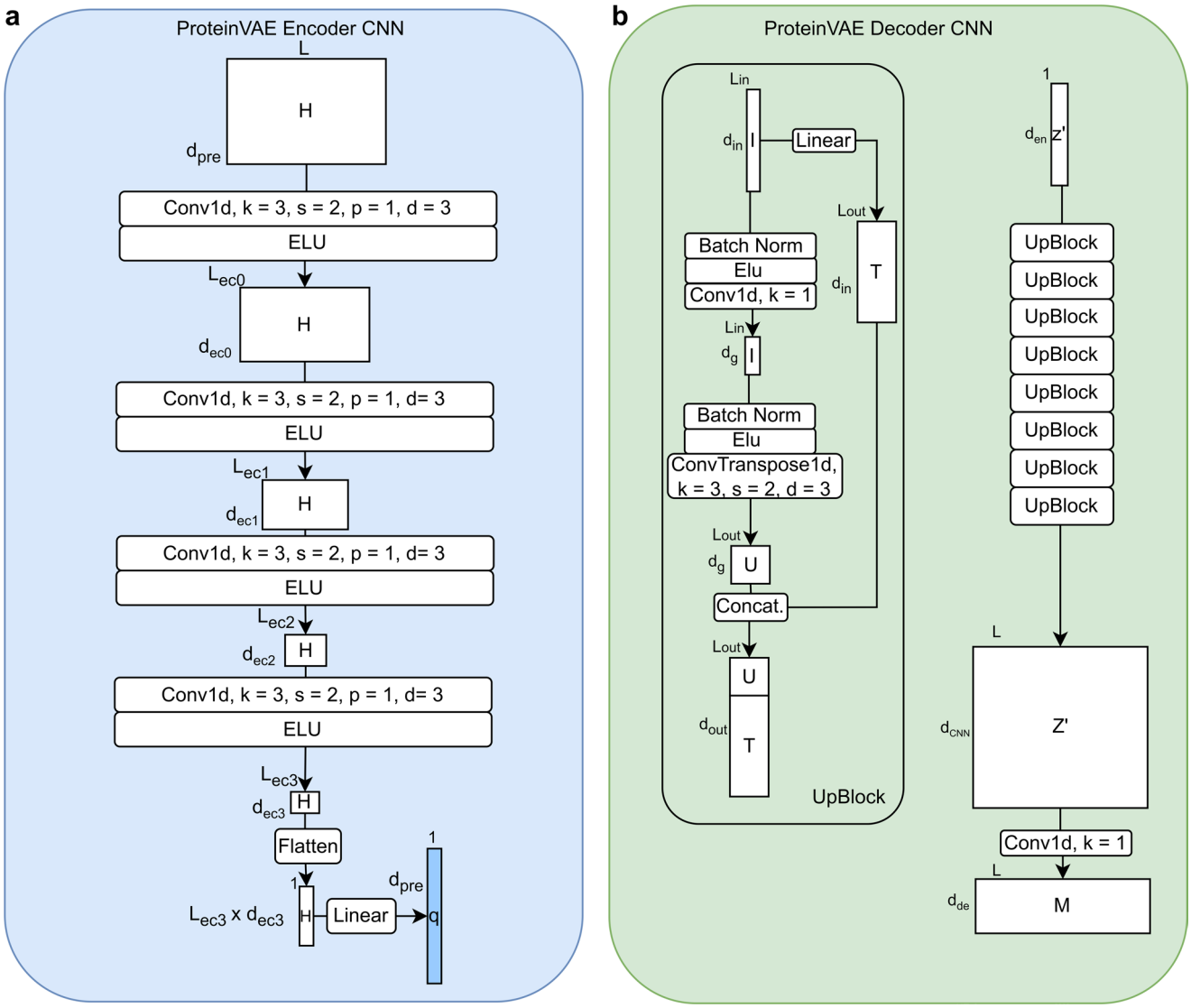
Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

