Computational and Structural Biotechnology Journal 21 (2023) 5839–5850

Contents lists available at ScienceDirect

Computational and Structural Biotechnology Journal

journal homepage: www.elsevier.com/locate/csbj

Research Article

Improving antibody optimization ability of generative adversarial network through large language model

Wenbin Zhao [a,1], Xiaowei Luo [a,1], Fan Tong [a], Xiangwen Zheng [a], Jing Li [b], Guangyu Zhao [b], Dongsheng Zhao [a,*]

[a] Academy of Military Medical Sciences, Beijing 100850, China
[b] Beijing Institute of Microbiology and Epidemiology, State Key Laboratory of Pathogen and Biosecurity, Beijing 100071, China

ARTICLE INFO

Keywords:
Antibody optimization
Generative Adversarial Network
Language model

ABSTRACT

Generative adversarial networks (GANs) have successfully generated functional protein sequences. However, traditional GANs often suffer from inherent randomness, resulting in a lower probability of obtaining desirable sequences. Due to the high cost of wet-lab experiments, the main goal of computer-aided antibody optimization is to identify high-quality candidate antibodies from a large range of possibilities, yet improving the ability of GANs to generate these desired antibodies is a challenge. In this study, we propose and evaluate a new GAN called the Language Model Guided Antibody Generative Adversarial Network (AbGAN-LMG). This GAN uses a language model as an input, harnessing such models' powerful representational capabilities to improve the GAN's generation of high-quality antibodies. We conducted a comprehensive evaluation of the antibody libraries and sequences generated by AbGAN-LMG for COVID-19 (SARS-CoV-2) and Middle East Respiratory Syndrome (MERS-CoV). Results indicate that AbGAN-LMG has learned the fundamental characteristics of antibodies and that it improved the diversity of the generated libraries. Additionally, when generating sequences using AZD-8895 as the target antibody for optimization, over 50% of the generated sequences exhibited better developability than AZD-8895 itself. Through molecular docking, we identified 70 antibodies that demonstrated higher affinity for the wild-type receptor-binding domain (RBD) of SARS-CoV-2 compared to AZD-8895. In conclusion, AbGAN-LMG demonstrates that language models used in conjunction with GANs can enable the generation of higher-quality libraries and candidate sequences, thereby improving the efficiency of antibody optimization. AbGAN-LMG is available at http://39.102.71.224:88/.

1. Introduction

Monoclonal humanized antibodies have proven successful in treating various diseases, including tumors and infections [1–3]. The COVID-19 pandemic garnered new clinical attention for these antibodies due to their specificity and effectiveness in neutralizing viruses [4,5]. Before being deployed as treatments, antibodies require optimization that enhances the affinity of a target antibody for the antigen or that improves a target antibody's broad-spectrum activity (usually focusing on affinity enhancement) [6]. Sequentially altering target antibodies is a common way to optimize them functionally and structurally. However, before such optimization can begin, it is critical to determine an initial sequence space of libraries of appropriate quantity and quality. This is difficult, as the diversity of antibody sequences entails a vast search space, which problem is only compounded by the high cost and low efficiency of wet-lab experiments. Hence, researchers often employ computer-aided methods to progressively narrow down the search space and ultimately select a few high-quality candidate antibodies for wet-lab validation [7–9].

Recent global health crises such as COVID-19 have underscored the need to develop antibody treatments efficiently, and while COVID-19 itself is no longer a public health emergency, its endemic presence in communities and the ongoing mutations of SARS-CoV-2 continue to pose significant implications for human health [10,11]. The need for novel and effective treatments for such present and future crises necessitates an innovative model for developing antibody treatments that maximize their specificity, affinity, and therapeutic utility [12]. Current approaches in the field are not yet adequate [13]. While computer-aided

* Corresponding author.
E-mail address: dszhao@bmi.ac.cn (D. Zhao).
[1] The authors contributed equally to this article.

https://doi.org/10.1016/j.csbj.2023.11.041
Received 7 October 2023; Received in revised form 21 November 2023; Accepted 21 November 2023
Available online 23 November 2023

methods for protein optimization exist, they are not optimally efficient in engineering antibodies. It is therefore vital to develop a customized and efficient antibody generation model that is specifically designed to expedite antibody development.

Computer-aided antibody generation typically relies on a text-generating language model trained on a large dataset, typically an autoregressive model [14]. However, autoregressive models suffer from degradation caused by error accumulation. Each generated element depends on previously generated elements, leading to degraded quality in longer sequences [15]. Moreover, with limited training data, these models may not capture crucial features of antibody sequences, resulting in suboptimal outcomes. In contrast, GANs [16], comprising a generator and a discriminator trained through mutual adversarial learning, generate and evaluate their own data based on a training set and can produce text sequences as cohesive wholes. Sequences generated through a GAN thus do not degrade according to length. Yet it remains crucial to generate sequences that possess the essential characteristics of the target antibodies. One potential way to achieve this involves using the encoded target sequences from pre-trained language models as part of the input to the GAN. This so-called 'deep learning-based sequence embedding' would help gather complex and extensive representations of antibodies. Such representations encompass information from diverse levels in protein or antibody sequences, including biophysical properties, evolutionary information, and protein structure information [17]. Integrating language models with a GAN could hence allow the resultant model to capture the features of the target sequences and generate similar sequences, and could thus improve training efficiency and the likelihood of useful generated outputs.

We attempted to optimize antibody generation by combining language models with a GAN, culminating in a model called AbGAN-LMG. This model attempted to generate high-quality antibody sequence libraries for screening. We validated the capability of the model by generating sequences for anti-SARS-CoV-2 antibodies and anti-MERS-CoV antibodies. Multiple metrics were used to evaluate the generated libraries and sequences. We explored the impact of the representation information derived from the language model on the GAN by comparing AbGAN-LMG with various baseline models. The baseline models comprised a GAN model without representation information, a GAN model with traditional sequence features added, and a retrained protein sequence generation model. In addition, we compared the performance of different language models when applied to the GAN. AbGAN-LMG can be accessed at http://39.102.71.224:88/.

2. Related work

Advanced computer models have recently been used to generate diverse antibody and protein sequences. This section critically assesses the range of currently used models, analyzing their applications and limitations for antibody sequence generation. Additionally, it explores large language models, such as BERT-based frameworks, as a promising avenue for comprehensive sequence representation, highlighting their application across various tasks and their specific implications for research in antibody generation.

2.1. Generative models for proteins or antibodies

Recent research has used computer models to generate candidate antibody libraries. Through self-supervised training on a large amount of sequence data, these models learn the underlying characteristics within the sequences, enabling them to generate new antibody sequence libraries that exhibit the same basic features as those on which they are trained. Autoregressive language models, like IgLM developed by Richard et al. [18], have been used to learn the features of sequences and generate new ones, operating akin to natural language text generation. Antibodies generated through these autoregressive models exhibit better developability than sequences obtained through random mutations. Other researchers have followed suit. Xu et al. introduced Ab-Gen [19], which uses reinforcement learning methods to generate antibodies with specified attribute constraints. Melnyk proposed ReprogBert [20], in which a pre-trained English language model is repurposed for protein sequence infilling. This model demonstrates high diversity in CDR sequencing without compromising structural integrity and naturalness, even with low-resourced antibody sequence datasets. However, while the autoregressive models demonstrate proficiency in generating sequences for specific regions, their limitations become apparent as the length of the sequence expands. Error accumulation within the models hampers their ability to replicate the natural conformation observed in real sequences, impacting the fidelity and authenticity of the generated antibody sequences.

Prior studies have applied GANs to tasks involving the generation of proteins or DNA sequences [21,22]. ProteinGAN captures evolutionary dependencies among amino acids, expanding the protein sequence space and generating fully functional protein sequences with physicochemical properties similar to natural proteins [23]. FeedbackGAN is a DNA sequence generation model [24]. It optimizes generated DNA sequences to obtain desired characteristics by using a highly effective functional-prediction feedback loop. ProteoGAN, operating as a conditional GAN, uses GO labels as conditional input information to generate protein sequences with desired functionalities [25]. Amimeur et al. introduced a GAN [26], which uses transfer learning to bias the GAN to generate antibodies with key properties of interest. But they only optimized for one property of the antibody, rather than optimizing for multiple properties at once. Due to the distinctive evolutionary characteristics of antibodies as compared to conventional proteins [27], the application of existing protein generation models for the generation of antibody sequences is deemed inadequate. The use of these models has revealed a fundamental inadequacy in producing antibody libraries of requisite quality. The shortcomings observed with extant protein generation models underscore the necessity for specialized methodologies that account for the idiosyncrasies of antibody sequences. Such methodologies would establish robust platforms capable of producing high-quality antibody libraries essential for advanced therapeutic and diagnostic applications.

2.2. Large language model for sequences representation

Sequence representation methods fall into two categories: traditional feature extraction, and deep learning-based embedding. Traditional sequence feature extraction methods extract features such as amino acid types, proportions of different amino acids, and proportions of amino acids with different physicochemical properties [28–30]. However, these methods offer limited information in capturing specific aspects of the sequences, and they often yield discrete, sparse features, impacting computational efficiency. Self-attention language models have shown powerful capabilities in handling both natural language and biological sequences such as those in nucleic acid and proteins. Many studies have demonstrated the advantages of using language models to represent biological sequences. For instance, Li proposed an antibody design framework that combines language models, Bayesian optimization, and high-throughput experimentation [31]. The pre-trained BERT-based model optimized antibody affinity through fine-tuning using high-throughput experimental data. Hie used the ESM2 language model for affinity maturation of seven wild-type antibodies, resulting in stable designs effective against the Ebola virus or SARS-CoV-2 [32].

