Controllable Protein Design by Prefix-Tuning Protein Language Models

Jiawei Luo[1], Xianliang Liu[1], Jiahao Li[1], Qingcai Chen[1], Junjie Chen [1,*]

[1] School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, Guangdong, China;

* Corresponding author: Junjie Chen; E-mail: junjiechen@hit.edu.cn.

E-mail addresses of the authors:
J. Luo: luojiawei@stu.hit.edu.cn
X. Liu: liuxianliang@stu.hit.edu.cn
J. Li: lijiahao@stu.hit.edu.cn
Q. Chen: qingcai.chen@hit.edu.cn
J. Chen: junjiechen@hit.edu.cn

Running Title: Protein Design with Prefix-Tuning

Conflict of Interest: The authors declare no competing interests.

Abstract

Designing novel proteins tailored for specific purposes (e.g. drug discovery, vaccine design) presents a promising approach to address various biomedical challenges. Due to the similarity between protein sequences and natural languages, motivated by the remarkable success in NLP tasks that pre-trained language models have enabled text generation with human-like capabilities, protein language models (ProtLMs) are constructed to generate protein sequences with a predictable function across large protein families. The text generation can be controllable by constructing prefix-phase as control tags to prompt NLP language models. However, the vocabulary of protein sequences only contains 20 amino acid residues, which is not like natural language vocabulary to make up flexible control tags. In this study, we propose a controllable protein design method, named PrefixProt, which utilizes prefix tuning to learn virtual tokens as control tags, enabling to efficiently prompt the pre-trained ProtLM for protein generation tailored for specific purposes. The virtual tokens can be learned on any protein properties by data-driven and are flexible to be combined for fine-grained control. To demonstrate the effectiveness of PrefixProt, we train three virtual tokens on alpha-helix structure dataset, antimicrobial peptide (AMP) dataset and anticancer peptide (ACP) dataset, respectively. Our results show that prefix virtual tokens are efficient to prompt the pre-trained ProtLM by optimizing fewer trainable parameters compared with fine-tuning, especially under low-data settings. When combining the virtual tokens, the proportion of generated proteins with multiple properties are significantly improved. Therefore, PrefixProt offers a flexible and controllable protein design solution. We anticipate that PrefixProt will contribute to drug discovery and biomedical advancement.

Keywords: controllable protein design; prefix tuning; protein language model.

Availability and implementation

The models and associated code are available at:
https://github.com/chen-bioinfo/PrefixProt

1 Introduction

Proteins perform diverse functions in various aspects of life, but when repurposed for new functions, they generally suffer from a series of limitations: instability, not having the desired geometry or size, too low recombinant expression levels for large-scale production, exhibiting undesired interactions with other proteins or folding through too complex processes, among others [27]. Since a protein's structure and function are determined by its amino acid sequence, exploring the amino acid space to design novel proteins tailored for specific purposes (e.g. drug discovery, vaccine design) presents a promising approach to address above biomedical challenges. In the past decades, protein design has achieved huge advancements, such as enzyme design [45], vaccine design [10], and protein drug design [7], through directed evolution-based methods [47] or heuristic search based methods [30,31]. However, these methods heavily rely on strongly natural selection pressures or homologous fragments respectively, limiting their exploration of the vast protein sequence space to a minuscule fraction.

The unprecedented protein datasets provide an opportunity for artificial intelligence (AI)-based protein design approaches, which seek to learn the sequence-function relationship directly from natural protein sequences. In contrast with the discriminative models that intend to train machine learning models to predict the properties of given protein sequences, generative models can generate novel protein sequences from the learned natural protein space, providing the capability to directly explore the functional amino acid space to design novel proteins tailored for specific purposes, without the need to test a large number of nonfunctional protein sequence variants. Recently, the widely used AI-based generative methods [16, 24] , such as variational autoencoder (VAE), generative adversarial networks (GANs) and protein language models (ProtLMs), are getting remarkable achievements in the field of protein design. For instance, VAE-based approaches (e.g. MSA VAE [21] and AR-VAE [22], Dean et al. [14], PepVAE [13]) model the distribution underlying natural proteins in an self-supervised manner to learn biologically meaningful representation space. By sampling data points in the representation space, the trained decoders in VAE-based approaches can generate novel and functional protein variants. GAN-based approaches (ProteinGAN [44], gcWGAN [25], ProteoGAN [28], Anand et al. [2] and GANDALF [48]) learn the probability distribution of natural proteins by adversarially training generator and discriminator, where the generator tries to fool the discriminator by generating proteins that have natural-like physical properties. Recently, motivated by the success in NLP tasks that pre-trained language models have enabled text generation with human-like capabilities, due to the similarity between protein sequences and natural languages, ProtLMs are constructed to generate protein sequences with a predictable function across large protein families [37, 38, 55]. The BERT-like ProtLMs [15] (epiBERTope [40], ESM-2 [46], or ProteinBERT [5]) are pre-trained by reconstructing the corrupted protein sequences, and the GPT-like ProtLMs [6] (ProtGPT2 [17], RITA [23], and DARK [35]) are trained to predict subsequent amino acids given a prefix context, which is well-known autoregressive training. The BERT-like models are most used in sequence embedding, while the GPT-like models excel at generating long and coherent sequences, showing strong potential in protein design.

However, above protein design methods learn a general amino acid space from natural proteins, subsequently, they are not controllable to design proteins with tailored properties. Some researchers employed their conditional variants to disentangle amino acid space according to various attributes toward controllable protein design. PepCVAE [12] employed a conditional VAE (CVAE) model to disentangle antimicrobial activity space. CLaSS [11] leveraged guidance from attribute classifiers and a rejection sampling scheme for conditional protein generation with the desired attributes. Feedback GAN [20] utilized feedback mechanism to guide GAN models to generate coding protein genes with desirable functions. Inspired by the success of prompt learning [19] in NLP that utilize the prompts to control text generation with varied topics and sentiments, ProGen2 [39] trained a 1.2B-parameter GPT-like model on ∼280M protein sequences conditioned on taxonomic and keyword tags such as molecular function and cellular component, enabling to control protein generation. But the combination of taxonomic and keyword tags are not flexible for fine-grained control. Moreover, the vocabulary of ProtLMs is confined to 20 amino acids, which are not like natural language vocabulary to make up control tags [6] (**Figure 1A**). The other strategy to fully leverage the advantage of the pre-trained ProtLMs on protein design is to fine-tune the whole ProtLMs on a specific domain or a particular task. However, fine-tuning needs to modify all the model parameters. Due to the huge number of parameters of ProtLMs, adapting ProtLMs to a specific domain or a particular task, especially under data-limited scenarios, usually encounters some challenges, including overfitting, catastrophic forgetting and model drift. Nevertheless, prefix-tuning [32], a lightweight alternative to fine-tuning for generation tasks, keeps language model parameters frozen, but optimizes a small continuous task-specific vector (called the prefix). Prefix-tuning draws inspiration from prompting, allowing subsequent tokens to attend to this prefix as if it were *virtual tokens*. Thus, prefix-tuning has the advantages of both prompt learning and fine-tuning.

In this study, we propose a controllable protein design method, named PrefixProt, which employs prefix-tuning to learn virtual tokens as control tags for any protein properties by data-driven. Subsequently, the learned virtual tokens are used to prompt pre-trained ProtLMs for generating proteins with tailored structures and functions. During the training process, only the prefix virtual tokens are trainable, while the pre-trained ProtLM is frozen. To demonstrate the effectiveness of PrefixProt, we train three virtual tokens on alpha-helix structure dataset, antimicrobial peptide (AMP) dataset and anticancer peptide (ACP) dataset, respectively. Our results show that prefix virtual tokens are efficient to prompt the pre-trained ProtLM by optimizing fewer trainable parameters compared with fine-tuning, especially under low-data settings. Moreover, these learned virtual tokens are combined to control protein generation with multiple properties (e.g. proteins with antimicrobial function and alpha-helix structure, proteins with antimicrobial and anticancer functions). To the best of our knowledge, this is the first study that leverages prefix tuning to prompt ProtLMs for protein sequence generation. We aspire for this method to gain widespread adoption and contribute to the advancement of drug design and protein discovery.