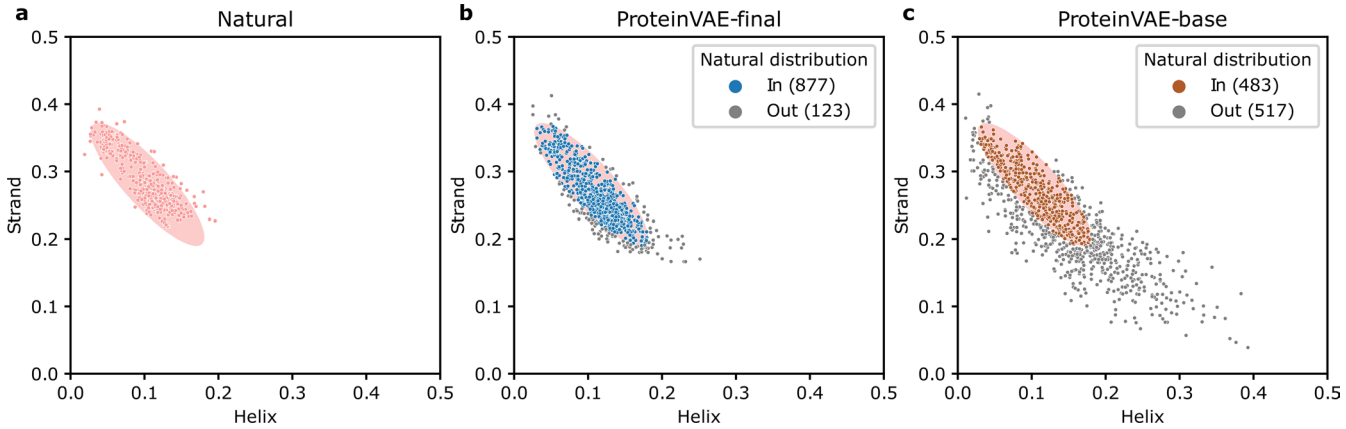
© The Author(s), under exclusive licence to Springer Nature Limited 2024