Numerous protein language models (PLMs) or antibody language models (ALMs) are now available to researchers. ESM2 has been pre-trained using the ESM architecture and a large amount of protein sequence data [33]. ProtBERT [34] has been trained using the BERT [35] architecture and the UniRef100 and BDF100 datasets. Other alternatives are AntiBERTy [36], which was pretrained using the BERT architecture and the OAS database [37], and which is utilized as a tool for sequence representation by the antibody modeling tool IgFold [38].

BERT2DAb (available at https://github.com/Xiaoxiao0606/BERT2DAb), developed by our research group, uses the BERT architecture and the OAS database for pretraining and incorporates secondary structure information for representation learning. It has performed promisingly in tasks such as antigen-antibody binding specificity classification. Alternatively, AbLang [39] uses the Roberta architecture and the OAS database for pretraining and performs well in tasks involving the recovery of missing residue information in antibody sequences due to sequencing errors.

3. Materials and methods

3.1. Workflow

The research workflow consisted of three steps: Training the Model, Generating the Antibody Library, and Evaluating the Generated Library and Sequences (Fig. 1).

3.2. Data

3.2.1. Training data

The training data was sourced from the CoV-AbDab database [40], which aggregates information on 12,021 anti-coronavirus antibodies, including both light chain and heavy chain variable region sequences. To ensure data uniformity, sequences in the database longer than 128 amino acids or containing non-standard amino acids were excluded, resulting in a refined pool of 11,205 antibody sequences. Seeking greater diversity in the resulting dataset, we used MMseq2 to cluster the antibody sequences at a 70% similarity threshold [41]. In clusters with fewer than 3 sequences, 1 sequence was randomly selected, while in other clusters, 5% of the sequences were randomly allocated to the test set of 642 total sequences. The remaining 10, 563 sequences formed the training set.

3.2.2. Representing sequences with language models

Language models established distributed representations for each antibody sequence in the dataset. These representations served as feature vectors inputted into the GAN model. ESM2–150 M, ProtBERT, which are PLMs, and BERT2DAb, AntiBERTy, and AbLang, which are ALMs, were used to characterize antibody sequences. Detailed model parameters can be found in the Supplementary File A.1. The names of GANs are different for different language models. We use the above five language models, so there are five models trained by us: AbGAN-ESM2–150 M, AbGAN-ProtBERT, AbGAN- BERT2DAb, AbGAN-AntiBERTy and AbGAN-AbLang.

3.3. Model architecture and training

3.3.1. Model architecture

The AbGAN-LMG architecture follows the conventional GAN structure [42], comprising a generator and a discriminator module, each integrating five residual blocks (Fig. 2). In the generator module, an input is formed by combining a noise vector, drawn from a normal distribution, and the antibody sequence representation vector, derived from the language model. The input passes through successive residual blocks and a self-attention layer to generate optimized sequences. Meanwhile, the discriminator module encodes the sequences generated by the generator or the wild-type antibody sequences into one-hot encoding for its inputs. As the input traverses the 1st, 3rd, and 5th residual blocks, the resulting hidden vector merges with the antibody sequence representation vector. The cumulative outcome of these fusion processes contributes to the discriminator score. Detailed parameters of the generator and discriminator modules can be found in Supplementary File A.2.

3.3.2. Model training

AbGAN-LMG was implemented and trained using the PyTorch framework [43]. The training process consists of two parts: training the generator module and training the discriminator module (Fig. 3). The training steps for both modules are set at a 1:1 ratio to maintain a balanced training process. The Adam algorithm optimizes both the generator and discriminator modules, with an initial learning rate of 1E-4. The learning rate decays by a factor of 0.98 after every 1000 epochs to enhance convergence. The Gumbel-Softmax strategy was used to address the issue of the non-differentiability of the Softmax function during the generator module's training [44]. Additionally, either the top-k or top-p method was used to sample the output sequences. AbGAN-LMG was trained for 12000 epochs with a batch size of 64. The training took 36 h on an NVIDIA Tesla A800 GPU.

The objective of training the generator module is to generate antibody sequences that closely resemble wild-type antibody sequences. This similarity poses a challenge to the discriminator module, which aims to distinguish between the generated sequences and the wild-type antibody sequences. The objective of training the discriminator module

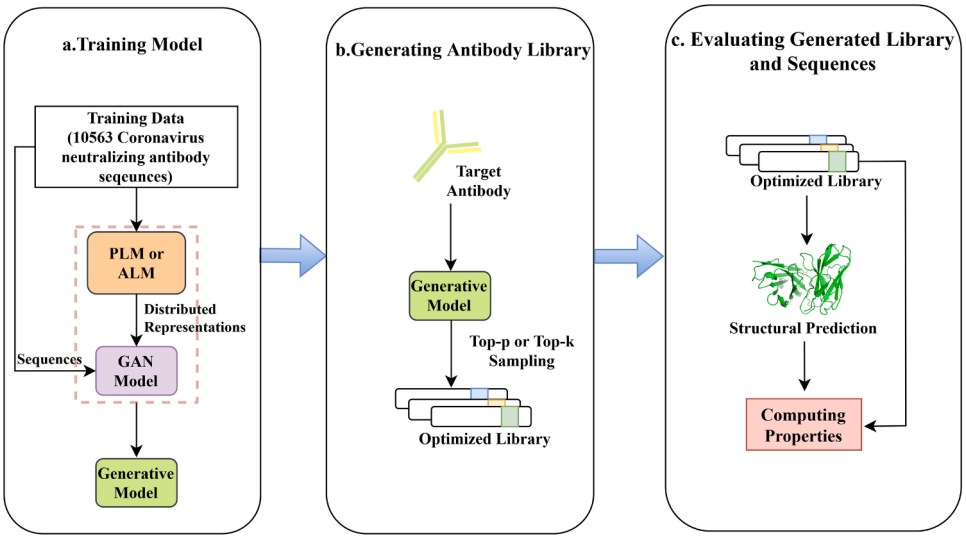


Fig. 1. Workflow. a. Processed training data is inputted into a PLM or ALM to extract distributed representations. These distributed representations and sequence information then train the generative model. b. The generative model created from step a. produces sequences of a target antibody that are sampled using either the Top-p or Top-k sampling method. c. The physicochemical properties, developability, and affinity of the optimized antibodies are evaluated.

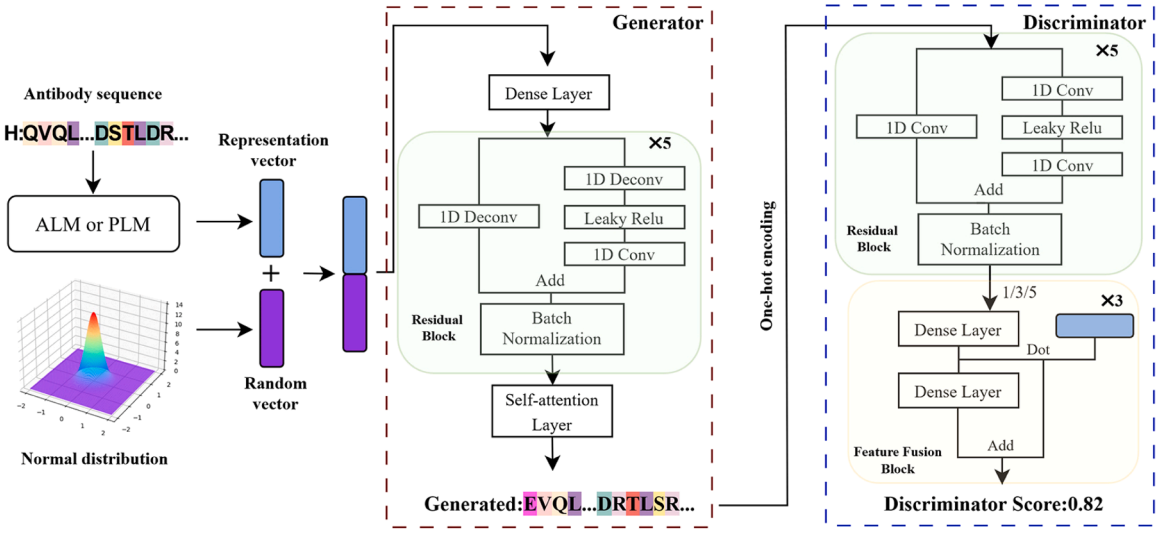


Fig. 2. Architecture of AbGAN-LMG. The generator module takes two inputs: a random vector drawn from a distribution and a representation vector extracted from the language model. The input to the discriminator is either the generated sequence or the wild-type antibody sequence. The higher the discriminator score, the greater the likelihood that the sequence is a wild-type sequence.