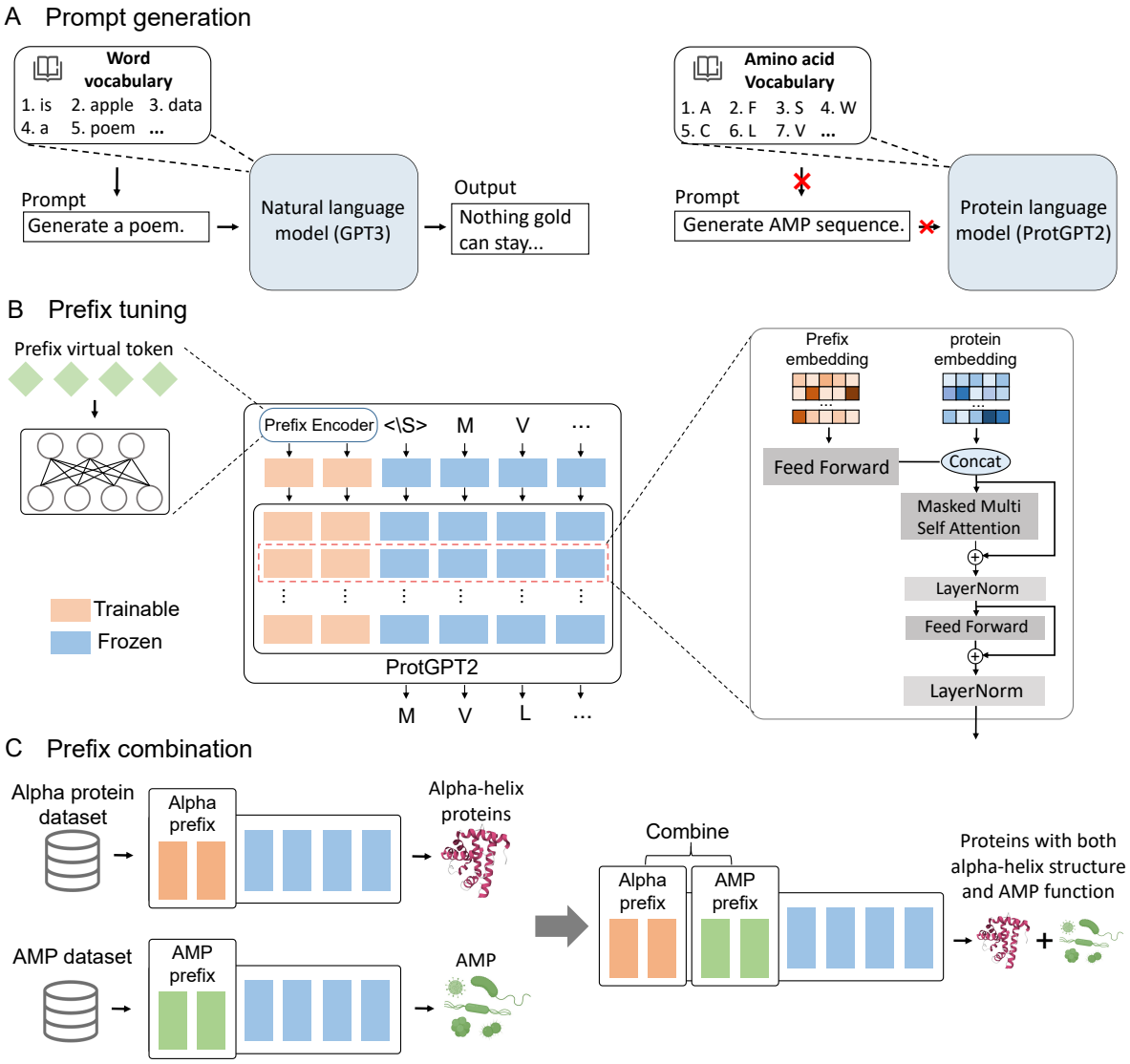

Figure 1: Illustration of the proposed PrefixProt. (A) Prompt generation. In contrast with the word vocabulary that can be used to compose the prompts for text generation, the amino acid vocabulary can't be used to construct valid prompts of protein language models. (B) Prefix-tuning architecture. The prefix virtual token is concatenated on the left of the protein sequence as the first context, thereby influencing the generation of amino acids on its right. The pre-trained protein language model's parameters are frozen and only the prefix-tuning parameters are trainable. (C) Prefix token combination. Multiple prefix virtual tokens can be combined to precisely control protein generation.

2 Methods

2.1 Datasets

We constructed three protein structure and function datasets, including alpha-helix structure dataset, antimicrobial function dataset and anticancer function dataset, to assess the effectiveness of prefix tuning in guiding the generation of ProtGPT2 from structure and function aspects, respectively. The alpha-helix structure dataset was collected from the Alpha class of Structural Classification of Proteins (SCOP2) database [3], which is created by manual inspection and automated methods to provide a comprehensive description of the structural and evolutionary connections among known proteins. After eliminating duplication, non-protein sequences, and sequences with non-standard amino acids, the final dataset contains 9,168 proteins. The antimicrobial function dataset was collected from the Database of Antimicrobial Activity and Structure of Peptides (DBAASP) [8,41]. Antimicrobial peptides (AMPs) are a class of small proteins with short amino acid sequences that possess broad-spectrum antimicrobial activity for the treatment of drug-resistant bacterial infections, making them an attractive alternative to traditional antibiotics [29,50]. AMPs were classified as active if their IC50 against *P. aeruginosa*, *A. baumanii*, or *S. aureus* are at least one below 10 μM, 10,000 nM, or 32 μgmL^{-1}. We got a total of 4,505 active peptides. The dataset of anticancer peptides was obtained from AntiCP2.0 [1], which comprises 861 experimentally validated anticancer peptides. In all experiments, 80% of the data is randomly sampled as training data, and the remaining 20% is the testing data. A dataset of peptides exhibiting both antimicrobial and anticancer functions was created by identifying overlapping sequences from the antimicrobial and anticancer peptide datasets. Furthermore, peptide datasets from the multi-functional peptide classification study MLBP [52], encompassing both antimicrobial and anticancer activities, were incorporated. The resultant dataset was utilized to establish a comprehensive reference for generated multi-functional peptide sequences.

2.2 Autoregressive Language Model

The autoregressive language model produces the next token based on its previous context in the sequence. Let $a = (a_1, ..., a_L)$ be a protein sequence with L amino acids. The chain rule of probability can be formulated as $p(x) = \prod_{i=1}^{L} p(a_i|a_{<i})$ [4]. An autoregressive ProtLM is to learn the distribution of amino acids in proteins, called linkage disequilibrium or evolutionary patterns in Proteomics. In practically, a ProtLM with parameters θ can be trained by minimizing the negative log-likelihood loss value across a given protein dataset $D = \{a^1, ..., a^{|D|}\}$. The loss function can be formulated as

$$L = \sum_{k=1}^{|D|} logp_\theta(a_i^k|a_{<i}^k) \ . \tag{1}$$

Here, we employed a pre-trained language model ProtGPT2 [17] as the base model for protein generation. ProtGPT2 is a transformer decoder-only language model trained

on UniRef50. Its architecture is similar to the previously released GPT2-large Transformer [42], which is constructed with a 36-layer transformer decoder and 12 attention heads in each layer with a model dimensionality of 1280.

2.3 Prefix-Tuning

Prefix tuning trains a virtual token as a prompt to guide ProtLM in generating proteins with desired functions and structures. The prefix token is a virtual token because it does not represent specific amino acids but is learned through training on a specialized dataset. The prefix token p is concatenated on the left of the protein sequence as the first context, thereby influencing the generation of amino acids on its right. Let $a' = [p; a]$ denote the concatenated input, p_i and $|p|$ be the i-th virtual prefix token and length of prefix respectively. The prefix embedding is generated by the prefix encoder $P_\beta \in R^{|p| \times d}$. The output h_i of the ProtLM with a prefix can be formulated as follows:

$$h_i = \begin{cases} P_\beta(i), & \text{if } i < |P| \\ ProtLM_\sigma(a'_i, h_{<i}), & otherwise \end{cases} , \tag{2}$$

where σ, β are parameters of the ProtLM and prefix encoder. The objective function of prefix tuning is the same as the autoregressive language model. Furthermore, to deal with the unstable optimization of parameters β, $P_\beta[i,:]$ is reparameterized by a matrix (P'_β) into $MLP_\beta(P'_\beta[i,:])$ by using feedforward neural network. During training, the parameters σ in the ProtLM are frozen, while only the parameters β in the prefix encoder are trainable. Thus, after training only the prefix encoder P_β needs to be stored.

2.4 Protein Sequence Generation

During the generation process, prefix tokens associated with desired structures and functions are first fed into prefix encoder to get related prefix embedding. Then in every generation step, a subsequent amino acid is produced according to its previous tokens. To increase the generative diversity and reduce the repeated amino acids, top-k sampling [42] with a repetition penalty [26] is employed, which is a sampling strategy that selects the subsequent tokens from the top k most probable tokens featured within the distribution generated by the ProtLM. Typically, the top k tokens are sampled by using the softmax function with a sampling temperature. The probability of the i-th token being sampled from the top k tokens is computed by Eq. (3).

$$p_i = \frac{\exp(\mathrm{z_i}/\mathrm{T})}{\sum_j \exp(\mathrm{z_j}/\mathrm{T})} , \tag{3}$$

where z_i is the prediction for token i, T is the temperature, and p_i is the sampling probability of token i. The next token is then chosen by sampling through a multinomial distribution with probabilities p_i clipped at the top-k tokens. To prevent repetition, a

repetition penalty is used in top-k sampling to reduce the probabilities of tokens that have been presented within priors. The probability of the i-th token is defined as Eq.(4)

$$p_i = \frac{\exp(z_i/T \cdot I(i \in g))}{\sum_j \exp(z_j/T \cdot I(j \in g)}, \ I(i \in g) = \theta \text{ if } i \in g \text{ is True else 1}, \tag{4}$$

where g represents a list of generated tokens.

2.5 Evaluation Metrics

We measured the quality of generated protein sequences by utilizing several metrics, including perplexity, percentage of alpha-helix, Rosetta energy score, protein function and physicochemical properties, as well as the identity.

Perplexity is a general metric in sequence generation tasks by calculating the exponential value of the cross-entropy loss computed for each token within a given dataset. A lower perplexity indicates a higher quality model. Percentage of alpha-helix is a measurement of protein structure, calculated by employing the DSSP tool to count the amino acids with alpha-helix structure in the protein structure predicted by ESMFold [33]. Each amino acid is labeled with the corresponding secondary structure property including α-helix(H), Isolated β-bridge residue(B), Strand(E), 3-10 helix(G), π-helix(I), Turn(T), Bend(S) and other(-). Rosetta energy scores were got by employing Rosetta Relax with the default Rosetta full-atom energy terms and weights in the PyRosetta [9]. Lower Rosetta energy conformers correlate with more relaxed structures [49]. To minimize the time needed for relax, we set the maximum number of iterations to 20. A general guideline is that the total score (Rosetta Energy Units, REU) should typically fall within the range of -1 to -3 per residue [54]. For protein physicochemical properties and functions, the physicochemical properties of generated AMPs and ACPs, including length, charge, charge density, hydrophobicity, hydrophobic moment, and isoelectric point were calculated by using modlAMP [36]. The antimicrobial function of generated AMPs were predicted by leveraging the CAMP server [53]. The anticancer function of generated ACPs were predicted through AntiCP2.0 [1]. Sequences with a length greater than 100 amino acids were removed before prediction. The sequence identity was performed by BLAST, and the structure identity is measured by TM score via TM-align [56].

2.6 Experimental Setup

To estimate the effectiveness of prefix tuning, we compared the proteins generated by prefix-tuned ProtGPT2 with the proteins generated by pre-trained ProtGPT2 and fine-tuned ProtGPT2, as well as two baselines: the natural proteins in testing data and the randomly generated proteins. All compared methods were trained in 100 epochs with the AdamW optimizer [34]. We set a learning rate of 5e-6 and 5e-3 for fine-tuning and prefix tuning with batch size 8 on the alpha-helix structure dataset, and a learning rate of 1e-6 and 1e-3 for fine-tuning and prefix tuning with batch size 32 on both antimicrobial and anticancer peptide datasets, respectively. We implemented a polynomial decay schedule with a final learning rate of 1e-7 and a polynomial exponent of 3. The

length of prefix virtual tokens is 100 and 20 for the alpha-helix structure dataset and functional protein dataset, respectively. During generation, the subsequent amino acids were sampled from the top 500 with repetition penalty of 1.2. The random dataset was created by randomly combining 20 amino acids according to their background distribution in the training dataset. We generated 2,000 proteins with alpha-helix structure, 1000 proteins with antimicrobial function and 200 proteins with anticancer function by each compared method.

3 RESULT

3.1 Controllable alpha-helix protein generation

In this section, we employed PrefixProt to learn a virtual token of alpha-helix structure, and evaluated its effectiveness to prompt ProtGPT2 to generate proteins with alpha-helix structure. The PrefixProt was compared with the pre-trained ProtGPT2, fine-tuned ProtGPT2, as well as random generation according to amino acid background distribution.

We first compared the percentage of amino acids with alpha-helical structure and the Rosetta energy score (Rosetta Energy Units, REU) on the generated proteins by all compared methods against the testing dataset. PrefixProt outperforms other methods in terms of both metrics. For the alpha-helix structures comparison (**Figure 2A**), although both PrefixProt and fine-tuned ProtGPT2 significantly improve the percentage of alpha-helix structure in generated proteins comparing to pre-trained ProtGPT2, PrefixProt achieves closer distribution to natural proteins in the testing dataset, having a higher proportion of amino acids with alpha-helix structures. To further estimate the model's capability to efficiently generate stable proteins, we calculated the Rosetta energy of generated proteins (**Figure 2B**). The energy scores of majority proteins are between -1 and -3 REU/residue, except proteins in the random generation dataset. The mean energy scores for testing, prefix-tuned, fine-tuned datasets are -1.59, -0.92, and -0.8 REU/residue, respectively. The proteins generated by PrefixProt have lower Rosetta energy scores than the proteins generated by the other methods, but are still slightly higher than the natural proteins in the testing dataset. The three 'all alpha' protein examples (**Figure 2C**) generated by PrefixProt also show the stability with low energy scores.

To further analyze the conformation and dihedral angular distribution of proteins, we conducted Ramachandran plots [43] to illustrate the co-occurrence frequency of dihedrals (ϕ, ψ) (**Figure 2D**). The results indicate that all methods are able to generate all three major secondary structure elements in protein backbones. Nonetheless, the proteins generated by PrefixProt exhibit greater similarity to the natural sequence conformation, while other methods produce dihedral angles that are more dispersed and less reasonable, scattered across the periphery. We also conducted a comparison of sequence identity for generated proteins against the training dataset (**Figure2E**). The proteins produced by PrefixProt exhibit a higher sequence identity, whereas the proteins generated by fine-tuning predominantly display identities falling within the range of 20%