https://doi.org/10.1038/s42256-023-00787-2



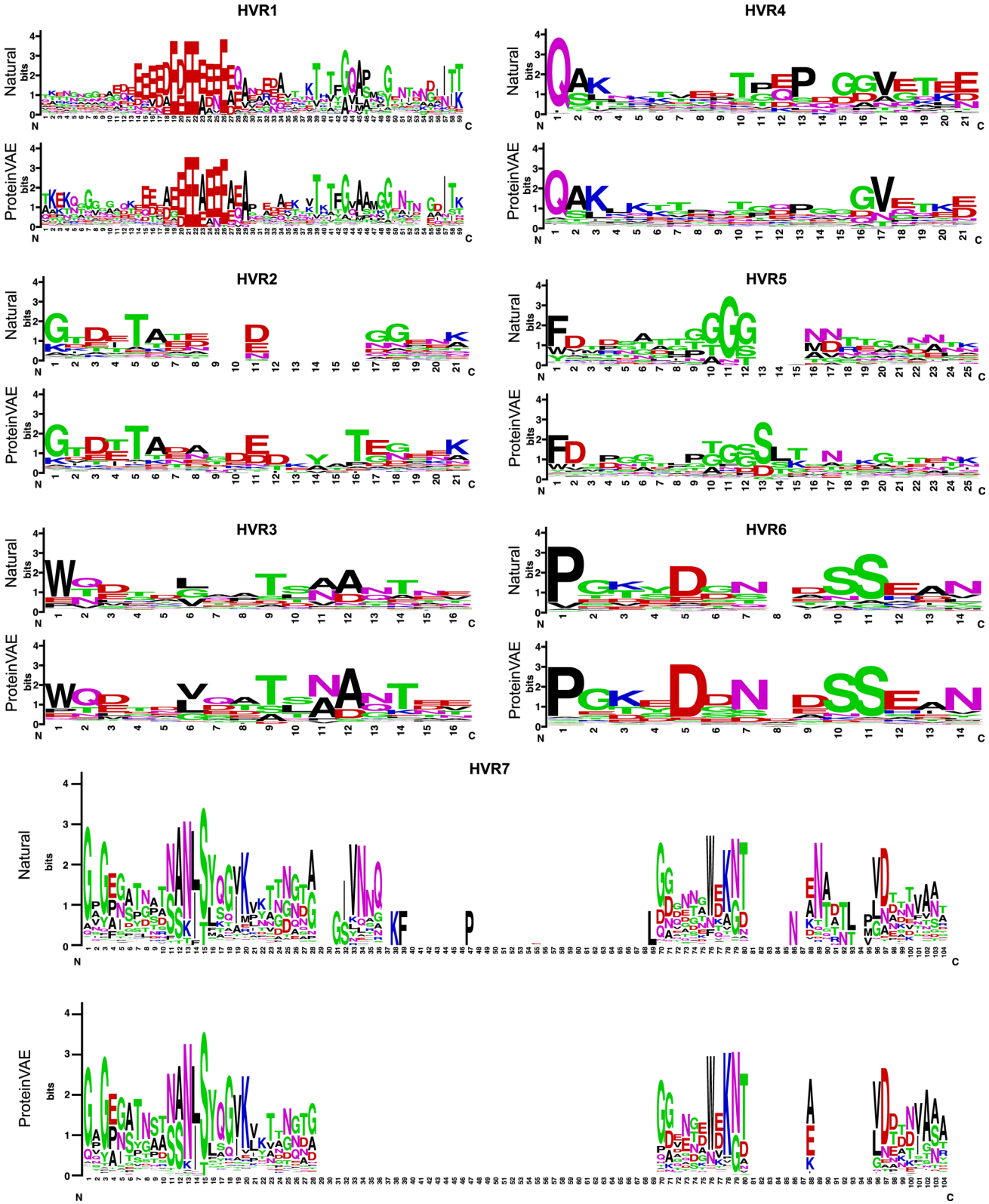
Extended Data Fig. 1 | Detailed Architecture of Encoder and Decoder CNN. (**a**) Encoder CNNs used a series of dilated 3 × 3 convolution layers along the sequence length dimension to reduce dimensionality of the pretrained language model amino-acid level embeddings. The flattened matrix is then transformed to the same length as the latent size of the pretrained language model embeddings to be used as the query in bottleneck attention. (**b**) Decoder CNNs used 8 UpBlocks to upsample the VAE latent vector length (equals 1) to maximal sequence length. In each UpBlock, a 1 × 1 convolutional layer is used to transform input to a lower dimension, which reduces the number of parameters needed in the following layer with large kernels. The dilated 3 × 3 deconvolutional layer with stride of 2 is used to upsample the low-dimensional input. To prevent gradient vanishing, the input is also passed through a linear layer to get an identity matrix (T) of the same length as the deconvolutional output (U). The upsampled matrix U and the identity matrix is then concatenated as the input for the following UpBlock. The output of the final UpBlock is transformed to the decoder hidden dimension with another 1 × 1 convolutional layer.