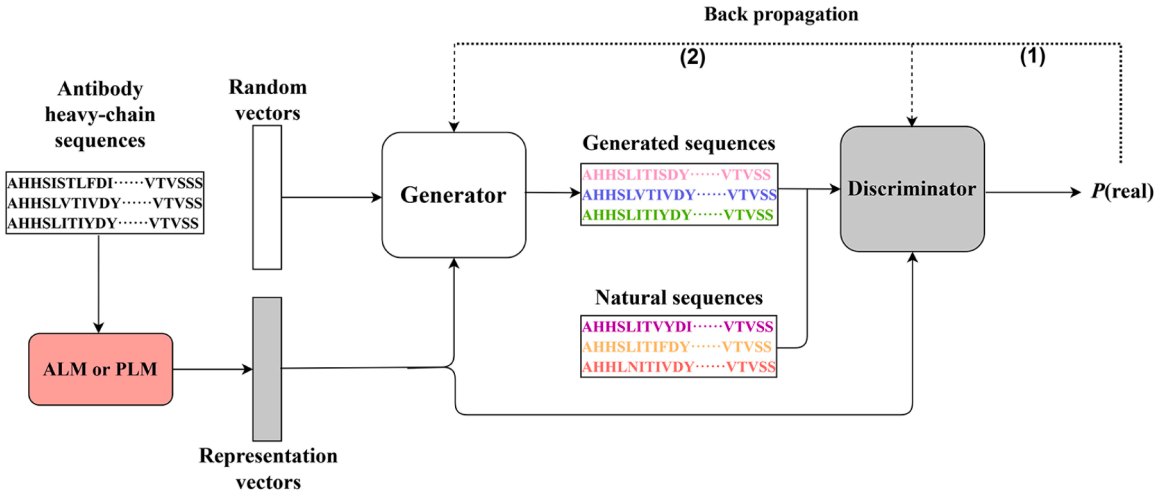


Fig. 3. Model training processes. During forward propagation, the generator module generates antibody sequences based on random noise vectors and representation vectors obtained from the language model. The discriminator then assigns scores to both the generated antibody sequences and the wild-type antibody sequences (solid lines). During backward propagation, the loss function is backpropagated to both the discriminator module (dashed lines (1)) and generator module (dashed lines (2)) for learning and optimization.

is to enhance its ability to accurately distinguish between the generated sequences and the wild-type antibody sequences. The objective function of the model training is represented as Eq. (1).

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x|y)] + E_{z\sim p_z(z)}[\log(1-D(G(z|y)))] \quad (1)$$

Here, V represents the objective function; G represents the generator module; D represents the discriminator module; E represents the expectation operator; y represents the representation vector of the wild-type antibody sequence encoded by the language model.; $x \sim p_{data}(x)$ indicates that x is sampled from the wild-type antibody sequence sample space; and $z \sim p_z(z)$ indicates that z is sampled from the sample space of noise data. $G(z|y)$ represents the sequence generated by the generator module when the representation vector encoded by the language model is used as input. $D(x|y)$ represents the probability that the discriminator module judges the wild-type antibody sequence as real when the representation vector encoded by the language model is used as input. $D(G(z|y))$ represents the probability that the generated antibody sequence by the generator module, when the representation vector encoded by the language model is used as input, is judged as real by the discriminator network.

The objective of training the generator network is to generate antibody sequences using the generator module and then evaluate these generated sequences through the discriminator module. Any discrepancy between the generated sequences and the wild-type antibody sequences is subsequently backpropagated to the generator module to aid learning and optimization. The objective function for the generator module's training is represented as Eq. (2).

$$\min_G V(D,G) = E_{z\sim p_z(z)}[\log(1-D(G(z|y)))] \quad (2)$$

In Eq. (2), the objective is to minimize the value of $V(D,G)$.When the value of $D(G(z|y))$approaches 1, the value of $V(D,G)$ is minimized. At this point, the discriminator module is more likely to judge the generated antibody sequences from the generator module as real, indicating a high similarity between the generated antibody sequences and the wild-

type ones.

The primary objective of training the discriminator module is to have it distinguish between the wild-type antibody sequences and the antibody sequences generated by the generator module. The discrepancy information between the wild-type and generated antibody sequences is then backpropagated to the discriminator module to facilitate learning and optimization. The objective function for the discriminator module's training is represented as Eq. (3).

$$\max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x|y)] + E_{z\sim p_z(z)}[\log(1-D(G(z|y)))] \quad (3)$$

In Eq. (3), the objective is to maximize the value of $V(D,G)$. When $D(x|y)$ approaches 1 and $D(G(z|y))$ approaches 0, $V(D,G)$ is at its peak. At this point, the discriminator module is more likely to judge the wild-type antibody sequences as real with a probability close to 1. This indicates the discriminator is accurately identifying the wild-type sequences as real. Similarly, when the discriminator evaluates the generated antibody sequences from the generator module as fake with a probability close to 0, it effectively recognizes the generated sequences as generated.

3.4. Antibody generation and evaluation

3.4.1. Generation method

Two different methods were used for generating libraries using AbGAN-LMG. In the first method, antibody sequences from the CoV-AbDab served as input for AbGAN-LMG to generate a library of 12,021 sequences. Distributed representations of each antibody sequence in CoV-AbDab extracted by the language model were used as input to the generator, where each antibody in CoV-AbDab corresponded to a unique distributed representation and thus generated a new antibody, resulting in a library size of 12,021. In the second method, AZD-8895 [45] was used as the input for AbGAN-LMG to generate a library of 2000 sequences. AZD-8895 is a monoclonal antibody targeting the receptor binding domain of the SARS-CoV-2 spike protein, and it is thus used to work against the virus's entry into human cells. This antibody was chosen to assess the effectiveness of AbGAN-LMG in optimizing antibody treatments. In this specific scenario, only the sequence of AZD-8895 was embedded as a distributed representation using a language model. This distributed representation was then replicated to match the desired library size, and these duplicated representations were used as inputs to the generator. Although each input in the distributed representation represents AZD-8895, each generates a unique antibody. The library size generated in this case is predetermined.

3.4.2. Evaluation methods for generated antibody sequence library

3.4.2.1. Generative complexity. Generative complexity measures the diversity or variability at each residue position across all sequences in the generated library. It is derived from the final layer output of the generating module, which results in a vector of dimensions [20,128]. Here, 128 corresponds to the positions of amino acid residues from 1 to 128 in the sequence, and 20 represents the probability of 20 different amino acids occurring at each position. A higher value indicates greater variability at that residue position. The calculation employs Eq. (4):

$$GC = -\sum_{i=1}^{20} p(x_i)\log_{20} p(x_i) \quad (4)$$

Here, $p(x_i)$ represents the probability of amino acid x_i occurring at the current position in the sequence.

3.4.2.2. Percentage of different types of amino acids and different types of secondary structures in the antibody sequence. Each antibody sequence undergoes calculation to determine the proportions of polar, nonpolar, positively charged, and negatively charged amino acids. Additionally, the secondary structure prediction tool ProteinUnet [46] is used to predict the secondary structure of the antibody sequences, classifying them into three categories: alpha-helix, beta-sheet, and random coil. The proportions of these secondary structures are calculated for each sequence.

3.4.2.3. Generation of t-distributed stochastic neighbor embedding(t-SNE) plot. The scikit-learn t-SNE module was used for dimensionality reduction to visualize the generated antibody sequences in a lower-dimensional space [47]. Default settings were used, including an early exaggeration of 12, a learning rate of 200, and a maximum of 1000 iterations. Clustering of generated antibody sequences was performed with MMseq2 with a sequence identity threshold of 70%. Clusters were then categorized based on their sizes, spanning 1, 10, 100, and 1000 sequences per cluster. Representative sequences from each cluster were subjected to Clustal Omega [48] for computing the distance matrix, then they subsequently served as input for t-SNE dimensionality reduction. The resulting t-SNE coordinates were used to generate a visualization plot, where the size of the data points corresponds to cluster sizes based on the number of sequences they contain.

3.4.2.4. Evaluation of the distribution of the generated antibody sequence library. The evaluation of the distribution of the sequence library includes an assessment of the overall distribution similarity, conditional consistency, and diversity, as first proposed by Kucera et al.[25]. The Z_m score function, proposed by Santoni et al. [49], was assessed to evaluate the distribution of amino acid pairs in the generated antibody sequences (Supplementary File A.3).

3.4.2.5. Multiple sequence alignment. Multiple sequence alignment is performed by merging the wild-type antibody sequences with the generated sequences, with Clustal Omega used for the alignment. The Shannon entropy is then computed separately for each aligned position in both the wild-type and generated sequences to measure the variation at each position.

3.4.3. Evaluation methods for generated individual antibody sequence

3.4.3.1. Methodology. We first evaluated the developability of the generated antibody sequences in the library. Next, we randomly selected 100 sequences from the generated library and used IgFold to model their 3D structures, assessing the stability of these structures. Finally, we performed molecular docking of the selected 100 antibodies with the wild-type SARS-CoV-2 RBD (PDB: 8D8R(C)), and we calculated the affinity between the generated antibodies and the antigen. These evaluations provide valuable insights into the quality, stability, and binding capabilities of the generated antibody library, and help assess the performance of AbGAN-LMG in generating antibodies for the specific target antibody AZD-8895.

3.4.3.2. Antibody developability evaluation. We used computational biology tools to calculate important properties of the generated antibodies, including the isoelectric point, hydrophobicity, and specific developability metrics such as Aggregation Propensity [50], CamSol Score [51], Heavy OASis Percentile [52], and Average NetMHCIIpan Percentage [53]. These metrics indicate various characteristics of the generated antibodies, such as their propensity for aggregation, their solubility, their humanization potential, and their immunogenicity. The evaluation aims to assess the overall quality and suitability of the generated antibodies for therapeutic development. Supplementary File A.4 contains detailed information on the evaluation tools and methods.

3.4.3.3. Antibody tertiary structure stability evaluation. The stability of the antibody's 3D structure was evaluated using two metrics: a Per-residue Local Distance Difference Test (Per-residue LDDT) [54] and a Discrete Optimized Protein Energy (DOPE) score [55]. The Per-residue

LDDT assesses the accuracy of structure predictions by comparing the predicted protein structure with a known high-resolution structure (PDB ID: 8D8R(HL)). It measures the local distance difference between the predicted and known structures, where higher LDDT scores indicate greater similarity between the predicted and known local structures. Conversely, the DOPE score is based on the physical energy function of a protein's molecular mechanics force field. It evaluates the stability and folding quality of protein structures, with lower DOPE scores indicating more stable and reasonable protein structures.

3.4.3.4. Antigen-antibody docking and affinity prediction. Initially, the antigen-antibody interface was computed via the complex structure of AZD-8895 with the wild-type SARS-CoV-2 (PDB ID: 8D8R). Subsequently, molecular docking of the antigen-antibody complex was performed using Lightdock [56], with the contact residues between the antigen and the target antibody AZD-8895 and the CDR3 of the generated antibodies as the constraining residues. The docking results were then evaluated using the dfire scoring function, with the top-scoring docking result being selected. Finally, Prodigy was employed to predict the affinity of the resulting 3D structures [57].

3.5. Baselines

AbGAN-No-Guided, AbGAN-FEGS, and ProteinGAN were selected as baseline models to compare the impact of feature input and different types of features on model performance. The specific information for each model is shown in Table 1:

4. Results

4.1. AbGAN-LMG learns fundamental features of antibodies and generates libraries with high diversity

In this analysis, the AbGAN-LMG-generated antibody sequence library is examined from three perspectives: the variations of amino acids at each residue position, the proportions of different amino acid types and secondary structure types, and the library's diversity.

The generative complexity of each residue position in the generated library was calculated. the generative complexity in the three CDRs was higher than in the framework region, with CDR3 showing the highest generative complexity of all (Fig. 4[a] and Table 2). Additionally, the 3D structures of the antibody sequences generated by AbGAN-LMG exhibited greater changes in the CDR1, CDR2, and CDR3 regions than in the framework region (Fig. 4[b]). These results indicate AbGAN-LMG's capacity to discern hidden structural variations across distinct domains within the sequence, with higher variability in the CDRs than in the framework region and CDR3 showing the highest variability.

We next analyzed the proportions of different amino acid types and secondary structures in the generated library. The proportions of different amino acid types (polar and non-polar, positively charged, and negatively charged) and secondary structures (α-helix, β-strand, and irregular coil) in the generated sequences were similar to those in the wild-type antibody sequences of the training set (Fig. 4[c, d]). This similarity indicates that AbGAN-LMG can learn both the features and structural characteristics of the wild-type antibody sequences. However, the figures also reveal that compared to the wild-type antibody sequences, the generated sequences exhibit a long-tail phenomenon in the proportions of different amino acids and secondary structures. This suggests that the model may generate a few undesirably extreme sequences with higher or lower proportions of certain amino acids or secondary structures.

Finally, t-SNE was employed to visualize the sequences in the generated library and thus analyze the library's diversity. AbGAN-ESM2–150 M's generated antibody sequences can partially overlap with the large clusters of wild-type antibody sequences, indicating that the model has learned the features of the wild-type antibodies (Fig. 4[e]). Moreover, there are numerous smaller clusters surrounding the large cluster, suggesting that the generated library has higher diversity than the training and test sets. Additionally, we analyzed the diversity of the sequences in the generated library at different levels, namely in sequence and structure. At the sequence level, the generated antibody sequences demonstrated clustering at different sequence identity levels (Fig. 4 [f]). AbGAN-LMG generated a more extensive array of sequence clusters across varying identity levels compared to the wild-type sequences, indicating greater diversity in the generated antibody sequences. At the structure level, when ten sequences randomly selected from the generated library were aligned for their 3D structures, variations were observed in the CDR H3 region, which increased structural diversity, while other regions remained relatively conserved (Fig. 4 [g]).

Table 1
Baseline models.

Model	Network Framework	Feature Type	Strategy to Generate Antibody Library
AbGAN-FEGS	AbGAN-LMG	Utilizes FEGS to extract antibody sequence features; model input consists of sequence feature vectors and random noise vectors	Utilizes the same library generation strategy as AbGAN-LMG
AbGAN-No-Guided	In the AbGAN-LMG framework, the sequence embedding is removed.	Removes sequence embedding from AbGAN-LMG framework; input consists of only random noise vectors	AbGAN-No-Guided and ProteinGAN do not use representation; they first generate antibody libraries from random noise and then use the MMseq2 tool to align the generated library sequences with AZD-8895, incorporating sequences into AZD-8895 antibody library if sequence identity is greater than 70%
ProteinGAN	ProteinGAN		

4.2. AbGAN-LMG generates high-quality antibody sequence libraries

Firstly, the quality of the antibody library generated by AbGAN-LMG is evaluated based on its distribution characteristics. The evaluation included overall distribution similarity (MMD), conditional consistency (MRR and MRR_{LE}), and diversity (ΔEntropy and ΔDistance). AbGAN-LMG outperformed the baselines in all five metrics. AbGAN-BERT2DAb, which utilized sequence feature vectors generated by BERT2DAb, demonstrated the best performance in overall distribution similarity. AbGAN-AntiBERTy, which used sequence feature vectors from AntiBERTy, showed the best performance in conditional consistency. AbGAN-ESM2–150 M, employing sequence feature vectors from ESM2–150 M, exhibited the best diversity (Table 3).

Next, we assessed AbGAN-LMG's ability to capture long-range amino acid interactions. We computed the amino acid pair correlation matrices for the generated antibody sequences and the wild-type antibody sequences and then calculated the Pearson's correlation (Fig. 5[a]). The correlation between the amino acid pair correlation matrices of the sequences generated by AbGAN-LMG and the wild-type antibody sequences was superior to those of the baselines. Furthermore, to evaluate AbGAN-LMG's capability to capture the global distribution of antibody sequences, we performed multiple sequence alignments for the generated and wild-type antibody sequences and calculated the Shannon entropy for each position (Fig. 5[b]). We found that the correlation between the sequences generated by AbGAN-LMG and the wild-type antibody sequences was again better than the baselines. These results indicate that AbGAN-LMG captures long-range amino acid interactions

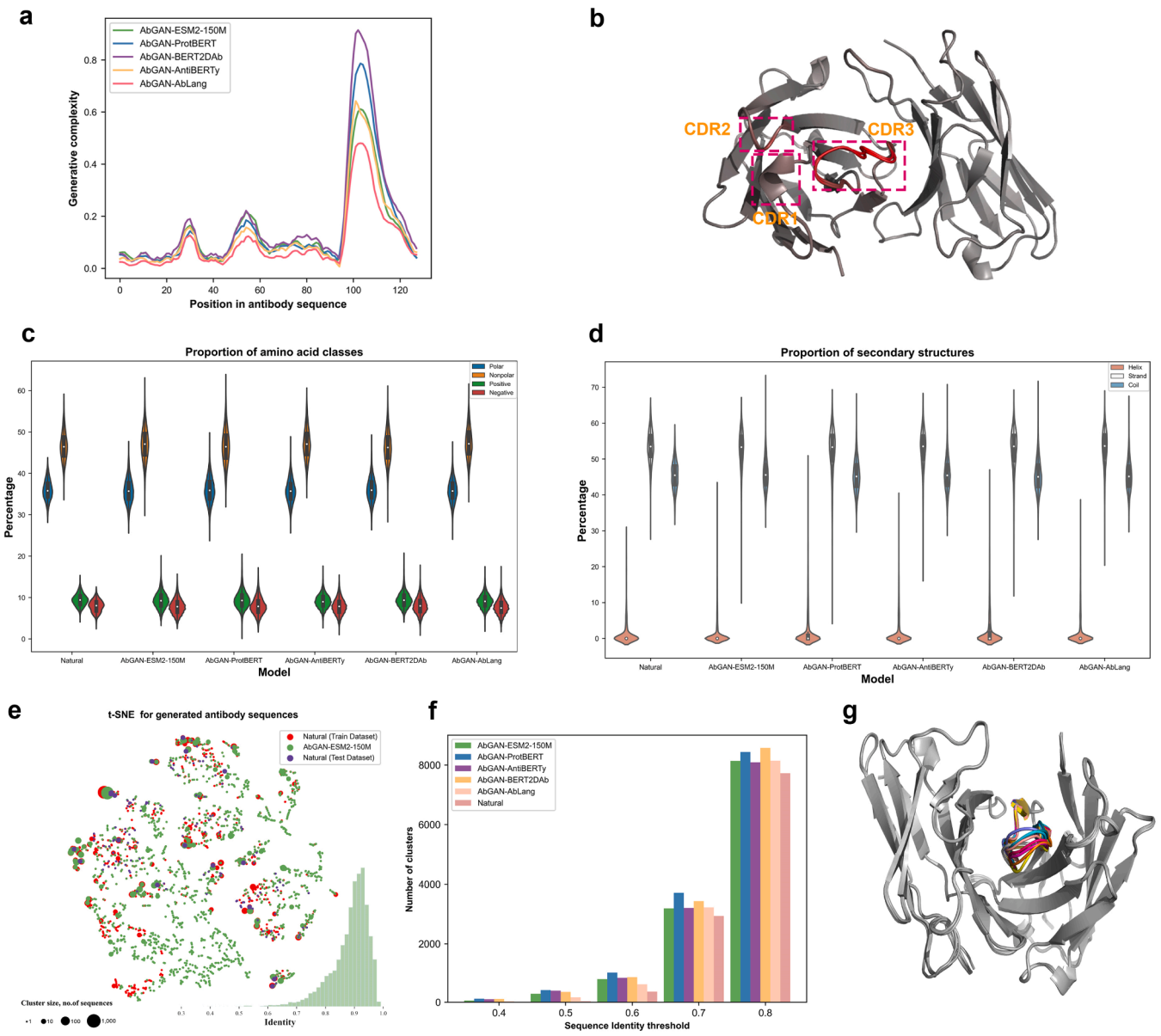


Fig. 4. AbGAN-LMG learns essential features of antibodies and generates libraries with high diversity. a. The plot demonstrates the diversity of amino acid residues at each position in the generative model. The regions exhibit higher complexity, indicating greater diversity in the generated antibody sequences. b. The plot compares the 3D structures of the generated antibodies to the wild-type antibody. Pinker areas represent larger Root Mean Square Deviation (RMSD) values, indicating significant changes in these areas of the generated antibodies. **c.** Distribution of amino acid types in the generated library and wild-type antibody sequences. d. Distribution of 2D structural in the generated library and wild-type antibody sequences. e. The visualization of the generated sequences using t-SNE. The results for the other four models can be found in the Supplementary File A.5. f. The generated antibody sequences and wild-type antibody sequences were clustered in different sequence identities, and the number of clusters was counted. g. The structural predictions for 10 optimized AZD-8895 sequences are aligned to the wild-type sequence. The colored region is the CDR H3 region of the antibody sequence. The pink represents the wild-type sequence.

Table 2
Generative complexity of generated antibody sequences in CDRs and framework region (FR).

Model	CDR1	CDR2	CDR3	FR
AbGAN-ESM2–150 M	0.123	0.187	0.438	0.076
AbGAN-ProtBERT	0.103	0.160	0.559	0.072
AbGAN-BERT2DAb	0.139	0.191	0.639	0.087
AbGAN-AntiBERTy	0118	0.136	0.445	0.066
AbGAN-AbLang	0.093	0.101	0.343	0.048

and the global distribution of antibody sequences more effectively than alternative current models.

4.3. *AbGAN-LMG generates better candidate antibodies*

The primary goal of antibody optimization is to obtain new antibodies based on a target antibody that have a higher affinity for the target antigen. Desirably, the generated antibodies should also be easy to develop. In this study, the antibody library generated with AbGAN-LMG was used to optimize the target antibody AZD-8895 and then compared with the same optimization performed by baseline models. Then, affinity and developability prediction screenings were used to further assess AbGAN-LMG's optimization outcomes.

Table 3
Evaluation of the distribution of generated antibody sequence libraries.

Model	MMD↓	MRR↑	MRR_{LE} ↑	ΔEntropy	ΔDistance
AbGAN-ESM2–150 M	0.0757 ± 0.0003	0.3795 ± 0.0026	0.5397 ± 0.0034	**0.0003 ± 0.0001**	**-0.0028 ± 0.0006**
AbGAN-ProtBERT	0.0853 ± 0.0004	0.2863 ± 0.0040	0.3998 ± 0.0040	-0.0016 ± 0.0001	-0.0386 ± 0.0008
AbGAN-BERT2DAb	**0.0725 ± 0.0005**	0.3085 ± 0.0027	0.3787 ± 0.0027	-0.0006 ± 0.0001	-0.0330 ± 0.0005
AbGAN-AntiBERTy	0.0819 ± 0.0006	**0.5501 ± 0.0028**	**0.6765 ± 0.0028**	0.0006 ± 0.0001	-0.0034 ± 0.0006
AbGAN-AbLang	0.0817 ± 0.0007	0.5238 ± 0.0024	0.6675 ± 0.0041	0.0004 ± 0.0001	0.0063 ± 0.0005
AbGAN-No-Guided	0.1230 ± 0.0027	—	—	0.0050 ± 0.0004	0.0920 ± 0.0043
AbGAN-FEGS	0.1123 ± 0.0007	0.2863 ± 0.0026	0.3415 ± 0.0034	-0.0044 ± 0.0002	-0.0581 ± 0.0008
ProteinGAN	0.1376 ± 0.0077	—	—	0.0109 ± 0.0006	0.1570 ± 0.0135

Note: An arrow indicates that lower (↓) or higher (↑) is better. The horizontal line (—) denotes that the model cannot calculate this metric due to the absence of representation information.

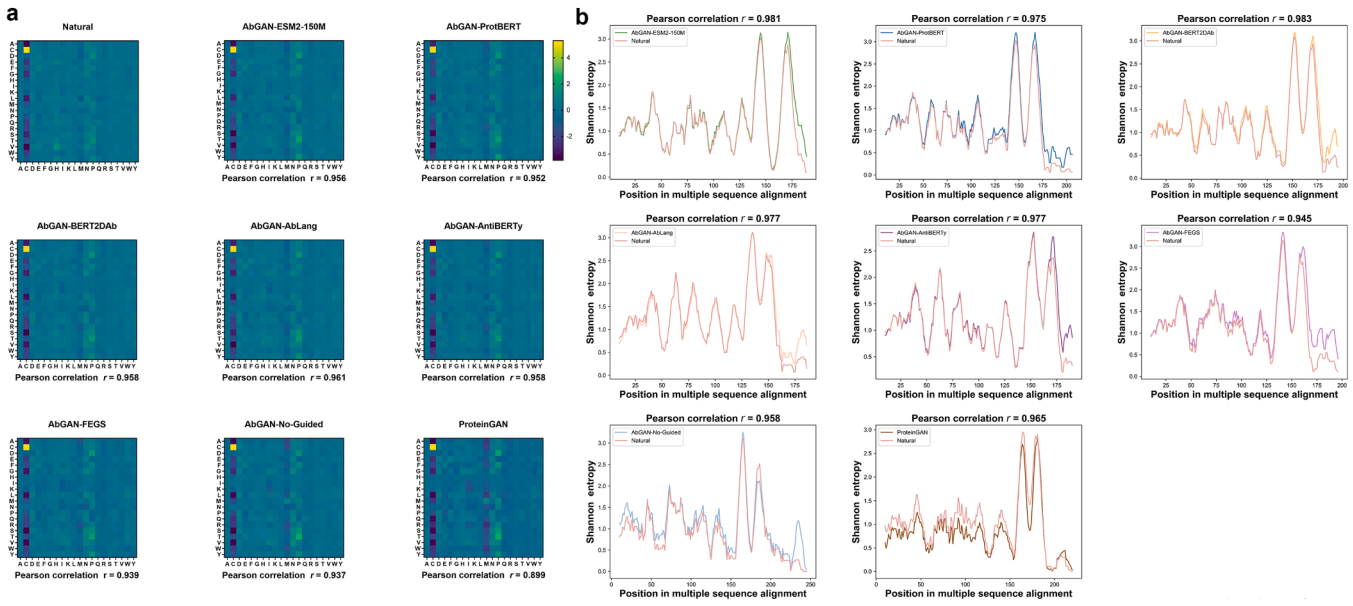


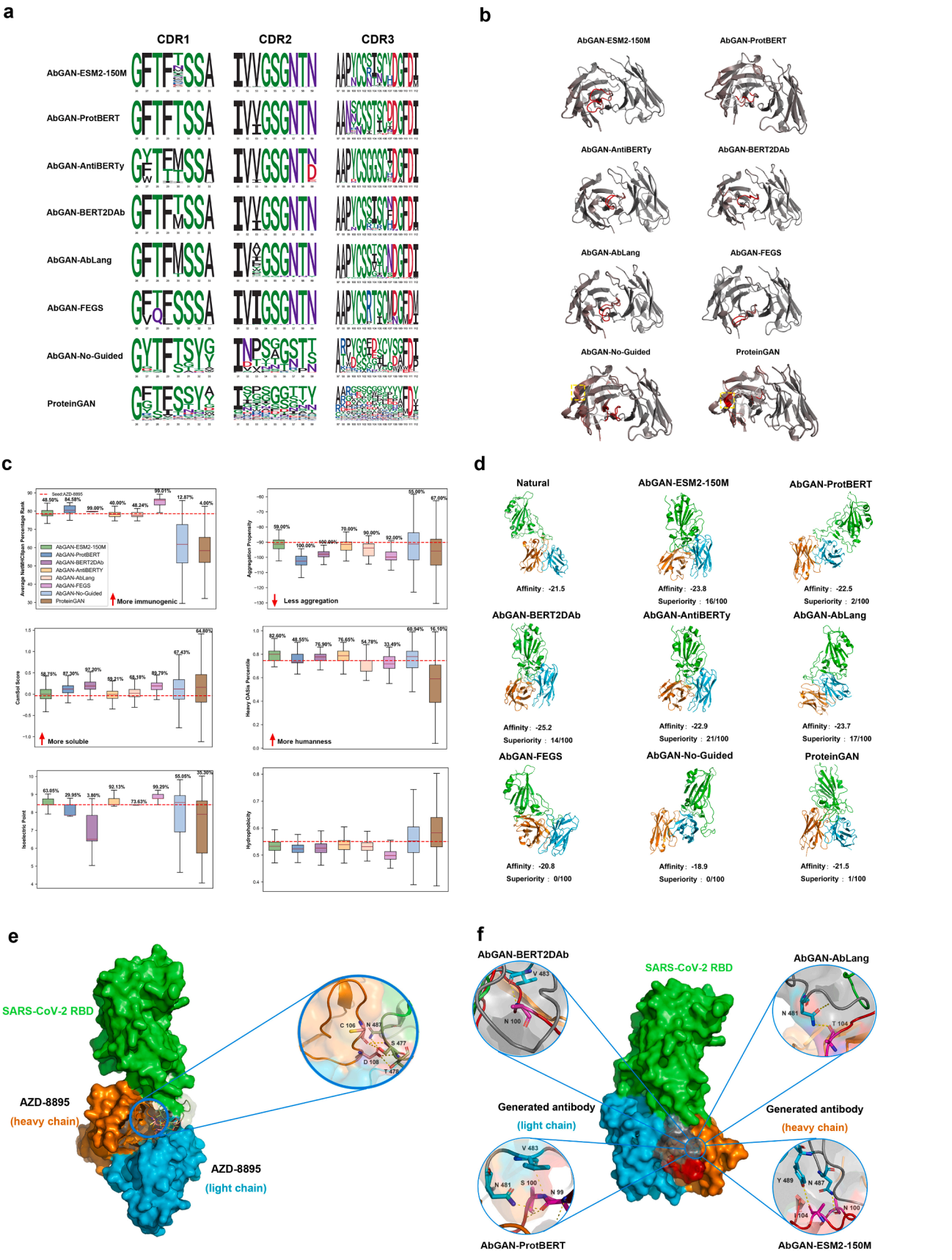
Fig. 5. AbGAN-LMG catches the amino acids distributions of antibody sequences. a. Amino acids pair correlation is shown in a correlation matrix. A higher Pearson correlation indicates stronger consistency in the distribution of amino acid pairs between the generated sequences and the wild-type sequences. b. The multiple sequence alignment between the generated sequence and the wild-type sequence is shown. Higher Pearson correlation indicates greater consistency in the global distribution between the generated and wild-type sequences.

Initially, we analyzed the complexity of various residue positions in the high variability regions of the antibody sequences generated by AbGAN-LMG based on AZD-8895 as the antibody template. Some residue positions in the CDR1, CDR2, and CDR3 regions showed conservation. The non-conserved residue positions exhibited consistent changes in different sequences generated by different AbGAN-LMG models (Fig. 6 [a]). However, the changes in the baseline models' generated sequences in these regions were more random.

Next, we assessed the structural stability of the antibodies generated by AbGAN-LMG based on AZD-8895. We randomly selected 100 antibodies from each AbGAN-LMG antibody based on AZD-8895 and performed structure modeling using Igfold. Subsequently, we calculated the DOPE score and per-residue LDDT value using modeller [58] and Pyrosetta [59] tools, respectively. The antibodies generated by the models without language model-guided feature vectors exhibited lower structural stability than AZD-8895 (Table 4). In contrast, the antibodies generated by the models with language model-guided feature vectors demonstrated better structural stability. Additionally, The models without language model-guided feature vectors tended to generate disordered structures with larger changes in the CDR3 region, while the models with language model-guided feature vectors tended to generate ordered structures with smaller changes in the CDR3 region (Fig. 6 [b]). AbGAN-No-Guided and ProteinGAN generated antibodies showed cases of increased or decreased β-strand content in the CDR1 or CDR2 regions.

Thirdly, we assessed the developability of the antibodies generated by AbGAN-LMG based on AZD-8895. We calculated various indicators, including aggregation, solubility, humanization potential, and immunogenicity for the generated antibodies. Based on the distribution of the bar chart and the proportion of sequences that outperformed AZD-8895, the majority of antibodies generated by AbGAN-LMG achieved over 50% of generated sequences with improved properties compared to AZD-8895 (Fig. 6[c]). This indicates that AbGAN-LMG can generate optimized antibodies with higher developability than the original unoptimized antibody. Furthermore, we found that the model using BERT2DAb-guided feature vectors (AbGAN-BERT2DAb) exhibited the best or second-best performance across all the evaluated indicators (Supplementary File A.6).

Next, we compared the generated sequences with known effective antibodies. We obtained the AZD-8895 antibody-antigen complex file (8D8R) from the PDB bank. We targeted the CDR3 of AZD-8895 to identify the hydrogen bonding sites with the SARS-CoV-2 RBD. Our analysis revealed that C106 and D108 of AZD-8895 form hydrogen bonds with SARS-CoV-2 RBD, suggesting their pivotal role in binding with SARS-CoV-2 RBD (Fig. 6[e]). Comparing these residues with the generated antibody sequence, we found a high degree of conservation in C106 and D108 (Fig. 6[a]). This signifies that the generative model has learned relevant binding information. We also observed hydrogen bond formations between the altered residues in the generated antibody



(caption on next page)

Fig. 6. AbGAN-LMG generates better candidate antibodies. a. The amino acid conversions in the CDR of the antibody library were measured. A larger letter size indicates a higher probability of occurrence at this position. b. The RMSD value of the 3D structure of the generated antibody compared to the 3D structure of the wild-type antibody is displayed. The color transitions from gray to pink, where more pink represents a larger RMSD value, indicating significant structural changes. The yellow box highlights cases where the 3D structure has gained or lost β-strands compared to the wild-type antibody sequence. c. Biophysical properties, as well as the developability of the libraries, were measured. In this context, a lower aggregation propensity value is considered better, while higher values for solubility, humanization, and immunogenicity indicators are preferred. d. The 3D structure of the antibody with the highest affinity among the 100 sequences is visualized and labeled with its affinity value, where green represents the antigen, orange represents the antibody heavy chain, and blue represents the antibody light chain. Also, the count of the 100 docked antibodies with superior affinity to the original antibody is indicated. e. The molecular interactions between AZD-8895 CDRH3 and SARS-CoV-2 RBD. Five hydrogen bonds are formed through the residues C106 and D108 in AZD-8895 and S477, T478, and N487 in SARS-CoV-2 RBD. f. The molecular interactions between generated antibody CDRH3 and SARS-CoV-2 RBD. The hydrogen bonds occur at the sites where the sequence has changed. The name on the view indicates that the model generated this sequence. The 3D conformation here is not the actual conformation but rather a representative conformation sampled for this particular case.

Table 4
Evaluation of Stability in Generated Antibody 3D Structures.

Model	per-residue LDDT↑	DOPE↓
AbGAN-ESM2–150M	0.9103 ± 0.0141	2.3048 ± 0.0462
AbGAN-ProtBERT	0.9509 ± 0.0186	2.3424 ± 0.0544
AbGAN-BERT2DAb	0.9516 ± 0.0090	**2.2704 ± 0.0266**
AbGAN-AntiBERTy	**0.9541 ± 0.0221**	2.3399 ± 0.0485
AbGAN-AbLang	0.9539 ± 0.0935	2.2425 ± 0.0437
AbGAN-No-Guided	0.5272 ± 0.0437	2.4495 ± 0.0194
AbGAN-FEGS	0.9182 ± 0.0004	2.3545 ± 0.0017
ProteinGAN	0.3250 ± 0.0264	2.4783 ± 0.1913

sequences and the SARS-CoV-2 RBD (Fig. 6[f]). For instance, in the sequence generated by AbGAN-BERT2DAb, residue Y at position 100 was altered to N, leading to a hydrogen bond between N100 and V483. Additionally, the sequence generated by AbGAN-AbLang, residue I at position 104 was altered to T, leading to a hydrogen bond between T104 and N481. These introduced hydrogen bonds indicate that the model can enhance the antigen-antibody binding affinity by altering the sequence. In all, AbGAN-LMG identified 70 antibodies out of 500 generated sequences with an affinity superior to that of the wild-type antibody for the antigen (Fig. 6[d]).

Additionally, we applied AbGAN-LMG to generate and evaluate a library of nanobody VHH-01 against MERS-CoV. Remarkably, AbGAN-BERT2DAb showed lower developability in generating nanobody libraries, possibly due to BERT2DAb being trained without nanobody data, resulting in the model not learning relevant features of nanobodies. The other four AbGAN-LMG models maintained the biophysical properties consistent with the wild-type antibodies while generating libraries that identified sequences with higher affinity and improved developability (Supplementary File A.7).

5. Discussions

Our AbGAN-LMG model combines natural language models with a GAN to generate antibody libraries and optimize wild-type antibodies. We demonstrated that AbGAN-LMG can generate high-quality libraries for screening and can produce superior candidate antibodies. Additionally, we analyzed the performance of applying feature vectors from different language model outputs in the AbGAN-LMG model.

Compared to other available baseline models, the use of representations extracted from language models enabled our GAN to capture the evolutionary characteristics of antibody sequences. Unlike AbGAN-No-Guided and ProteinGAN sequences [23], which are generated without guided representations from language models, the sequences in the libraries generated by AbGAN-LMG display certain evolutionary tendencies in the CDR region. This difference might be explained by the language models' input of biophysical properties, evolutionary information, and relationships between amino acids into the GAN. The embedding of this data allowed the model to generate antibody sequences that align with evolutionary tendencies. This finding fits with those of numerous other studies, which have suggested that language models have a powerful ability to represent and embed relevant information in larger programs. For example, IgFold [38] combines the language model AntiBERTy [60] with graph neural networks to quickly and accurately achieve predictions of antibody 3D structures. RGN2 [61] uses a PLM to analyze potential structural information of individual proteins, resulting in a shorter and better-performing computation of predictions than comparable models. DRN-1D2D_Inter [62] combines a hybrid residual network with a language model to significantly improve the prediction performance of protein contacts. In our research, we found that both AbGAN-FEGS and AbGAN-LMG generated libraries with good developability, but AbGAN-LMG can generate more candidate antibodies with stronger antigen binding affinity. Residue variations in the generated antibodies indicated that AbGAN-LMG's generated antibodies exhibited a high degree of conservation at important antigen-binding sites. Conversely, AbGAN-No-Guided, AbGAN-FEGS, and ProteinGAN did not display such conservation at these crucial sites, and as a result, failed to generate high-affinity antibodies. Moreover, the altered residues in the antibody sequences generated by AbGAN-LMG form new hydrogen bonds with the antigen. These observations are compelling evidence of the model's capability to enhance affinity and suggest that the traditional methods of relying on amino acid composition and physiochemical properties to identify antibodies might not offer the best way to maximize affinity. By contrast, the evolutionary information in the representation extracted by the language models played a crucial guiding role for our GAN, allowing the resulting model to generate high-quality libraries and discover superior candidate antibodies with strong affinity.

Different types of language models represented diversity and evolutionary direction in different ways to the GAN in our study. The antibody sequences generated by the GAN guided by the Protein Language Models (PLMs) exhibited higher diversity in our study, while the GAN guided by Autoregressive Language Models (ALMs) better fit the real distribution of antibody sequences. Specifically, the PLM-guided AbGAN-ProtBERT identified 2 antibodies with stronger binding affinity than the wild-type, whereas the same GAN guided by ALMs (AntiBERTy, BERT2DAb, and AbLang) were able to screen an average of 17 antibodies with stronger binding affinity. This may be explained by the training data for PLMs being more extensive, introducing more diversity into the model. In contrast, the training data for ALMs consists solely of antibody sequences, resulting in a more focused model that can better capture the evolutionary characteristics of antibody sequences. Additionally, because ALM BERT2DAb incorporated important secondary structure information of antibody sequences into the tokenization strategy, the libraries generated by the GAN this model guided demonstrated better developability. In light of these findings, we recommend using ALMs for sequence representation in future programs. Additionally, fine-tuning general language models with antibody sequence data or training domain-specific antibody language models could prove useful for such applications.

This work has certain limitations. Our results have not been validated through wet lab experiments, though we have evaluated the performance of AbGAN-LMG through multiple benchmark tests and provided open access to our code and model parameters to encourage researchers to reproduce our results. Additionally, the current

generative models are designed only for antibody-heavy chains and do not account for the pairing of light and heavy chains. Further improvements therefore must be made to the models before their practical application. During generation, the models only consider the antibody information and not that of relevant antigens. Future work should incorporate antigen information in these models to support the generation of high-affinity antibodies.

6. Conclusion

We designed AbGAN-LMG, a GAN that takes important guiding information from language models to optimize antibodies. By integrating information from these language models, AbGAN-LMG improves the quality of generated antibody libraries and increases the likelihood of finding antibodies with higher affinity and developability than known examples. The improved efficiency and accuracy in generating high-quality new antibodies positions AbGAN-LMG as a pivotal tool for developing potent new therapies. Although we focused on antibody generation in this work, similar strategies in computer modeling may be useful for proteins generally. For example, a universal protein sequence generative model may be designed and applied to identify functional protein sites.

Code availability statement

The source code, Generative model of AbGAN-LMG and analysis in the study are available in GitHub: https://github.com/Zhaowenbin98/AbGAN-LMG. The AbGAN-LMG website is freely accessible at http://39.102.71.224:88/.

CRediT authorship contribution statement

Wenbin Zhao: Methodology, Software, Validation, Formal analysis, Data curation, Investigation, Writing – original draft, Visualization Preparation. **Xiaowei Luo:** Methodology, Software, Validation, Formal analysis, Data curation, Investigation, Writing – original draft, Writing – review & editing. **Fan Tong:** Methodology, Software, Formal analysis, Writing – review & editing. **Xiangwen Zheng:** Methodology, Software, Formal analysis, Writing – review & editing. **Jing Li:** Methodology, Resources, Writing – review & editing. **Guangyu Zhao:** Methodology, Resources, Writing – review & editing. **Dongsheng Zhao:** Conceptualization, Methodology, Resources, Formal analysis, Writing – review & editing, Supervision, Project administration.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The Cov-AbDab dataset that support the model training of this study are available in https://opig.stats.ox.ac.uk/webapps/covabdab/.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.csbj.2023.11.041.

References

[1] Castelli MS, McGonigle P, Hornby PJ. The pharmacology and therapeutic applications of monoclonal antibodies. Pharm Res Perspect 2019;7:e00535. https://doi.org/10.1002/prp2.535.
[2] Kaplon H, Crescioli S, Chenoweth A, Visweswaraiah J, Reichert JM. Antibodies to watch in 2023. MAbs 2023;15:2153410. https://doi.org/10.1080/19420862.2022.2153410.
[3] Lyu X, Zhao Q, Hui J, Wang T, Lin M, Wang K, et al. The global landscape of approved antibody therapies. Antib Ther 2022;5:233–57. https://doi.org/10.1093/abt/tbac021.
[4] Yang L, Liu W, Yu X, Wu M, Reichert JM, Ho M. COVID-19 antibody therapeutics tracker: a global online database of antibody therapeutics for the prevention and treatment of COVID-19. Antib Ther 2020;3:205–12. https://doi.org/10.1093/abt/tbaa020.
[5] Zhang J, Zhang H, Sun L. Therapeutic antibodies for COVID-19: is a new age of IgM, IgA and bispecific antibodies coming? MAbs 2022;14:2031483. https://doi.org/10.1080/19420862.2022.2031483.
[6] Wang B, Gallolu Kankanamalage S, Dong J, Liu Y. Optimization of therapeutic antibodies. Antib Ther 2021;4:45–54. https://doi.org/10.1093/abt/tbab003.
[7] Bai G, Sun C, Guo Z, Wang Y, Zeng X, Su Y, et al. Accelerating antibody discovery and design with artificial intelligence: Recent advances and prospects. Semin Cancer Biol 2023;95:13–24. https://doi.org/10.1016/j.semcancer.2023.06.005.
[8] Li J, Kang G, Wang J, Yuan H, Wu Y, Meng S, et al. Affinity maturation of antibody fragments: a review encompassing the development from random approaches to computational rational optimization. Int J Biol Macromol 2023;247:125733. https://doi.org/10.1016/j.ijbiomac.2023.125733.
[9] Sormanni P, Aprile FA, Vendruscolo M. Third generation antibody discovery methods: in silico rational design. Chem Soc Rev 2018;47:9137–57. https://doi.org/10.1039/c8cs00523k.
[10] The I-SPY COVID Consortium, Writing group, Calfee CS, Liu KD, Asare AL, Beitler JR, et al. Clinical trial design during and beyond the pandemic: the I-SPY COVID trial. Nat Med 2022;28:9–11. https://doi.org/10.1038/s41591-021-01617-x.
[11] Brightling CE, Evans RA, Singapuri A, Smith N, Wain LV, Brightling CE, et al. Long COVID research: an update from the PHOSP-COVID Scientific Summit. Lancet Respir Med 2023;11:e93–4. https://doi.org/10.1016/S2213-2600(23)00341-7.
[12] Chungyoun MF, Gray JJ. AI models for protein design are driving antibody engineering. Curr Opin Biomed Eng 2023;28:100473. https://doi.org/10.1016/j.cobme.2023.100473.
[13] Zhou Y, Huang Z, Li W, Wei J, Jiang Q, Yang W, et al. Deep learning in preclinical antibody drug discovery and development. Methods 2023;218:57–71. https://doi.org/10.1016/j.ymeth.2023.07.003.
[14] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez AN, et al. Attention Is All You Need. In: Guyon, Luxburg U, Bengio S, Wallach H, Fergus R, Vishwanathan S, et al., editors. ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 30 (NIPS 2017), vol. 30, LA JOLLA: Neural Information Processing Systems (Nips); 2017.
[15] Gao J., He D., Tan X., Qin T., Wang L., Liu T.-Y. Representation Degeneration Problem in Training Natural Language Gener ation Models. 2019.
[16] Goodfellow I.J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., et al. Generative Adversarial Networks 2014.
[17] Huang T, Li Y. Current progress, challenges, and future perspectives of language models for protein representation and protein design. Innovation 2023;4:100446. https://doi.org/10.1016/j.xinn.2023.100446.
[18] Richard W.S., Jeffrey A.R., Jeffrey J.G. Generative language modeling for antibody design. bioRxiv 2022:2021.12.13.472419. https://doi.org/10.1101/2021.12.13.472419.
[19] Xu X, Xu T, Zhou J, Liao X, Zhang R, Wang Y, et al. AB-Gen: antibody library design with generative pre-trained transformer and deep reinforcement learning. S167202292300092X Genom, Proteom Bioinforma 2023. https://doi.org/10.1016/j.gpb.2023.03.004.
[20] Melnyk I., Chenthamarakshan V., Chen P.-Y., Das P., Dhurandhar A., Padhi I., et al. Reprogramming Pretrained Language Models for Antibody Sequence Infilling 2023.
[21] Lin E, Lin C-H, Lane H-Y. De novo peptide and protein design using generative adversarial networks: an update. J Chem Inf Model 2022;62:761–74. https://doi.org/10.1021/acs.jcim.1c01361.
[22] Wu Z, Johnston KE, Arnold FH, Yang KK. Protein sequence design with deep generative models. Curr Opin Chem Biol 2021;65:18–27. https://doi.org/10.1016/j.cbpa.2021.04.004.
[23] Repecka D, Jauniskis V, Karpus L, Rembeza E, Rokaitis I, Zrimec J, et al. Expanding functional protein sequence spaces using generative adversarial networks. Nat Mach Intell 2021;3:324–33. https://doi.org/10.1038/s42256-021-00310-5.
[24] Gupta A, Zou J. Feedback GAN for DNA optimizes protein functions. Nat Mach Intell 2019;1:105–11. https://doi.org/10.1038/s42256-019-0017-4.
[25] Kucera T, Togninalli M, Meng-Papaxanthos L. Conditional generative modeling for de novo protein design with hierarchical functions. Bioinformatics 2022;38:3454–61. https://doi.org/10.1093/bioinformatics/btac353.
[26] Amimeur T, Shaver JM, Ketchem RR, Taylor JA, Clark RH, Smith J, et al. Designing feature-controlled humanoid antibody discovery libraries using generative adversarial networks. Immunology 2020. https://doi.org/10.1101/2020.04.12.024844.
[27] Hovden AO, Haldorsen K. The seventh edition of the janeway's immunobiology. 112–112 Scand J Immunol 2008;68. https://doi.org/10.1111/j.1365-3083.2008.02123.x.
[28] Bonidia RP, Domingues DS, Sanches DS, de Carvalho A. MathFeature: feature extraction package for DNA, RNA and protein sequences based on mathematical descriptors. Brief Bioinform 2022;23. https://doi.org/10.1093/bib/bbab434.