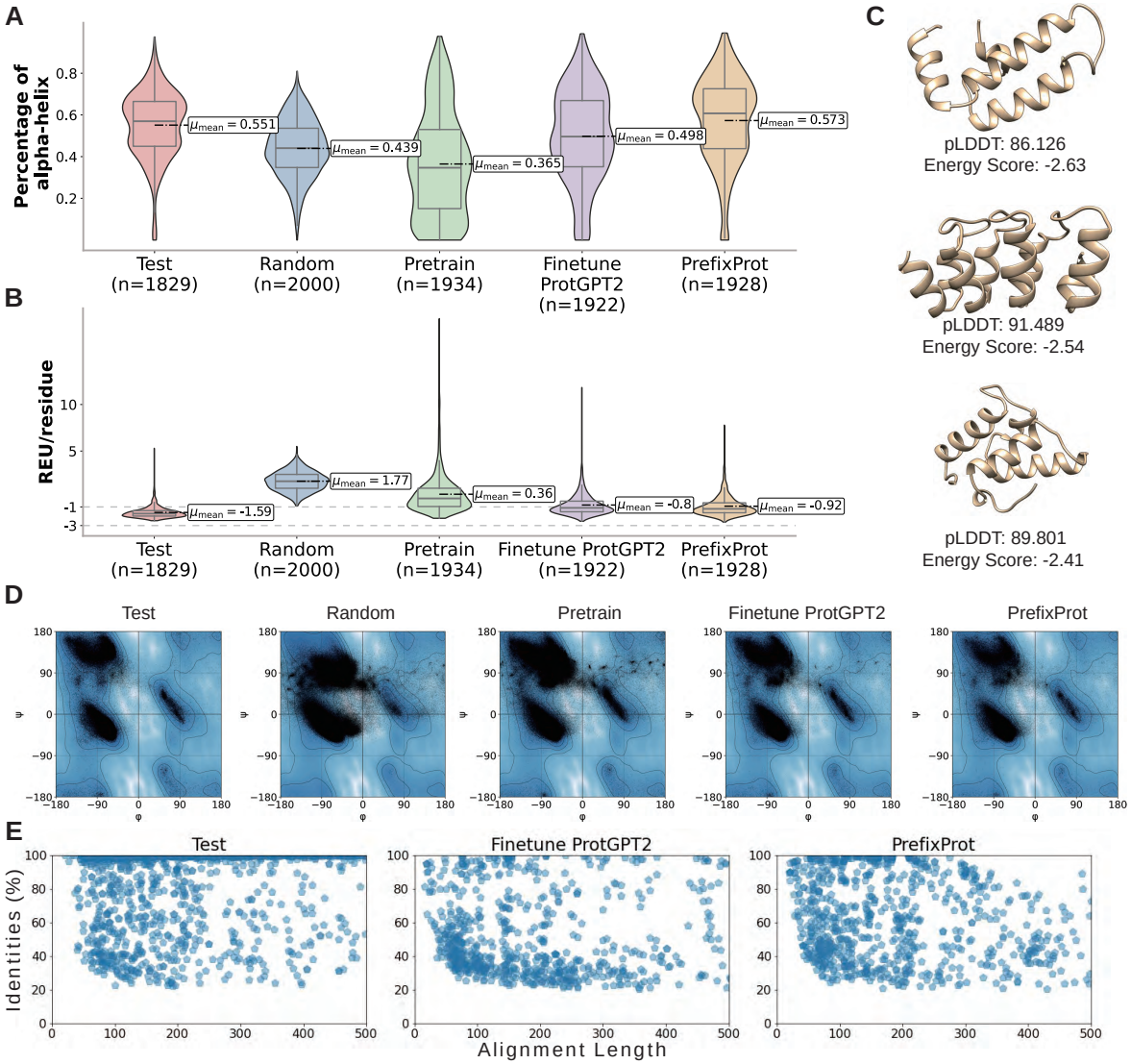

Figure 2: Comparison in generating proteins with alpha-helix structure. (A) Percentage of amino acids within alpha-helix structure in the compared datasets. (B) Average Rosetta energy units per residue in the compared datasets. (C) Three proteins with 'all alpha-helix' generated by prefix-tuned ProtGPT2. (D) Ramachandran plots comparing the (ϕ, ψ) dihedral angles for the testing set and generated proteins by different methods. (E) Pairwise sequence identity comparison of testing dataset, proteins generated by fine-tuned and prefix-tuned ProtGPT2 against training dataset. Note that the proteins generated by pre-trained ProtGPT2 and random generation have few sequences matched with training data. Thus their results are not presented.

to 50%. Overall, prefix tuning with a limited number of trainable parameters surpasses fine tuning with fully trainable parameters in generating proteins with stable energy and reasonable conformation, demonstrating the capability of prefix tuning to prompt large ProtLM with desired structures.

3.2 Controllable functional protein generation

In this section, we trained two prefix virtual tokens on antimicrobial protein and anticancer protein datasets, and evaluated their capability to generate the antimicrobial and anticancer proteins. We first compared the generated proteins in terms of 7 physicochemical properties, including amino acid distribution, length, charge, charge density, hydrophobicity, hydrophobic moment, and isoelectric point (**Supplementary Figure S1, S2**). Since the ProtGPT2 was pre-trained on a large-scale protein dataset, which includes peptides with short amino acid sequences and large proteins with long amino acid sequences, we limited the maximum length to 20 in the generation process to focus on peptide generation. Except for AMPs generated by pre-trained ProtGPT2, all other methods are able to produce AMPs and ACPs with amino acid distribution close to natural peptides with specific functional activity in the testing dataset. Note that the random generation is based on the background distribution of amino acids in the testing dataset. Thus it has the same distribution with training data. The other results of 6 physicochemical properties distribution show that the AMPs and ACPs generated by PrefixProt and fine-tuned ProtGPT2 are both notable resemblance in physicochemical properties of natural AMPs and ACPs in the testing dataset respectively, demonstrating that both of them are potential to direct ProtGPT2 toward the generation of protein sequences with target physicochemical properties [18].

We then compared antimicrobial and anticancer activity of generated by PrefixProt and several state-of-the-art methods. For the AMP generation, we employed PepLSTM [37], PepCVAE [12], and Hydramp [51]. PepLSTM designs AMPs by leveraging Long Short-Term Memory (LSTM) to recognize the grammar of amphipathicity in peptides. PepCVAE designs novel AMPs based on a semi-supervised variational autoencoder model. HydrAMP employed a conditional variational autoencoder to project amino acid sequences into reduced-dimensional, continuous representations. The activity of all generated AMPs was predicted by CAMP [53], an open-access AMP prediction web-server (**Figure 3A, B**). Due to the limited methods available for *de novo* design of anticancer peptides, our comparison focused exclusively on the performance of PrefixProt and fine-tuned ProtGPT2. The activity of all generated anticancer peptides was predicted using AntiCP2 [1]. It is obvious that PrefixProt produces AMPs and ACPs with higher probability of being active than the fine-tuned ProtGPT2. And the activity distributions of AMPs generated by PrefixProt and fine-tuned ProtGPT2 are significantly higher than other methods. Similarly, we also conducted a comparison of sequence identity for generated AMPs and ACPs against the training dataset (**Figure 3B**). Peptides produced by PrefixProt exhibit similar distribution to natural AMPs and ACPs, demonstrating they have higher sequence identity. In contrast, only one sequence generated by fine-tuned ProtGPT2 can be matched to the training set. Furthermore, amphipathicity has been demonstrated to be a significant property influencing the antimicrobial activity of antimicrobial peptides [18]. We generated helical wheel plots of 5 top-ranking AMPs generated by PrefixProt (**Figure 3C**). The plots show that the generated AMPs have the formation of amphipathic helical structures. The results suggest that prefix tuning can prompt ProtGPT2 to generate proteins with desired functions.

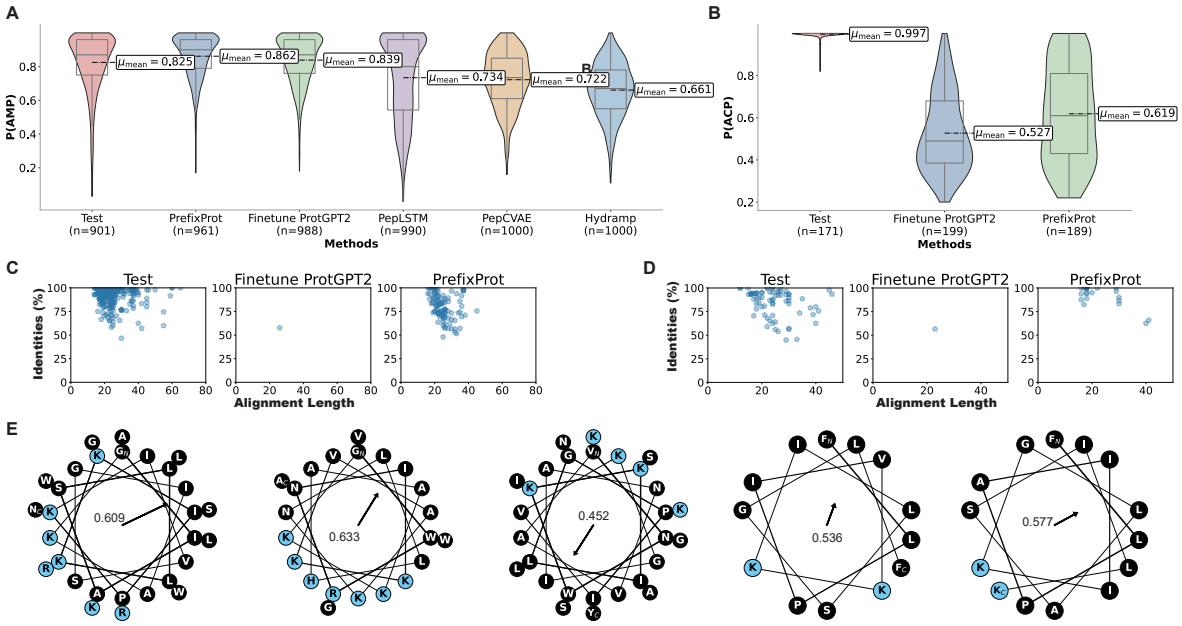

Figure 3: Comparison of generated functional proteins. (A) The probability distribution comparison of being the active AMPs. (B) The probability distribution comparison of being the active ACPs. (C) Pairwise sequence identity comparison of AMPs generated by fine-tuned ProtGPT2 and PrefixProt against training dataset. (D) Pairwise sequence identity comparison of anticancer peptides generated by fine-tuned ProtGPT2 and PrefixProt against training dataset. (E) The helical wheel plots of 5 top-ranking AMps generated by PrefixProt. The amino acids labeled with 'N' and 'C' subscript represent the N-terminus and C-terminus of the peptide sequences, respectively. The black circles represent hydrophobic residues, while the blue circles represent polar residues.

3.3 Combing virtual tokens for multi-properties control in protein generation

To further investigate the potential of prefix tuning in protein generation, we combined two prefix tokens to generate proteins with multi-properties.

3.3.1 Peptide generation with alpha-helix structure and antimicrobial function

We combined the alpha-helix token and antimicrobial token in two manners: concatenating or averaging. Regarding the limitation of input length and robustness of token combination, we retrained the alpha-helix token with length of 20. To illustrate the impact of the AMP token, we integrated the AMP token into the alpha-helix token, and conducted an ablation experiment by replacing the AMP token with randomly initial tokens. Since the AMPs are a class of small proteins with short amino acid sequences, we filtered out the generated sequences with length exceeding 100 amino acids. After filtering, the peptides prompted by only alpha-helix token are left with 254 sequences, with most of them having length around 80 amino acids and the average activity prob-

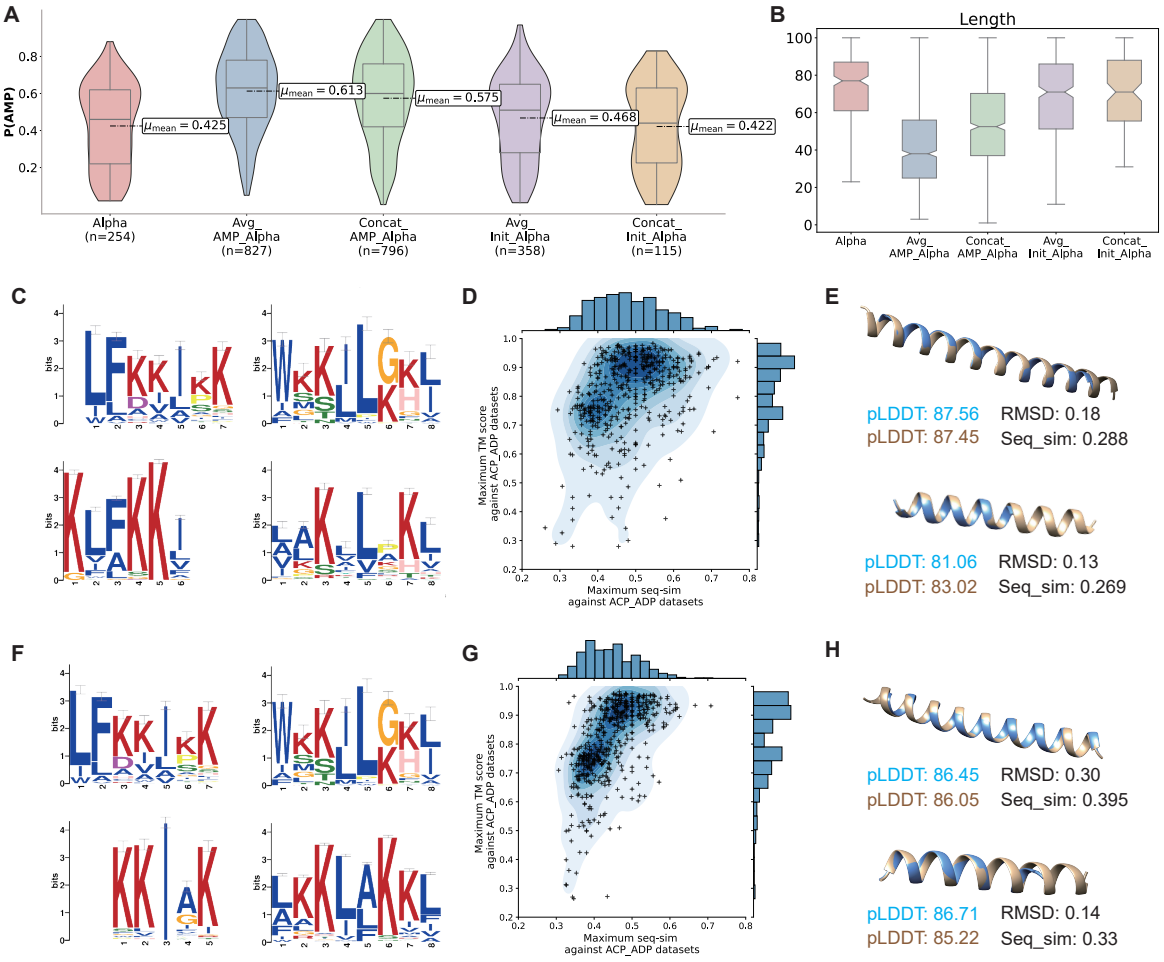

Figure 4: Comparison of multi-properties control in peptide generation. Activity (A) and sequence length (B) of peptides generated by PrefixProt with various virtual tokens. 'Avg_AMP_Alpha' and 'Concat_AMP_Alpha' indicate that AMP token and Alpha token were combined using averaging and concatenate manner, respectively. 'Init' represents the initial token without an training. (C, D, E) and (F, G, H) are results of peptides generated by PrefixProt with antimicrobial and anticancer virtual tokens through concatenating or averaging respectively. (C, F) shows the motif comparison for natural AMP_ACPs (top) and generated AMP_ACPs (bottom). (D, G) are the distribution of the highest sequence similarity (seq-sim) and structural similarity between the natural and generated AMP_ACPs. (E, H) presents the structure alignment of two samples, where the brown and blue colors represent the natural and generated AMP_ACPs, respectively.

ability of 0.425. When combining the AMP token into the alpha-helix token, especially using the averaging manner, the probability distribution of being AMP is notably improved and the length of generated peptides becomes shorter. However, the randomly initial token has no discernible impact on the peptide generation.

3.3.2 Peptide generation with antimicrobial and anticancer functions

We also evaluated the effectiveness of virtual token combination for multi-functional peptide generation. To remove the effect of multi-functional peptides in the training dataset, we initially preprocess the peptide dataset by excluding sequences that have both antimicrobial and anticancer functions (AMP_ACPs). Subsequently, we retrained the antimicrobial and anticancer tokens and combined them with two manners: concatenating or averaging. The functions of generated AMP_ACP peptides were predicted by two widely used peptide function predictors [1, 53].