https://doi.org/10.1038/s42256-023-00787-2



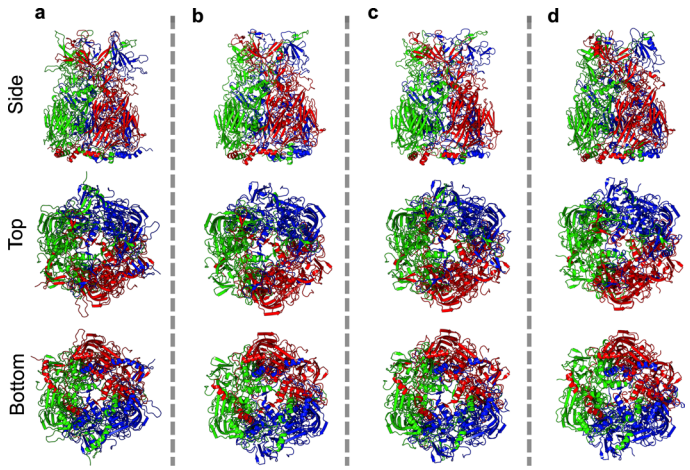
Extended Data Fig. 2 | Helix-to-strand Ratio in Sequences Generated by Base and Final version of ProteinVAE. (**a**) Natural hexon helix-strand ratio (n = 711). (**b**) helix-strand ratio from sequences generated using the final version of ProteinVAE model (n = 1000). (**c**) helix-strand ratio from sequences generated using the base version of ProteinVAE model (n = 1000). Secondary structure state was predicted from sequences directly using SPOT-1D.

https://doi.org/10.1038/s42256-023-00787-2



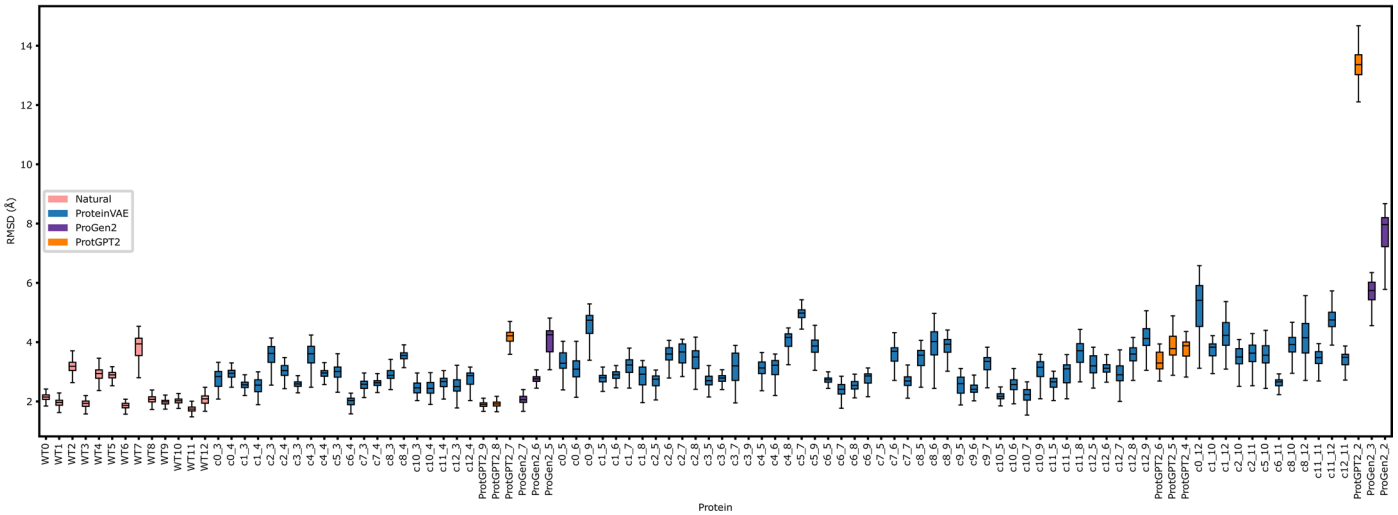
Extended Data Fig. 3 | Hypervariable regions in Natural and ProteinVAE-generated Sequences. Sequence logo of all 7 hypervariable regions for MSA of natural sequences and MSA of ProteinVAE generated sequences. As can be seen, both MSA have similar amino acid usage in majority of the columns.