[29] Ismail H, White C, Al-Barakati H, Newman RH, Kc DB. FEPS: a tool for feature extraction from protein sequence. Methods Mol Biol 2022;2499:65–104. https://doi.org/10.1007/978-1-0716-2317-6_3.
[30] Mu Z, Yu T, Liu X, Zheng H, Wei L, Liu J. FEGS: a novel feature extraction model for protein sequences and its applications. BMC Bioinforma 2021;22:297. https://doi.org/10.1186/s12859-021-04223-3.
[31] Li L, Gupta E, Spaeth J, Shing L, Jaimes R, Engelhart E, et al. Machine learning optimization of candidate antibody yields highly diverse sub-nanomolar affinity antibody libraries. Nat Commun 2023;14:3454. https://doi.org/10.1038/s41467-023-39022-2.
[32] Hie BL, Shanker VR, Xu D, Bruun TUJ, Weidenbacher PA, Tang S, et al. Efficient evolution of human antibodies from general protein language models. Nat Biotechnol 2023. https://doi.org/10.1038/s41587-023-01763-2.
[33] Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. Science 2023;379:1123–30. https://doi.org/10.1126/science.ade2574.
[34] Elnaggar A, Heinzinger M, Dallago C, Rehawi G, Wang Y, Jones L, et al. ProtTrans: toward understanding the language of life through self-supervised learning. IEEE Trans Pattern Anal Mach Intell 2022;44:7112–27. https://doi.org/10.1109/TPAMI.2021.3095381.
[35] Devlin J, Chang M-W, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. Association for Computational Linguistics; 2019. p. 4171–86. https://doi.org/10.18653/v1/N19-1423.
[36] Ruffolo J.A., Gray J.J., Sulam J. Deciphering antibody affinity maturation with language models and weakly supervised learning 2021.
[37] Kovaltsuk A, Leem J, Kelm S, Snowden J, Deane CM, Krawczyk K. Observed antibody space: a resource for data mining next-generation sequencing of antibody repertoires. J Immunol 2018;201:2502–9. https://doi.org/10.4049/jimmunol.1800708.
[38] Ruffolo JA, Chu LS, Mahajan SP, Gray JJ. Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies. Nat Commun 2023;14:2389. https://doi.org/10.1038/s41467-023-38063-x.
[39] Olsen TH, Moal IH, Deane CM. AbLang: an antibody language model for completing antibody sequences. Bioinform Adv 2022;2:vbac046. https://doi.org/10.1093/bioadv/vbac046.
[40] Raybould MIJ, Kovaltsuk A, Marks C, Deane CM. CoV-AbDab: the coronavirus antibody database. Bioinformatics 2021;37:734–5. https://doi.org/10.1093/bioinformatics/btaa739.
[41] Steinegger M, Söding J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. Nat Biotechnol 2017;35:1026–8. https://doi.org/10.1038/nbt.3988.
[42] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, et al. Generative adversarial nets 2014.
[43] Paszke A., Gross S., Massa F., Lerer A., Bradbury J., Chanan G., et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library 2019:arXiv:1912.01703. https://doi.org/10.48550/arXiv.1912.01703.
[44] Jang E., Gu S., Poole B. Categorical Reparameterization with Gumbel-Softmax 2016:arXiv:1611.01144. https://doi.org/10.48550/arXiv.1611.01144.
[45] Holland TL, Ginde AA, Paredes R, Murray TA, Engen N, Grandits G, et al. Tixagevimab–cilgavimab for treatment of patients hospitalised with COVID-19: a randomised, double-blind, phase 3 trial. Lancet Respir Med 2022;10:972–84. https://doi.org/10.1016/S2213-2600(22)00215-6.
[46] Kotowski K, Smolarczyk T, Roterman-Konieczna I, Stapor K. ProteinUnet—an efficient alternative to SPIDER3-single for sequence-based prediction of protein secondary structures. J Comput Chem 2020;42.
[47] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: machine learning in python. J Mach Learn Res 2011;12:2825–30. https://doi.org/10.48550/arXiv.1201.0490.
[48] Sievers F, Wilm A, Dineen D, Gibson TJ, Karplus K, Li W, et al. Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. Mol Syst Biol 2011;7:539. https://doi.org/10.1038/msb.2011.75.
[49] Santoni D, Felici G, Vergni D. Natural vs. random protein sequences: discovering combinatorics properties on amino acid words. J Theor Biol 2016;391:13–20. https://doi.org/10.1016/j.jtbi.2015.11.022.
[50] Kuriata A, Iglesias V, Pujols J, Kurcinski M, Kmiecik S, Ventura S. Aggrescan3D (A3D) 2.0: prediction and engineering of protein solubility. Nucleic Acids Res 2019;47:W300–7. https://doi.org/10.1093/nar/gkz321.
[51] Sormanni P, Aprile FA, Vendruscolo M. The CamSol method of rational design of protein mutants with enhanced solubility. J Mol Biol 2015;427:478–90. https://doi.org/10.1016/j.jmb.2014.09.026.
[52] Prihoda D, Maamary J, Waight A, Juan V, Fayadat-Dilman L, Svozil D, et al. BioPhi: a platform for antibody design, humanization, and humanness evaluation based on natural antibody repertoires and deep learning. MAbs 2022;14:2020203. https://doi.org/10.1080/19420862.2021.2020203.
[53] Lundegaard C, Lamberth K, Harndahl M, Buus S, Lund O, Nielsen M. NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8-11. Nucleic Acids Res 2008;36:W509–12. https://doi.org/10.1093/nar/gkn202.
[54] Zemla A. LGA program: a method for finding 3-D similarities in protein structures 2000.
[55] Shen M-y, Sali A. Statistical potential for assessment and prediction of protein structures. Protein Sci 2006.
[56] Jimenez-Garcia B, Roel-Touris J, Romero-Durana M, Vidal M, Jimenez-Gonzalez D, Fernandez-Recio J. LightDock: a new multi-scale approach to protein-protein docking. Bioinformatics 2018;34:49–55. https://doi.org/10.1093/bioinformatics/btx555.
[57] Vangone A, Bonvin A. PRODIGY: a contact-based predictor of binding affinity in protein-protein complexes. Bio Protoc 2017;7:e2124. https://doi.org/10.21769/BioProtoc.2124.
[58] Eswar N., Webb B., Marti-Renom M., Madhusudhan M.S., Eramian D., Shen M.-Y., et al. Comparative protein structure modeling using MODELLER. Current Protocols in Protein Science / Editorial Board, John E. Coligan. [et Al] 2007;Chapter 2:Unit 2.9. https://doi.org/10.1002/0471140864.ps0209s50.
[59] Adolf-Bryfogle J, Kalyuzhniy O, Kubitz M, Weitzner BD, Hu X, Adachi Y, et al. RosettaAntibodyDesign (RAbD): a general framework for computational antibody design. PLoS Comput Biol 2018;14:e1006112. https://doi.org/10.1371/journal.pcbi.1006112.
[60] Hie BL, Yang KK, Kim PS. Evolutionary velocity with protein language models predicts evolutionary dynamics of diverse proteins. e6 Cell Syst 2022;13:274–85. https://doi.org/10.1016/j.cels.2022.01.003.
[61] Chowdhury R, Bouatta N, Biswas S, Floristean C, Kharkar A, Roy K, et al. Single-sequence protein structure prediction using a language model and deep learning. Nat Biotechnol 2022;40:1617–23. https://doi.org/10.1038/s41587-022-01432-w.
[62] Yunda S., Chengfei Y. Protein language model embedded geometric graphs power inter-protein contact prediction. bioRxiv 2023:2023.01.07.523121. https://doi.org/10.1101/2023.01.07.523121.