Both two combination manners capture the patterns in natural AMP_ACP peptides by comparing motifs via STREME (Figure 4 C, F). The parameters of STREME is shown in **Supplementary Table S1**. Moreover, the distribution of the highest sequence similarity (seq-sim) and structural similarity (Figure 4 D, G) exhibit generated AMP_ACP peptides have low sequence similarity while keeping high structure similarity with natural ones, demonstrating the generated AMP_ACP peptides are novelty in amino acid sequence and effectiveness in functional activity. The structure alignments between generated AMP_ACPs and natural AMP_ACPs (Figure 4 E, H) also demonstrate the effectiveness of peptide generation with antimicrobial and anticancer functions. Although these two combination manners have no significant difference to the protein generation, selecting the appropriate combination manners enable more effective control of the generation process. When using the concatenation combination, the prefix length is the sum of length of all individual tokens, resulting in a greater number of parameters. Nonetheless, the averaging combination does not alter the prefix length. Besides, the sequences generated through concatenation tends to be slightly shorter because the virtual prefix words are part of the total sequence length. In multi-functional peptide generation, sequences generated with the averaging operation exhibit lower sequence similarity.

These results demonstrate that combining multiple virtual tokens has the potential to generate proteins with multiple properties simultaneously. Taking this strength, we can construct more virtual tokens to guide the ProtLM's behavior in a more precise manner, leading to more accurate and desired outputs.

3.4 Impact of low training data and length of prefix embedding

To evaluate the effectiveness of PrefixProt in low-data scenarios, we randomly subsampled the full antimicrobial function dataset to obtain 6 small datasets with 50, 100, 200, 500, 1000, 2000 AMPs. We compared PrefixProt and fine-tuned ProtGPT2 (**Figure 5A, B**). Although both of them exhibit an improvement trend in terms of activity probability of generated AMPs as the size of the training data increasing, PrefixProt outperforms the fine-tuned ProtGPT2 in all low-data settings, especially on the 4 lower-data settings with 50, 100, 200, 500 AMPs. But the gap narrows down as the dataset size increasing. These results highlight the effectiveness of prefix tuning when dealing with small datasets.

A longer virtual token results in more trainable parameters, leading to increased

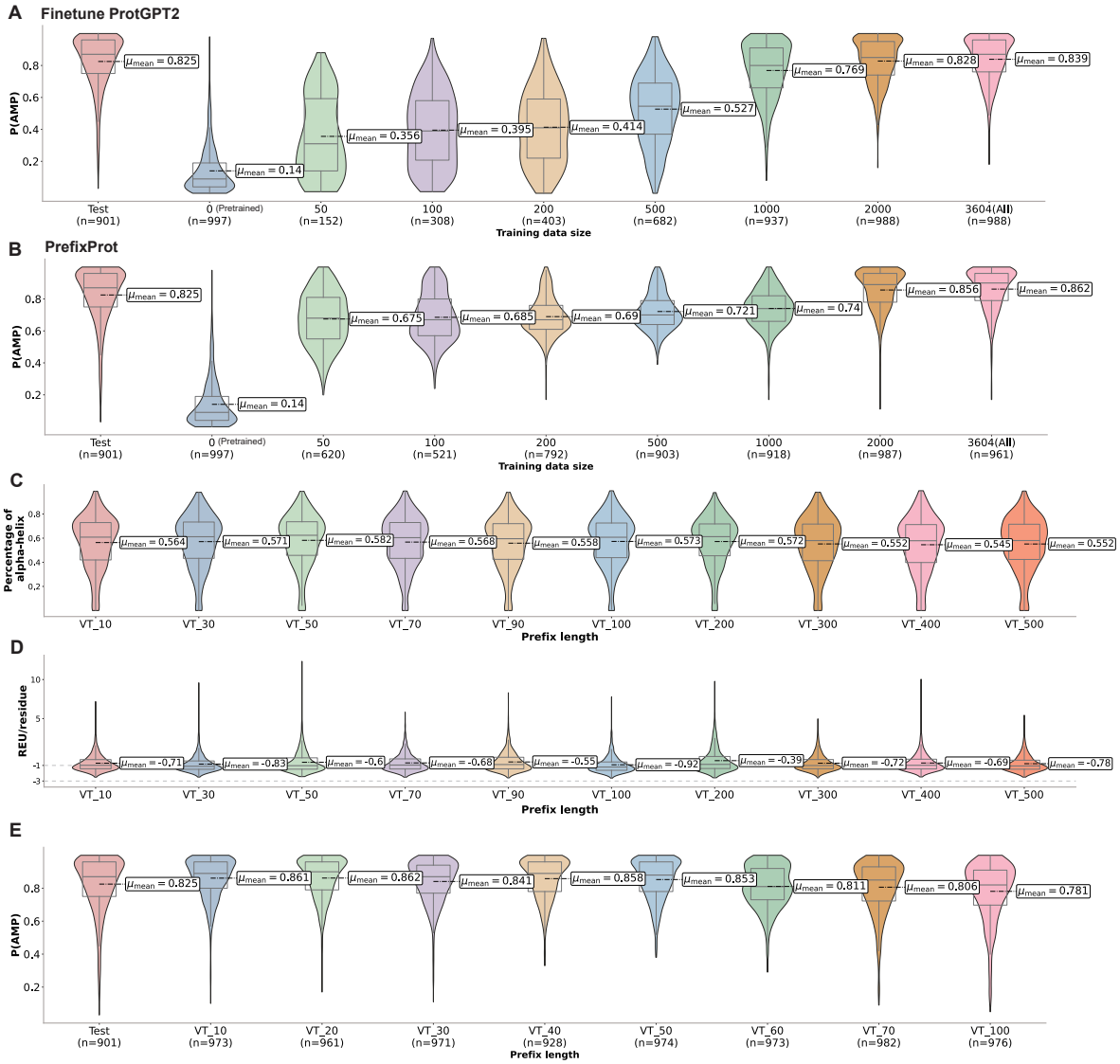

Figure 5: Comparison of antimicrobial activity in low-data settings and various prefix length settings. The probability distribution for AMPs generated by the fine-tuned ProtGPT2 (A) and the prefix-tuned ProtGPT2 (B) in a low-data setting. (C and D) The percentage of amino acids within alpha-helix structure and average Rosetta energy units per residue in the datasets generated by the prefix-tuned ProtGPT2 with various prefix length settings. (E) The probability distribution for AMPs generated by the prefix-tuned ProtGPT2 with various prefix length settings.

expressive power in the prompt generation, but more parameters also need larger data to train. We evaluated PrefixProt with different virtual token length settings (**Figure 5C-E**). In the alpha-helix protein generation, better Rosetta energy scores and comparable alpha-helix distributions are achieved when the virtual token length is 100. The performance reaches the peak at a threshold of 20 in the AMP generation and then slightly drops as the length increasing. These results indicate the token length is a

crucial hyper-parameter in protein generation.

4 Conclusion

In this study, we presented a controllable protein design method by introducing prefix tuning to learn virtual tokens as control tags, subsequently prompting large protein language models for generating proteins with desired properties. We evaluated the effectiveness of learned virtual tokens on alpha-helix structure, antimicrobial peptide and anticancer peptide. The results indicate prefix tuning is more efficient than fine tuning in terms of higher performance and fewer trainable parameters. Moreover, multiple prefix tokens can be combined to guide the protein language model in fine-grained control, leading to a more precise protein design.

In future study, we plan to integrate multiple modal information including structure-related representation and function-related features for protein generation. These methods can be effectively extended to various applications in the field of antibody and drug peptide discovery, thereby facilitating advancements in drug discovery research.

5 Acknowledgements

This work is supported by the National Key R&D Program of China (2022ZD0116002), Natural Science Foundation of China (62102118, 62276075, 61872113), Project of Educational Commission of Guangdong Province of China (2021KQNCX274), Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies, China (2022B1212010005), Shenzhen Colleges and Universities Stable Support Program (GXWD20220811170504001), and Shenzhen Science and Technology Program (JCYJ20230807094318038)

References

[1] Piyush Agrawal, Dhruv Bhagat, Manish Mahalwal, Neelam Sharma, and Gajendra PS Raghava. Anticp 2.0: an updated model for predicting anticancer peptides. *Briefings in bioinformatics*, 22(3):bbaa153, 2021.

[2] Namrata Anand, Raphael R. Eguchi, and Po-Ssu Huang. Fully differentiable full-atom protein backbone generation. In *DGS@ICLR*, 2019.

[3] Antonina Andreeva, Dave Howorth, Cyrus Chothia, Eugene Kulesha, and Alexey G Murzin. Scop2 prototype: a new approach to protein structure mining. *Nucleic acids research*, 42(D1):D310–D314, 2014.

[4] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

[5] Nadav Brandes, Dan Ofer, Yam Peleg, Nadav Rappoport, and Michal Linial. Proteinbert: a universal deep-learning model of protein sequence and function. *Bioinformatics*, 38:2102 – 2110, 2021.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[7] Longxing Cao, Brian Coventry, Inna Goreshnik, Buwei Huang, William Sheffler, Joon Sung Park, Kevin M Jude, Iva Marković, Rameshwar U Kadam, Koen HG Verschueren, et al. Design of protein-binding proteins from the target structure alone. *Nature*, 605(7910):551–560, 2022.

[8] Alice Capecchi, Xingguang Cai, Hippolyte Personne, Thilo Köhler, Christian van Delden, and Jean-Louis Reymond. Machine learning designs non-hemolytic antimicrobial peptides. *Chemical science*, 12(26):9221–9232, 2021.

[9] Sidhartha Chaudhury, Sergey Lyskov, and Jeffrey J Gray. Pyrosetta: a script-based interface for implementing molecular modeling algorithms using rosetta. *Bioinformatics*, 26(5):689–691, 2010.

[10] Bruno E Correia, John T Bates, Rebecca J Loomis, Gretchen Baneyx, Chris Carrico, Joseph G Jardine, Peter Rupert, Colin Correnti, Oleksandr Kalyuzhniy, Vinayak Vittal, et al. Proof of principle for epitope-focused vaccine design. *Nature*, 507(7491):201–206, 2014.

[11] Payel Das, Tom Sercu, Kahini Wadhawan, Inkit Padhi, Sebastian Gehrmann, Flaviu S. Cipcigan, Vijil Chenthamarakshan, Hendrik Strobelt, Cicero dos Santos, Pin-Yu Chen, Yi Yan Yang, Jeremy P. K. Tan, James L. Hedrick, Jason Crain, and Aleksandra Mojsilovic. Accelerated antimicrobial discovery via deep generative models and molecular dynamics simulations. *Nature Biomedical Engineering*, 5:613 – 623, 2021.

[12] Payel Das, Kahini Wadhawan, Oscar Chang, Tom Sercu, Cicero Dos Santos, Matthew Riemer, Vijil Chenthamarakshan, Inkit Padhi, and Aleksandra Mojsilovic. Pepcvae: Semi-supervised targeted design of antimicrobial peptide sequences. *arXiv preprint arXiv:1810.07743*, 2018.

[13] Scott N. Dean, Jerome Anthony E. Alvarez, Dan Zabetakis, Scott A. Walper, and Anthony P. Malanoski. Pepvae: Variational autoencoder framework for antimicrobial peptide generation and activity prediction. *Frontiers in Microbiology*, 12, 2021.

[14] Scott N Dean and Scott A Walper. Variational autoencoder for generation of antimicrobial peptides. *ACS omega*, 5(33):20746–20754, 2020.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[16] Wenze Ding, Kenta Nakai, and Haipeng Gong. Protein design via deep learning. *Briefings in bioinformatics*, 23(3):bbac102, 2022.

[17] Noelia Ferruz, Steffen Schmidt, and Birte Höcker. Protgpt2 is a deep unsupervised language model for protein design. *Nature communications*, 13(1):4348, 2022.

[18] Christopher D Fjell, Jan A Hiss, Robert EW Hancock, and Gisbert Schneider. Designing antimicrobial peptides: form follows function. *Nature reviews Drug discovery*, 11(1):37–51, 2012.

[19] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020.

[20] Anvita Gupta and James Zou. Feedback gan for dna optimizes protein functions. *Nature Machine Intelligence*, 1(2):105–111, 2019.

[21] Alex Hawkins-Hooker, Florence Depardieu, Sebastien Baur, Guillaume Couairon, Arthur Chen, and David Bikard. Generating functional protein variants with variational autoencoders. *PLoS Computational Biology*, 17, 2020.

[22] Alex Hawkins-Hooker, Florence Depardieu, Sebastien Baur, Guillaume Couairon, Arthur Chen, and David Bikard. Generating functional protein variants with variational autoencoders. *PLoS computational biology*, 17(2):e1008736, 2021.

[23] Daniel Hesslow, Niccoló Zanichelli, Pascal Notin, Iacopo Poli, and Debora S. Marks. Rita: a study on scaling up generative protein sequence models. *ArXiv*, abs/2205.05789, 2022.

[24] Michael Jendrusch, Jan O Korbel, and S Kashif Sadiq. Alphadesign: A de novo protein design framework based on alphafold. *Biorxiv*, pages 2021–10, 2021.

[25] Mostafa Karimi, Shaowen Zhu, Yue Cao, and Yang Shen. De novo protein design for novel folds using guided conditional wasserstein generative adversarial networks. *Journal of chemical information and modeling*, 60(12):5667–5681, 2020.

[26] Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*, 2019.

[27] Nobuyasu Koga, Rie Tatsumi-Koga, Gaohua Liu, Rong Xiao, Thomas B. Acton, Gaetano T. Montelione, and David Baker. Principles for designing ideal protein structures. *Nature*, 491:222–227, 2012.

[28] Tim Kucera, Matteo Togninalli, and Laetitia Meng-Papaxanthos. Conditional generative modeling for de novo protein design with hierarchical functions. *Bioinformatics*, 38(13):3454–3461, 2022.

[29] Brian P Lazzaro, Michael Zasloff, and Jens Rolff. Antimicrobial peptides: Application informed by evolution. *Science*, 368(6490):eaau5480, 2020.

[30] Andrew Leaver-Fay, Michael Tyka, Steven M Lewis, Oliver F Lange, James Thompson, Ron Jacak, Kristian W Kaufman, P Douglas Renfrew, Colin A Smith, Will Sheffler, et al. Rosetta3: an object-oriented software suite for the simulation and design of macromolecules. In *Methods in enzymology*, volume 487, pages 545–574. Elsevier, 2011.

[31] Julia Koehler Leman, Brian D Weitzner, Steven M Lewis, Jared Adolf-Bryfogle, Nawsad Alam, Rebecca F Alford, Melanie Aprahamian, David Baker, Kyle A Barlow, Patrick Barth, et al. Macromolecular modeling and design in rosetta: recent methods and frameworks. *Nature methods*, 17(7):665–680, 2020.

[32] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

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

[34] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[35] Lewis Moffat, Shaun M. Kandathil, and David T. Jones. Design in the dark: Learning deep generative models for de novo protein design. *bioRxiv*, 2022.

[36] Alex T Müller, Gisela Gabernet, Jan A Hiss, and Gisbert Schneider. modlamp: Python for antimicrobial peptides. *Bioinformatics*, 33(17):2753–2755, 2017.

[37] Alex T Muller, Jan A Hiss, and Gisbert Schneider. Recurrent neural network model for constructive peptide design. *Journal of chemical information and modeling*, 58(2):472–479, 2018.

[38] Deepesh Nagarajan, Tushar Nagarajan, Natasha Roy, Omkar Kulkarni, Sathyabaarathi Ravichandran, Madhulika Mishra, Dipshikha Chakravortty, and Nagasuma Chandra. Computational antimicrobial peptide design and evaluation against multidrug-resistant clinical isolates of bacteria. *Journal of Biological Chemistry*, 293(10):3492–3509, 2018.

[39] Erik Nijkamp, Jeffrey A. Ruffolo, Eli N. Weinstein, Nikhil Vijay Naik, and Ali Madani. Progen2: Exploring the boundaries of protein language models. *Cell systems*, 2022.

[40] Mi Ju Park, Seung woo Seo, Eunyoung Park, and Jinhan Kim. Epibertope: a sequence-based pre-trained bert model improves linear and structural epitope prediction by learning long-distance protein interactions effectively. *bioRxiv*, 2022.

[41] Malak Pirtskhalava, Anthony A Amstrong, Maia Grigolava, Mindia Chubinidze, Evgenia Alimbarashvili, Boris Vishnepolsky, Andrei Gabrielian, Alex Rosenthal, Darrell E Hurt, and Michael Tartakovsky. Dbaasp v3: database of antimicrobial/cytotoxic activity and structure of peptides as a resource for development of new therapeutics. *Nucleic acids research*, 49(D1):D288–D297, 2021.

[42] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[43] GN t Ramachandran and V Sasisekharan. Conformation of polypeptides and proteins. *Advances in protein chemistry*, 23:283–437, 1968.

[44] Donatas Repecka, Vykintas Jauniskis, Laurynas Karpus, Elzbieta Rembeza, Irmantas Rokaitis, Jan Zrimec, Simona Poviloniene, Audrius Laurynenas, Sandra Viknander, Wissam Abuajwa, et al. Expanding functional protein sequence spaces using generative adversarial networks. *Nature Machine Intelligence*, 3(4):324–333, 2021.

[45] Florian Richter, Andrew Leaver-Fay, Sagar D Khare, Sinisa Bjelic, and David Baker. De novo enzyme design using rosetta3. *PloS one*, 6(5):e19230, 2011.

[46] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

[47] Philip A Romero and Frances H Arnold. Exploring protein fitness landscapes by directed evolution. *Nature reviews Molecular cell biology*, 10(12):866–876, 2009.

[48] Allison M. Rossetto and Wenjin Zhou. Gandalf: A prototype of a gan-based peptide design method. *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 2019.

[49] Marion F Sauer, Alexander M Sevy, James E Crowe Jr, and Jens Meiler. Multi-state design of flexible proteins predicts sequences optimal for conformational change. *PLOS Computational Biology*, 16(2):e1007339, 2020.

[50] Josep M Sierra, Ester Fusté, Francesc Rabanal, Teresa Vinuesa, and Miguel Viñas. An overview of antimicrobial peptides and the latest advances in their development. *Expert opinion on biological therapy*, 17(6):663–676, 2017.

[51] Paulina Szymczak, Marcin Możejko, Tomasz Grzegorzek, Radosław Jurczak, Marta Bauer, Damian Neubauer, Karol Sikora, Michał Michalski, Jacek Sroka, Piotr Setny, et al. Discovering highly potent antimicrobial peptides with deep generative model hydramp. *Nature Communications*, 14(1):1453, 2023.

[52] Wending Tang, Ruyu Dai, Wenhui Yan, Wei Zhang, Yannan Bin, Enhua Xia, and Junfeng Xia. Identifying multi-functional bioactive peptide functions using multi-label deep learning. *Briefings in Bioinformatics*, 23(1):bbab414, 2022.

[53] Faiza Hanif Waghu, Lijin Gopi, Ram Shankar Barai, Pranay Ramteke, Bilal Nizami, and Susan Idicula-Thomas. Camp: Collection of sequences and structures of antimicrobial peptides. *Nucleic acids research*, 42(D1):D1154–D1158, 2014.

[54] Michael J Wedemeyer, Benjamin K Mueller, Brian J Bender, Jens Meiler, and Brian F Volkman. Modeling the complete chemokine–receptor interaction. In *Methods in cell biology*, volume 149, pages 289–314. Elsevier, 2019.

[55] Zachary Wu, Kevin K Yang, Michael J Liszka, Alycia Lee, Alina Batzilla, David Wernick, David P Weiner, and Frances H Arnold. Signal peptides generated by attention-based neural networks. *ACS Synthetic Biology*, 9(8):2154–2161, 2020.

[56] Yang Zhang and Jeffrey Skolnick. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, 33(7):2302–2309, 01 2005.

Supplementary Files

Figures

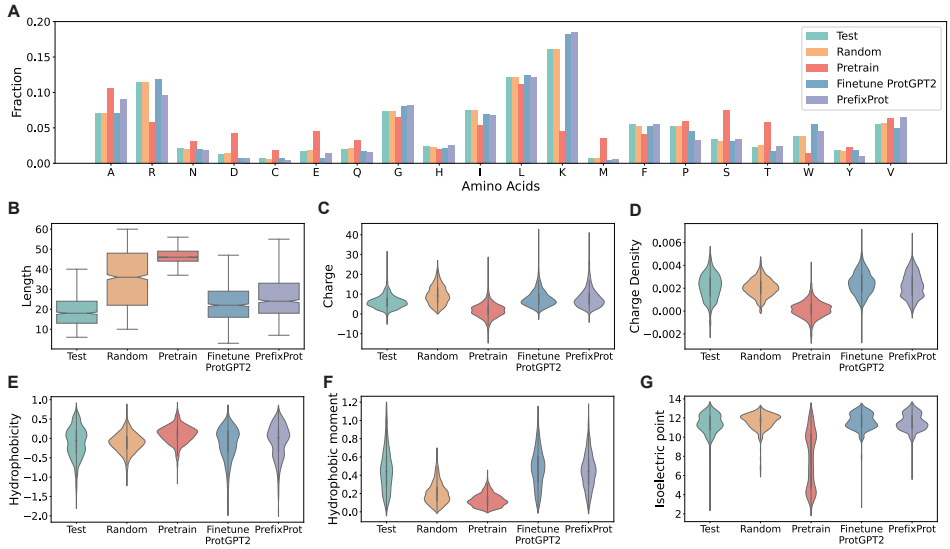

Figure S1: Physicochemical property comparison in generating proteins with antimicrobial function.

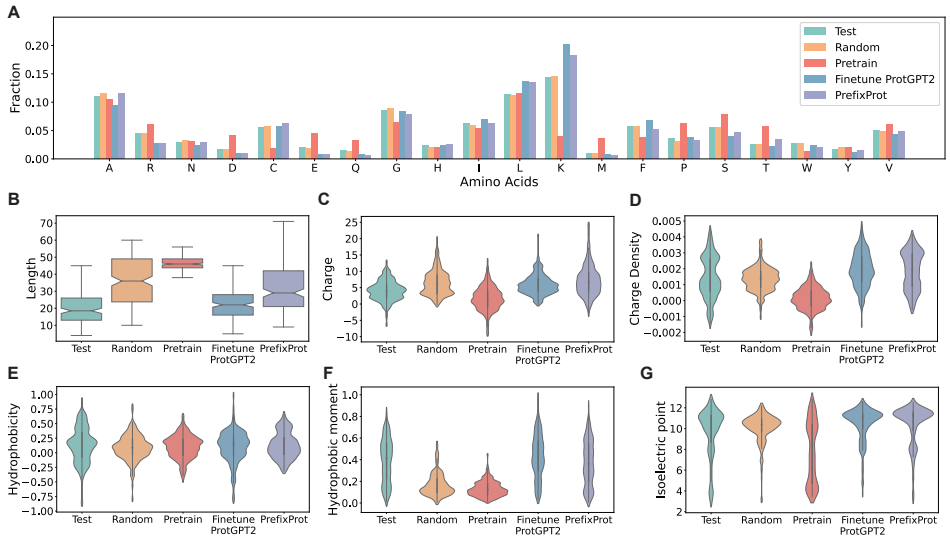

Figure S2: Physicochemical property comparison in generating proteins with anticancer function.

Tables

Table S1: The settings of STREME for finding the motif of the generated candidate multi-functional peptide, and the peptide with both antibacterial and anticancer functions.

Parameter	value
Minimum width	3
Maximum width	8
p value threshold	0.005
Number of motifs	10
Align sequences on their	Centers