https://doi.org/10.1038/s42256-023-00787-2



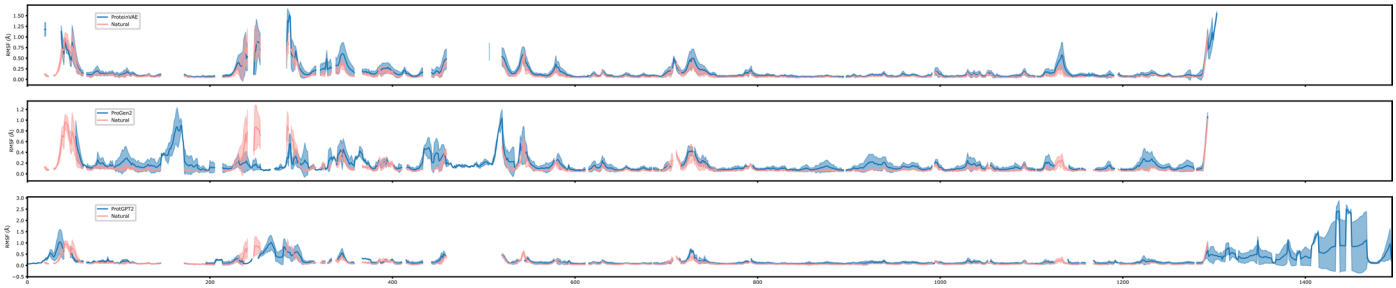
Extended Data Fig. 4 | Molecular Dynamics Representative Structures. Each column shows a hexon homotrimer from one hexon sequence. Side, top, and bottom views of all structures were shown in the first, second, and third row, respectively. Red, green, and blue colouring represent different subunits of the homotrimer. Column (**a**) is a wild-type structure. Columns (**b**–**d**) each display structure of a ProteinVAE generated sequence at 91.5%, 85.6%, 75.4% sequence identity with respect to their respective closest natural sequence.

https://doi.org/10.1038/s42256-023-00787-2



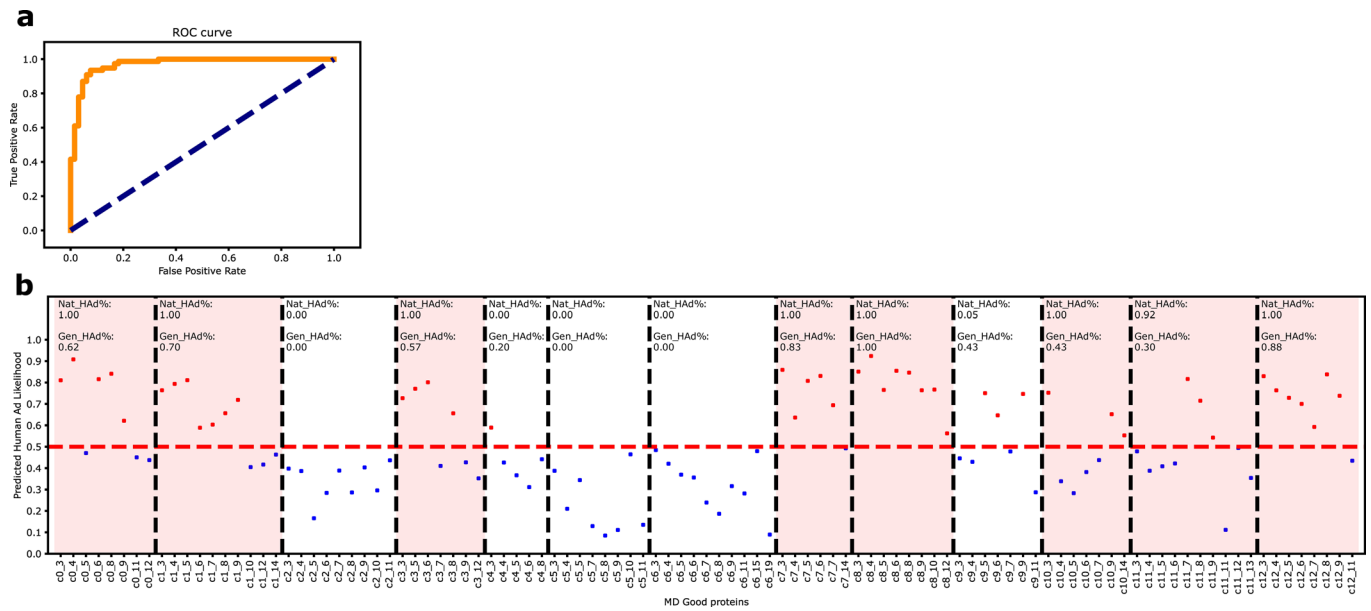
Extended Data Fig. 5 | RMSD for Simulated Sequences. RMSD for all natural representative sequences, ProteinVAE generated sequences, ProGen2 generated sequences (3 generated structures had structural clashes), and ProteinGPT2 generated sequences (3 generated structures had structural clashes). Each box-plot shows the first and third quartiles, central line is median, and whiskers show range of data with outliers are omitted for readability. For each sample, the RMSD value for every picosecond from 5 ns to 100 ns were analyzed (n = 950).

https://doi.org/10.1038/s42256-023-00787-2



Extended Data Fig. 6 | RMSF Aligned According to MSA with Gaps Preserved. Top: Average RMSF for ProteinVAE generated sequences (blue) and natural representative sequences (pink) Middle: Average RMSF for ProtGen2 generated sequences (blue) and natural representative sequences (pink). Bottom: Average RMSF for ProtGPT2 generated sequences (blue) and natural representative sequences (pink). ProtGen2 and ProtGPT2 generated sequences inserted long fragments that are not homologous to any natural hexon. These fragments also have increased flexibility which could reduce structure stability. Data in (a-c) are presented as mean values +/− SD.

https://doi.org/10.1038/s42256-023-00787-2



Extended Data Fig. 7 | Human AdV Classifier. (a) Receiver operating characteristic (ROC) curve of latent human adenovirus hexon classifier. Area under the ROC curve is 0.97. **(b)** Predicted human AdV hexon likelihood for all sequences generated from each cluster. Sequences predicted to be human AdV hexon were shown as a red dot, and predicted non-human AdV hexon were shown as a blue dot. Percentages of human AdV in corresponding natural sequences were labeled as Nat_HAd% in each cluster. Clusters with more than 90% natural human AdV hexons were colored with a pink background. Predicted percentages of human AdV for generated sequences were labeled as Gen_HAd%. Decision threshold is shown as a dashed red line.

nature portfolio

Corresponding author(s): Michael Garton

Last updated by author(s): Nov 27, 2023

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☐	☒	A description of all covariates tested
☐	☒	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☐	☒	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	No software was used
Data analysis	pingouin 0.5.2, alphafold 2.0.0, Clustal Omega 1.2.4, MMSeqs2 (release Feb 24, 2021), SPOT-1D (original release), FreeSASA 2.1.0, PhyML 3.0, torchmetrics 0.8.1, pytorch-lightning 1.6.5, pytorch 1.12.1, scikit-learn 1.2.21, wandb 0.15.01, CHARMM-GUI server v3.8, GROMACS/2021.3, Protein Imager 0.5.60, https://doi.org/10.24433/CO.2530457.v2

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

Sequences of all 711 natural hexon can be found at /data/hexon_711.fasta in the CodeOcean capsule (https://doi.org/10.24433/CO.2530457.v2). All natural hexon sequences were downloaded from the UniprotKB26,37 database. Source data are provided with this paper.

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender	N/A
Population characteristics	N/A
Recruitment	N/A
Ethics oversight	N/A

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	Sample size was chosen based on total publicly available data (711 adenovirus hexon sequences).
Data exclusions	Adenovirus hexon sequence below a certain length threshold were excluded on the basis that they were incomplete
Replication	Study reports a new machine learning model and therefore does not report experimental findings
Randomization	Sequences were randomly split into the training/validation/test set at a ratio of 7/2/1 to avoid bias in training and evaluation
Blinding	Blinding was not relevant as the study is computational and the outcomes are algorithm dependent, therefore not subject to human biases that would warrant blinding.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging