A deep unsupervised language model for protein design

Noelia Ferruz[1,*], Steffen Schmidt[2], Birte Höcker[1]

[1]Department of Biochemistry, University of Bayreuth, Bayreuth, Germany. [2]Computational Biochemistry. University of Bayreuth, 95447 Bayreuth, Germany.

Email: noelia.ferruz-capapey@uni-bayreuth.de

Keywords: protein design, language model, natural language processing, transformer.

Abstract

Protein design aims to build new proteins from scratch thereby holding the potential to tackle many environmental and biomedical problems. Recent progress in the field of natural language processing (NLP) has enabled the implementation of ever-growing language models capable of understanding and generating text with human-like capabilities. Given the many similarities between human languages and protein sequences, the use of NLP models offers itself for predictive tasks in protein research. Motivated by the evident success of generative Transformer-based language models such as the GPT-x series, we developed ProtGPT2, a language model trained on protein space that generates *de novo* protein sequences that follow the principles of natural ones. In particular, the generated proteins display amino acid propensities which resemble natural proteins. Disorder and secondary structure prediction indicate that 88% of ProtGPT2-generated proteins are globular, in line with natural sequences. Sensitive sequence searches in protein databases show that ProtGPT2 sequences are distantly related to natural ones, and similarity networks further demonstrate that ProtGPT2 is sampling unexplored regions of protein space. AlphaFold prediction of ProtGPT2-sequences yielded well-folded non-idealized structures with embodiments as well as large loops and revealed new topologies not captured in current structure databases. ProtGPT2 has learned to speak the protein language. It has the potential to generate *de novo* proteins in a high throughput fashion in a matter of seconds. The model is easy-to-use and freely available.

Introduction

Natural Language Processing (NLP) has seen extraordinary advances in recent years. Large pre-trained language models have drastically transformed the NLP field and with it many of the tools we use in our daily lives, such as chatbots, smart assistants, or translation machines. Analogies between protein sequences and human languages have long been noted by us and others (1, 2). Protein sequences can be described as a concatenation of letters from a chemically defined alphabet, the natural amino acids. These letters arrange to form secondary structural elements ('words'), which assemble to form domains ('sentences') that undertake a function ('meaning'). Although protein sequences and human languages are not without dissimilarities, their analogies have stimulated the application of NLP methods to solve protein research problems for decades. One of the most attractive similarities is that protein sequences, like natural languages, are information-complete: they store structure and function entirely in their amino acid order with extreme efficiency. With the extraordinary advances in the NLP field in understanding and generating language with near-human capabilities, we hypothesized that these methods might help us understand and '*speak fluently*' the protein language. Indeed, a few pioneering works have already implemented protein language models capable of capturing the diversity of sequences in the protein space, producing internal representations that encode properties transcending the sequence level, like structure and function (3–6).

While these innovative applications show the emerging potential of language models leveraging the growing sequence data, they often produce a representation vector of a given input

sequence that can be coupled to downstream tasks. This fact is owed to their specific model architecture and masked modelling training objective. However, one of the most powerful properties of language models is that they can be autoregressive, i.e., they can be trained to predict subsequent words given a context accurately. These models, from where the most well-known are possibly the GPT-x series, excel at generating long, coherent text – sometimes to the extent that much debate has been raised about their potential misuse (7).

Generative language models provide excellent opportunities for the protein design field. Since proteins are the basic machinery in all living organisms, their design has the potential to tackle many industrial and biomedical problems. Unfortunately, protein design is often a lengthy process requiring costly computational and experimental characterizations. A fast and reliable way to generate new protein sequences would be desirable. As a step in this direction, we introduce ProtGPT2, an autoregressive Transformer model with 738 million parameters capable of generating *de novo* protein sequences in a high-throughput fashion. ProtGPT2 has effectively learned the protein language upon being trained on about 50 million sequences spanning the entire protein space. ProtGPT2 generates protein sequences with amino acid and disorder propensities on par with natural ones while being "evolutionarily" distant from the current protein space. Secondary structure prediction calculates 88% of the sequences to be globular, in line with natural proteins. Representation of the protein space using similarity networks reveals that ProtGPT2 sequences explore 'dark' areas of the protein space, expanding natural superfamilies. Since ProtGPT2 has been already pre-trained, it can be used to generate sequences on standard workstations in a matter of seconds or be further finetuned on sequence sets of a user's choice to augment specific protein families. The model and datasets are available in the HuggingFace repository (8) at (https://huggingface.co/nferruz/ProtGPT2). We believe that ProtGPT2 poses a significant step towards efficient high-throughput protein engineering and design.

Results

Learning the protein language

The major advances in the NLP field can be partially attributed to the scale-up of unsupervised language models. Unlike supervised learning, which requires the labelling of each data point, self-supervised (or often named unsupervised) methods do not require annotated data, promoting the use of ever-growing datasets, such as Wikipedia or the C4 Corpus (9). Given both the growth of protein sequence databases and the lack of annotation for a significant part of the protein space, protein sequences have become great candidates for unsupervised training (3, 4, 10) and now offer the opportunity to learn and *speak* the protein language.

To achieve this goal, we trained a Transformer (11) to produce a model that generates protein sequences. Language models are statistical models that assign probabilities to words and sentences. We are interested in the model that assigns high probability to sentences (W) that are semantically and syntactically correct or fit and functional in the case of proteins. Besides, because we are interested in a generative language model, we trained the model using an autoregressive strategy. In autoregressive models, the probability of a particular token or word (w_i) in a sequence depends solely on its context, namely the previous tokens in the sequence. The total probability of a sentence (W) is the combination of the individual probabilities for each word (w_i):

$$p\,(W) = \prod_i^n p(w_i|w_{<i}) \quad (1)$$

We trained the Transformer by minimising the negative log-likelihood over the entire dataset. More intuitively, the model must learn the relationships between a word w_i - or amino acid - and all the previous ones in the sequence, and must do so for each sequence *k* in the dataset (D):

$$\mathcal{L}_{CLM} = -\sum_{k=1}^{D} logp\left(w_i^k \middle| w_{<i}^k\right) \quad (2)$$

To learn the protein language, we used UniRef50 (UR50) (version 2021_04), a clustering of UniProt at 50% identity. We chose this dataset versus larger versions of UniParc (such as UR90 or UR100) as it was previously shown to improve generalization and performance for the ESM

Transformers (3). Uniref50's sequences populate the entire protein space, including the dark proteome, regions of the protein space whose structure is not accessible via experimental methods or homology modelling (12, 13). For evaluation, we randomly excluded 10% of the dataset sequences – these sequences are not seen by ProtGPT2 during the training process. The final training datasets contained 44.9 and 4.9 million sequences for training and evaluation, respectively (**Methods**). We tokenized our dataset using the BPE algorithm (14) (**Methods**). The final model is a decoder-only architecture of 36 layers and 738 million parameters.

Analogous to the GLUE benchmark (15) - a collection of tools that computational linguists use to evaluate language models on different tasks such as question answering or translation - we also developed a series of extrinsic tests to assess the quality of ProtGPT2 generated sequences. The following sections elaborate on how ProtGPT2 generates *de novo* sequences with properties that resemble modern protein space.

Statistical sampling of natural amino acid propensities

Autoregressive language generation is based on the assumption that the probability distribution of a sequence can be decomposed into the product of conditional next-word distributions (eq. 1). However, there is still considerable debate about the best decoding strategy to emit sequences from a model (16). It is not uncommon that well-trained generic language models that perform well in GLUE tasks generate incoherent gibberish or repetitive text depending on the sampling procedure (16). We briefly summarize here an overview of the most used sampling strategies for language generation that we applied in this study.

Greedy search selects the word with the highest probability at each timestep. Although algorithmically simple, the generated sequences are deterministic and soon also become repetitive (**Fig. 1a**). Beam search tries to alleviate this problem by retaining the most probable candidates, although the resulting texts still suffer from repetitiveness and are not as surprising as those from humans, which tend to alternate low and high probability tokens (16) (**Fig. 1b**). Lastly, random sampling moves away from deterministic sampling by randomly picking a word out of the top-k most probable ones (**Fig. 1c**).

In a recent study, Holtzman et al. (16) investigated several sampling strategies to find the best parameters for text generation. Inspired by this work, we systematically generated sequences following different sampling strategies and parameters (**Fig. 1, Methods**). To assess what sampling procedure generates the most natural-like sequences, we compared the amino acid propensities of the generated set to that found in natural protein sequences. As stated by Hoffmann et al., we also observe greedy and beam search to produce repetitive, deterministic sequences, while random sampling dramatically improves the generated propensities (**Fig. 1**). Moreover, we also observe that high values of k are needed to generate sequences that resemble natural ones, i.e., our best results occur in the range of k>800 and we specifically chose k=950 in this work (**Fig. 1h**). As observed with other generative models (5, 17), our sampling improves when applying a repetition penalty of 1.2 (**Methods**). Consequently, we used these sampling parameters for the rest of this work.

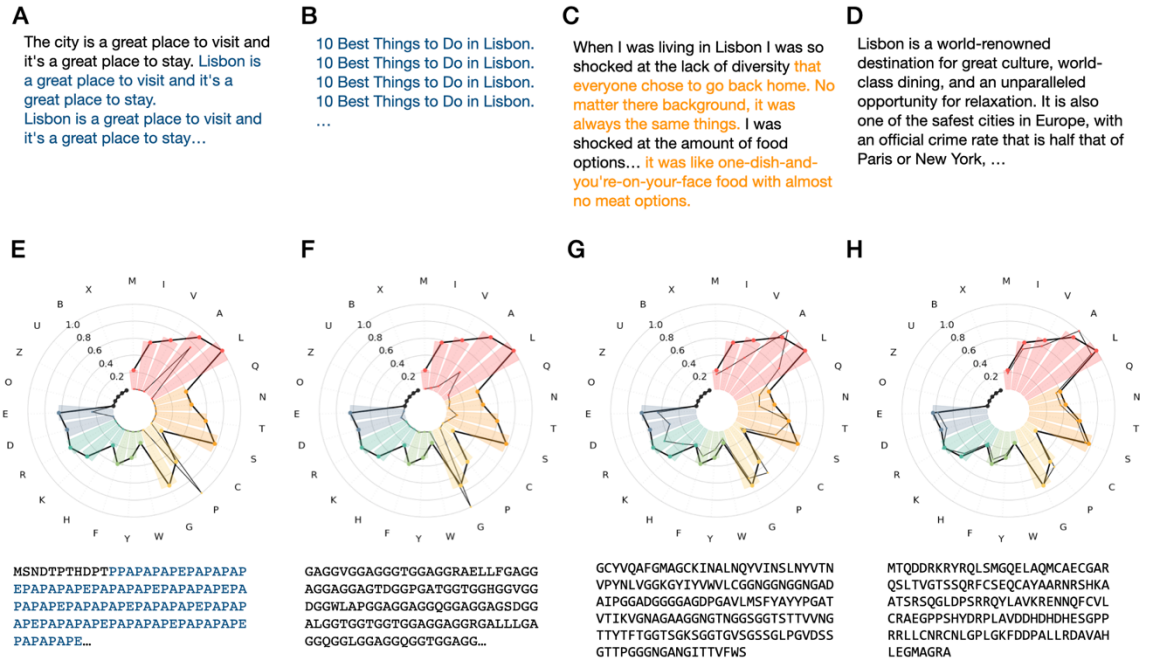

Figure 1: Examples with different sampling parameters for GPT2-large after the context input "*10 best things to do in Lisbon*" (A-D) and ProtGPT2 without context (E-H). While greedy and beam search produce repetitive sentences (A, B) and protein sequences (E, F), sampling generates creative texts, which however, can be degenerate (C) or not sample natural sequence propensities (G) for small values of k. Larger values of k produce quality text (D) and sequences whose propensities match natural ones. Repetitive and degenerative text are shown in blue and orange, respectively.

ProtGPT2 sequences encode globular proteins

In order to evaluate ProtGPT2's generated sequences in the context of sequence and structural properties, we created two datasets, one with sequences generated from ProtGPT2 using the previously described inference parameters and the other with randomly chosen sequences from UR50. Each dataset consists of 10,000 sequences.

We studied disordered and secondary structural content in the datasets. It has been previously shown that around 14% of the proteins found in bacteria and archaea are disordered (12). To this end we ran IUPred3 (18) to analyze if the ProtGPT2 generated sequences are more prone to be disordered than a set of natural sequences. Interestingly, IUPred3 predicts a similar number of structured among the ProtGPT2 generated sequences (87.59%) and natural sequences (88.40%). A detailed analysis of the protein sequences at the amino acid level also shows a similar distribution of ordered/disordered regions for the natural and the ProtGPT2 dataset (**Table 1**).

We also investigated whether the similarities in disorder are a consequence of equivalent secondary structure element content. To this end, we computed PSIPRED (19) predictions for the ProtGPT2 and natural sequence datasets (**Methods**). The natural sequences display alpha helical, coil and beta sheet contents of 45.19%, 41.87% and 12.93%, respectively. The ProtGPT2 dataset presented percentages of 48.64%, 39.70%, and 11.66%, respectively.

These results indicate that ProtGPT2 generates sequences that resemble globular domains whose secondary structure contents are in line with those in the natural space.

Table 1: Summary of results for disorder and secondary structure prediction.

	ProtGPT2 dataset	Natural dataset
IUPred3 (globular domains)	87.59%	88.40%
Ordered content	82.59%	79.71%
Alpha helical content	48.64%	45.19%
Beta sheet content	39.70%	41.87%
Coil content	11.66%	12.93%

ProtGPT2 sequences are similar yet distant to natural ones

Proteins have diversified immensely in the course of evolution via point mutations as well as duplication and recombination. Using sequence comparisons, it is however possible to detect similarities between two proteins even when their sequences have significantly diverged. We wondered how related ProtGPT2 sequences are to natural ones. To this end, we utilized HHblits, a sensitive remote homology detection tool that uses profile hidden Markov models to search query sequences against a database (20). We searched for homologues of the 10,000 sequences in ProtGPT2's dataset against the Uniclust30 database (21). For comparison purposes, we also performed the same search with the natural dataset using the same settings. In addition, to analyze how completely random sequences would compare against ProtGPT2 ones, we also crafted a third dataset by randomly concatenating the 25 letters in the vocabulary (**Methods**).

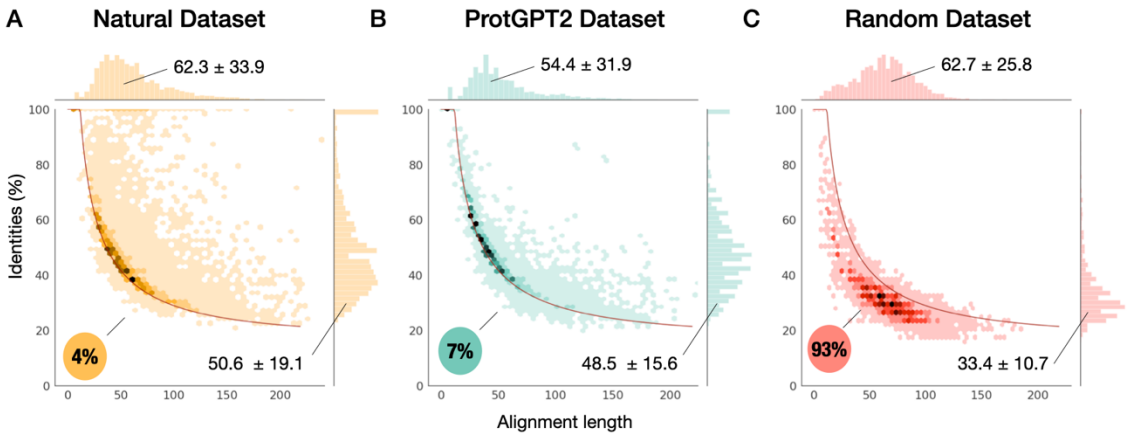

Figure 2: **Pairwise sequence identities vs. alignment length for each of the datasets (A: natural, B: ProtGPT2, and C: random) as computed with HHblits against the Uniclust30 database**. The lines depicted in red on each plot represent the HSSP curve, which we use as reference to compare the three datasets (22). Each plot shows a *hexbin* compartmentalization of the best scoring identities and their distributions. While natural (**A**) and protGPT2 (**B**) sequences show similar percentages below the curve, 93% of the sequences in the random dataset (**C**) do not have significantly similar sequences in the Uniclust30 database. Natural and ProtGPT2 datasets show significant differences in the high-identity range.

Because we want to provide a quantitative comparison of the datasets' relatedness to modern protein space, we produced identity vs sequence length plots (**Fig. 2**). In detail, for each of the alignments found in Uniclust30, we depict the one with the highest identity and length (**Methods**). As a reference point in this sequence identity-length space, we use the HSSP curve (22), a boundary set to define the confidence of protein sequences relatedness. Proteins whose identity falls below this curve, an area known as 'twilight zone', do not necessarily have similar 3D structures nor are likely homologous. Since the sequences in the ProtGPT2 and random datasets are not the consequence of protein evolution, we use the curve as a well-known threshold to compare the datasets.

When looking at the distribution of hits above and below the curve, we observe that HHblits finds many hits in the Uniclust30 database that are related to the dataset of natural sequences (**Fig. 2a**). Specifically, out of the 10,000 dataset sequences, 9,621 (96.2%) showed identities above the

HSSP curve. Similarly, 9,295 of ProtGPT2 generated sequences (93%) also have counterparts in the Uniclust30 database that align above the HSSP curve (**Fig. 2b**). Conversely, 93% of the randomly generated sequences fall below this threshold (**Fig. 2c**). Despite these similar patterns for the natural and ProtGPT2 datasets, the two datasets show differences in their distribution of hits. With a one-standard-deviation range of 31.6 – 69.7 %, the natural dataset has a higher mean identity than the ProtGPT2 set, with a range of 32.85 – 64.1 % (**Fig. 2a, b**). The differences between the natural and ProtGPT2 sequence distributions are not statistically significant (p-value < 0.05 Kolmogorov-Smirnoff). However, substantial differences between the natural and ProtGPT2 datasets occur in the high-identity range (> 90 %). Although 365 sequences in the ProtGPT2 dataset have high-identity sequences in Uniclust30, they correspond in all cases to alignments below 15 amino acids, whereas the natural dataset displays 760 sequences over 90% with an alignment length in the one-standard-deviation range of 14.8 – 77.3 amino acids. These results suggest that ProtGPT2 effectively generates sequences that are distantly related to natural ones but are not a consequence of memorization and repetition.

ProtGPT2 generates ordered structures

One of the most important features when designing *de novo* sequences is that they fold into stable ordered structures. We have predicted the structures of the 10,000 ProtGPT2 sequences using Alphafold (23, 24). AlphaFold produces a per-residue estimate of its confidence on a scale from 0-100 (pLDDT). This score has been shown to correlate with order (25): low scores (pLDDT > 50) tend to appear in disordered regions, while excellent scores (pLDDT > 90) appear in ordered ones (25). Here we produced five structure predictions per sequence. The mean pLDDT of the dataset is 63.2 when taking the best-scoring structure per sequence and 59.6 when averaging across all five predictions per sequence. Besides, 32% of sequences present pLDDT values over 70, agreeing with other recent studies (6).

ProtGPT2 transcends the boundaries of the current protein space

Several publications tried to reduce the large dimensionality of protein sequences into a few discernible dimensions for their analysis. Most representation methods consist of (i) hierarchical classifications of protein structures such as the ECOD and CATH databases (26, 27), (ii) Cartesian representations (28), and similarity networks (29, 30). We recently represented the structural space in a network that showed proteins as nodes, linked when they have a homologous and structurally-similar fragment in common (31) and made the results available in the Fuzzle database (32). The network represented 25,000 domains from the seven major SCOP classes and showed that the modern known protein space has both connected and 'island-like' regions.

It is implausible that evolution has explored all possible protein sequences (33). Therefore, the challenge has been posed whether we can design proteins that populate unexplored - or dark - regions of the protein space and if, by doing so, we can design novel topologies and functions (33). Here, we integrated the ProtGPT2 sequences into our network representation of the protein space. To this end, we generated an HMM profile for each SCOPe2.07 and ProtGPT2 sequence, compared them in an all-against-all fashion using HHsearch and represented the networks with Protlego (34) (**Methods**). To avoid that specific sequences with several alignments end up represented by the same node in the network, we duplicate entries with two non-overlapping alignments, as previously described (31).

The network contains 59,612 vertices and 427,378 edges, comprising 1,847 components or 'island-like' clusters (**Fig. 3**). The major component accumulates more than half of the nodes (30,690) – a number significantly higher than the number observed in a network produced with the same settings but excluding ProtGPT2 sequences (**Fig. S2**) – strongly suggesting that ProtGPT2 generates sequences that bridge separate islands in protein space. We select six examples across different areas of the network from topological different SCOPe classes to showcase ProtGPT2 sequences at the structural level (**Fig. 3**). In particular, we report an all-β (**751**), two α/β (**4266**, **1068**), one membrane protein (**4307**), an α + β (**486**) and all-α (**785**) structures. These structures attempt to illustrate ProtGPT2 versatility at generating *de novo* structures. For each case, we searched the most similar protein structure found in the PDB database using FoldSeek (35).

ProtGPT2 generates well-folded all-β structures (**751**, **4307**), which despite recent impressive advances (36), have for long remained very challenging (37). ProtGPT2 also produces membrane proteins (**4307**), which pose a difficult target for protein design due to the challenges at specifying structure within the membrane and the laborious experimental characterizations (38). Besides the generation of natural fold representatives, ProtGPT2 also produces novel topologies. For example, we report protein **4266**, whose topology does not match any of the currently reported structures in the PDB, with a low DALI Z-score of 5.4 and an RMSD of 3.0 Å to PDB 5B48 over 67 residues (identity 9%).

Nevertheless, possibly the most remarkable properties of ProtGPT2 sequences is that they significantly deviate from all previously designed *de novo* structures, which often feature idealized topologies with loops and minimal structural elements. De *novo* proteins have the advantage of not carrying any evolutionary history and are thus amenable as a scaffold for virtually any function, but in practice, the lack of embodiments and longer loops hamper the design of crevices, surfaces and cavities - necessary for the interaction with other molecules and function realization. ProtGPT2 sequences resemble the complexity of natural proteins, with multifaceted surfaces capable of allocating interacting molecules and substrates, thus paving the way for functionalization. In **Figure 3**, we show structures **486** and **1060**, two examples of such complex structures**.** In particular, **1068** shows a TIM-barrel fold, a topology which to date has met impressive success in *de novo* design (39–41), but whose idealized structure has nevertheless proven challenging to extend via additional secondary elements and longer loops (42, 43).

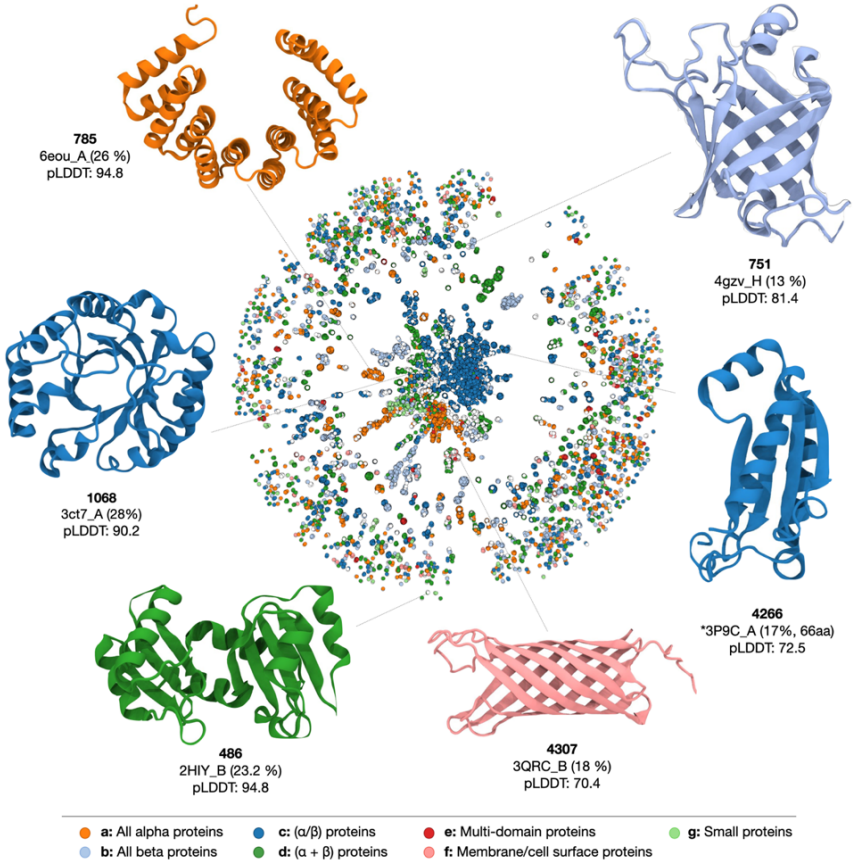

Figure 3: **An overview of the protein space and examples of proteins generated by ProtGPT2.** Each node represents a sequence. Two nodes are linked when they have an alignment of at least 20 amino acids and 70 % HHsearch probability. Colours depict the different SCOPe classes, and ProtGPT2 sequences are shown in white. As examples, we select proteins of each of the major 5 SCOP classes: all-β structures (751), α/β (4266, 1068), membrane protein (4307), α+β (486), and all-α (785). The selected structures are colored according to the class of their most similar hit. The structures were predicted with AlphaFold, and we indicate the code of the most similar structure in the PDB as found by FoldSeek (35), except for protein 4266, where no structures were found.

Preserved Functional Hotspots

Visual inspection of the structural superimposition of the best hits found with FoldSeek revealed several instances where the sidechains of ligand-interacting residues are conserved. We here show two examples (**Fig. 4**). The most similar structure of sequence **357** (**Fig. 4a**) corresponds to PDB code 1X0P (chain A), a blue-light sensor domain that binds FAD. When superimposing the structures, we observe that **357** has retained the sidechain binding hotspots, with three residues identical (D169, Q150, and N131) and two different but capable of forming the same interactions, Lysine at position R165 and Histidine at position K127. Sequence 475 is most similar to PDB code 5M1T (chain A), a phosphodiesterase that folds into a TIM barrel and binds to the bacterial second messenger cyclic di-3′,5′-guanosine monophosphate (PDB three-letter code C2E). Out of the five sidechain interacting residues, the ProtGPT2 sequence preserves three residues (Q455, R473, and E469), and includes one substitution for another residue capable of hydrogen-bonding (aspartic acid for Q513). It is remarkable to note that ProtGPT2 has generated these sequences in a zero-shot fashion, i.e., without further finetuning in these two particular folds. These results have impactful consequences for protein engineering because ProtGPT2 appears to preserve binding positions in the generated sequences, despite the low identities (31.1% and 29.2% for 357 and 45, respectively) and can be used to augment the repertoires of specific folds and families.

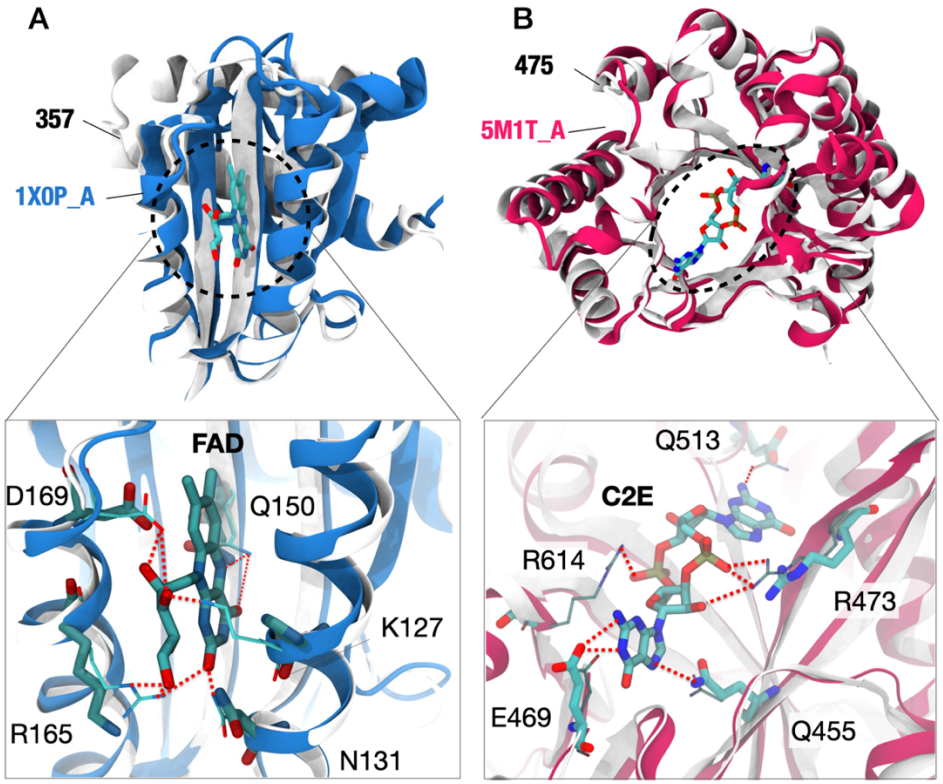

Figure 4: Superimposition of the predicted structures for sequences 357 and 475 and the respective top scoring proteins in FoldSeek. (A) Structural alignment of 357 with pdb 1X0P (chain A, blue). Shown are five residues in 1X0P that interact via their sidechains with the ligand FAD. Of these, 3 are identical in **357**, and another two correspond to substitutions to the same amino acid type (R165 to lysine and Q150 to histidine). **(B)** Structural alignment of **475** with pdb 5M1T (chain A) depicting five sidechain-interacting residues with ligand C2E. All amino acids in **475** are conserved except for residue R614 which was substituted by a glycine. The PDB structures are shown in color with their sidechains in a thinner representation.

Discussion

The design of *de novo* proteins harnessing artificial intelligence methods is meeting incredible success in the last two years (3, 24). Motivated by the unprecedented advances in NLP, we have implemented a generative language model, ProtGPT2, which has effectively learned and 'speaks' the protein language. ProtGPT2 can generate sequences that are distantly related to natural ones and whose structures resemble the known structural space, with non-idealized complex structures. Since ProtGPT2 has been trained on the entire sequence space, the sequences produced by the model can sample any region, including the dark proteome and areas traditionally regarded as very challenging in the protein design field, such as all-β structures and membrane proteins. Visual superimposition of ProtGPT2 proteins with distantly related natural protein structures reveals that ProtGPT2 has also captured some functional determinants, preserving ligand-binding interactions. As the design of novel proteins can solve many biomedical and environmental problems, we see extraordinary potential in our protein language model. ProtGPT2 designs fit globular proteins in a matter of seconds without requiring further training on a standard workstation. ProtGPT2 can also be conditioned towards a particular family, function or fold by fine-tuning the model on a set of sequences of a user's choice. Thus, ProtGPT2 constitutes a big step forward towards efficient protein generation. We envision future efforts towards the inclusion of conditional tags, which will enable the controlled generation of specific functions.

Materials and Methods

Vocabulary Encoding

Given the success of the Byte Pair Encoding (BPE) algorithm to tokenize sequences in previous generative models, we use a BPE (44) tokenizer to train the vocabulary of our dataset. BPE is a sub-word tokenization algorithm that finds the most frequently used word roots, ensuring better performance than one-hot tokenization and avoiding the out-of-vocabulary problem. Given the size of Uniref50, we used Swiss-Prot (2021_04) containing > 0.5 M sequences to train our tokenizer. Following the training strategy followed when training GPT2, (45) our final vocabulary contained 50,256 tokens that correspond to the most widely reused oligomers in protein space, with an average size of 4 amino acids per token (**Fig. S1**).

Dataset preparation

We took Uniref50 version 2021_04 as the dataset for training, containing 49,874,565 sequences. 10% of the sequences were randomly selected to produce the validation dataset. The final training and validation datasets contained 44.88 and 4.99 million sequences, respectively. We produced two datasets, one using a block size of 512 tokens, and another one with 1024 tokens. Results shown in this work correspond to a model train with a block size of 512 tokens. The training dataset has been released to HuggingFace at (https://huggingface.co/nferruz/ProtGPT2) (8).

Model pre-training

We use a Transformer decoder model as architecture for our training which processes input sequences tokenized with a BPE strategy. The model uses during training the original dot-scale self-attention as introduced by Vaswani et al. (11). All our models consist of 36 layers with a model dimensionality of 1280. The architecture matches that of the previously released GPT2-large Transformer (45) which was downloaded from HuggingFace (8). Model weights were reinitialized prior training. The model was optimized using Adam (β_1 = 0.9, β_2 = 0.999) with a learning rate of 1e-03. For our main model, we trained 65,536 tokens per batch (128 GPUs x 512 tokens). A batch size of 8 per device was used totaling 1024. The model trained on 128 NVIDIA A100s in 4 days. Parallelism of the model was handled with DeepSpeed (46).

Model Inference

We systematically sampled sequences using our main model using different inference parameters. In particular, we varied repetition penalty from a range of 1.1 to 3.0 at each 0.1 units, top_k from 250 to 1000 sampling every 50 units, and a top_p from 0.7 to 1.0 with a window of 0.05 units. 100 sequences were produced for each sampling parameter set and the frequency of their amino acids compared to natural sequences. We observed which parameters produced fewer differences to the set of seven most common amino acids in natural sequences. We also explored the beam search algorithm for beams in the range 50 to 100 using a window of 1 unit but it produced worse matches in all cases. To determine amino acid frequencies in natural sequences for comparison to ProtGPT2 samples we randomly picked 1 million sequences from the Uniref50 dataset. The best matching parameters were further downsampled with finer windows and their frequencies compared with radar plots such shown in **Fig. 1** in the main text. Best performing parameters in our dataset were top_k 950, repetition penalty of 1.2 and default temperature and top_p values of 1.

Model and Dataset availability

The model is freely available at https://huggingface.co/nferruz/ProtGPT2. The datasets are available at https://huggingface.co/datasets/nferruz/UR50_2021_04.

Sequence dataset generation

Three sequence datasets were produced to compare their properties. The ProtGPT2 dataset was generated by sampling 1000 batches of 100 sequences each with the selected inference parameters and a window context of 250 tokens. This step produced 100,000 sequences. We filtered from this set those sequences whose length had been cut due to the window context, giving

a total of 29,876 sequences. From this set, we randomly selected 10,000 sequences. Their average length is 149.2 ± 50.9 amino acids. The natural dataset was created by randomly sampling 100,000 sequences from Uniref50. 10,000 of these sequences were further chosen to ensure their average and standard deviation lengths matched that of the ProtGPT2 dataset sequences. The random dataset was created by concatenating the 25 amino acids that appear in UniRef50, which includes the 20 standard amino acids and other IUPAC codes such as 'X', 'B', 'U', 'O', and 'Z', by randomly concatenating them into sequences with a length taken from a normal distribution between 5 and 267 amino acids.

Homology detection

Each sequence in the three 10k datasets was searched for similarity against the PDB70 and uniclust30 databases using HHblits (47). We used the Uniclust30 database version 2018_08 and the pdb70 version 2021_04. As HHblits produces a list of alignments we selected all those over the HSSP curve as possible matches, and from these selected the largest alignment. Thus, for each sequence in each dataset the longest and the highest identity scoring alignment was selected and represented in **Fig. 2**.

Disorder prediction

IUPred3 was run on both datasets using all three possible options to detect shorter ('short') or longer ('longer') unstructured regions, as well as structured regions ('glob'). Ordered content was determined with the 'short' option. The output of the 'glob' analysis also reports if any structured, globular domain was found as show in Table 1. We run secondary structure prediction using PSIPRED v4.0 for each sequence in natural and ProtGPT2 datasets. The alignments of the above mentioned HHblits searches were used as multiple sequence alignments. We compute the percentages for each secondary element by dividing the number of amino acids with a certain prediction by the total number of amino acids with a confidence value of 5 or more.

AlphaFold2 structure prediction

We predicted 5 structures for each sequence in the ProtGPT2 dataset using AlphaFold ColabFold batch (23).

Network construction

Sequences in the ProtGPT2 and SCOP 2.07 filtered at 95% datasets were joined. For each sequence we produced a multiple sequence alignment (MSA) using HHblits against database Uniclust 2018_08. Hidden Markov model profiles were produced for each MSA using HHblits (47), and an all-against-all search for each profile was performed using HHsearch (20). The network was constructed by representing every sequence as a node, and linking two nodes whenever they have an alignment of at least 20 amino acids with 70% HHsearch probability. Extensive details on the all-against-all comparison and network construction can be found in our previous works (32, 48). Detection of similar topologies was determined with FoldSeek (35).

Acknowledgments

We thank the generous computational time provided by the High-Performance Computing Center at the Friedrich-Alexander-University Erlangen-Nuremberg and Thomas Zeiser for his considerate support. We thank Surbhi Dhingra for feedback on the manuscript.

References

1. K. K. Yang, Z. Wu, F. H. Arnold, Machine-learning-guided directed evolution for protein engineering. *Nat. Methods* **16**, 687–694 (2019).
2. N. Ferruz, B. Höcker, Towards Controllable Protein design with Conditional Transformers *arXiv Prepr. arXiv 2201.07338* (2022).
3. A. Rives, *et al.*, Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci.* **118** (2021).
4. E. Alley, G. Khimulya, S. Biswas, M. AlQuraishi, G. Church, Unified rational protein engineering with sequence-only deep representation learning. *bioRxiv*, 589333 (2019).
5. A. Madani, *et al.*, ProGen: Language Modeling for Protein Generation. *bioRxiv*, 2020.03.07.982272 (2020).
6. L. Moffat, S. M. Kandathil, D. T. Jones, Design in the DARK: Learning Deep Generative Models for De Novo Protein Design. *bioRxiv*, 2022.01.27.478087 (2022).
7. Alex Hern; New AI fake text generator may be too dangerous to release, say creators. The Guardian, 2019.
8. T. Wolf, *et al.*, HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv Prepr. arXiv1910.03771* (2019).
9. C. Raffel, *et al.*, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.* **21**, 1–67 (2020).
10. A. Elnaggar, *et al.*, ProtTrans: Towards Cracking the Language of Life's Code Through Self-Supervised Learning. *bioRxiv*, 2020.07.12.199554 (2021).
11. A. Vaswani, *et al.*, Transformer: Attention is all you need in *Advances in Neural Information Processing Systems*, (2017), pp. 5999–6009.
12. N. Perdigão, *et al.*, Unexpected features of the dark proteome. *Proc. Natl. Acad. Sci. U. S. A.* **112**, 15898–15903 (2015).
13. N. Perdigão, A. C. Rosa, S. I. O'Donoghue, The Dark Proteome Database. *BioData Min.* **10** (2017).
14. P. Gage, A New Algorithm for Data Compression. C Users J 12, 23-38. (1994) https:/doi.org/10.5555/177910.177914 (January 26, 2021).
15. A. Wang, *et al.*, GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. *arXiv Prepr. arXiv1804.07461* (2018).
16. A. Holtzman, J. Buys, L. Du, M. Forbes, Y. Choi, The Curious Case of Neural Text Degeneration. *CEUR Workshop Proc.* **2540** (2019).
17. N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, R. Socher, CTRL: A Conditional Transformer Language Model for Controllable Generation. *arXiv Prepr. arXiv 1909.05858* (2019).
18. G. ´ Abor Erd, ˝ Os, M. ´ Atyásaty´atyás Pajkos, Z. Dosztányi, D. Dosztányi, IUPred3: prediction of protein disorder enhanced with unambiguous experimental annotation and visualization of evolutionary conservation. *Nucleic Acids Res.* **49**, W297–W303 (2021).
19. D. W. A. Buchan, D. T. Jones, The PSIPRED Protein Analysis Workbench: 20 years on. *Nucleic Acids Res.* **47**, W402–W407 (2019).
20. J. Söding, Protein homology detection by HMM-HMM comparison. *Bioinformatics* **21**, 951–960 (2005).
21. M. Mirdita, *et al.*, Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res.* **45**, D170 (2017).
22. B. Rost, Twilight zone of protein sequence alignments. *Protein Eng. Des. Sel.* **12**, 85–94 (1999).
23. M. Mirdita, S. Ovchinnikov, M. Steinegger, ColabFold - Making protein folding accessible to all. *bioRxiv*, 2021.08.15.456425 (2021).
24. J. Jumper, *et al.*, Highly accurate protein structure prediction with AlphaFold. *Nat. 2021 5967873* **596**, 583–589 (2021).
25. K. Tunyasuvunakool, *et al.*, Highly accurate protein structure prediction for the human proteome. *Nat. 2021 5967873* **596**, 590–596 (2021).
26. H. Cheng, *et al.*, ECOD: An Evolutionary Classification of Protein Domains. *PLOS Comput.*

Biol. **10**, e1003926 (2014).
27. I. Sillitoe, *et al.*, CATH: Increased structural coverage of functional space. *Nucleic Acids Res.* **49**, D266–D273 (2021).
28. M. Osadchy, R. Kolodny, Maps of protein structure space reveal a fundamental relationship between protein structure and function. *Proc. Natl. Acad. Sci. U. S. A.* **108**, 12301–12306 (2011).
29. V. Alva, M. Remmert, A. Biegert, A. N. Lupas, J. Söding, A galaxy of folds. *Protein Sci.* **19**, 124–130 (2010).
30. S. Nepomnyachiy, N. Ben-Tal, R. Kolodny, Global view of the protein universe. *Proc. Natl. Acad. Sci. U. S. A.* **111**, 11691–11696 (2014).
31. N. Ferruz, *et al.*, Identification and Analysis of Natural Building Blocks for Evolution-Guided Fragment-Based Protein Design. *J. Mol. Biol.* **432**, 3898–3914 (2020).
32. N. Ferruz, F. Michel, F. Lobos, S. Schmidt, B. Höcker, Fuzzle 2.0: Ligand Binding in Natural Protein Building Blocks. *Front. Mol. Biosci.* **8**, 805 (2021).
33. P. S. Huang, S. E. Boyken, D. Baker, The coming of age of de novo protein design. *Nature* **537**, 320–327 (2016).
34. N. Ferruz, J. Noske, B. Höcker, Protlego: A Python package for the analysis and design of chimeric proteins. *Bioinformatics* (2021) https:/doi.org/10.1093/bioinformatics/btab253 (April 29, 2021).
35. M. van Kempen, *et al.*, Foldseek: fast and accurate protein structure search. *bioRxiv*, 2022.02.07.479398 (2022).
36. E. Marcos, *et al.*, De novo design of a non-local β-sheet protein with high stability and accuracy. *Nat. Struct. Mol. Biol. 2018 2511* **25**, 1028–1034 (2018).
37. X. Pan, T. Kortemme, Recent advances in de novo protein design: Principles, methods, and applications. *J. Biol. Chem.* **296**, 100558 (2021).
38. C. Xu, *et al.*, Computational design of transmembrane pores. *Nat. 2020 5857823* **585**, 129–134 (2020).
39. S. Romero-Romero, *et al.*, The Stability Landscape of de novo TIM Barrels Explored by a Modular Design Approach. *J. Mol. Biol.* **433** (2021).
40. P. S. Huang, *et al.*, De novo design of a four-fold symmetric TIM-barrel protein with atomic-level accuracy. *Nat. Chem. Biol.* **12**, 29–34 (2016).
41. N. Anand, *et al.*, Protein sequence design with a learned potential. *Nat. Commun. 2022 131* **13**, 1–11 (2022).
42. S. Kordes, S. Romero-Romero, L. Lutz, B. Höcker, A newly introduced salt bridge cluster improves structural and biophysical properties of de novo TIM barrels. *Protein Sci.* **31**, 513–527 (2022).
43. J. G. Wiese, S. Shanmugaratnam, B. Höcker, Extension of a de novo TIM barrel with a rationally designed secondary structure element. *Protein Sci.* **30**, 982–989 (2021).
44. R. Sennrich, B. Haddow, A. Birch, "Neural Machine Translation of Rare Words with Subword Units." *arXiv Prepr. arXiv 1508.07909 [cs.CL]* (2015)
45. A. Radford, *et al.*, "Language Models are Unsupervised Multitask Learners" https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (Accessed October 5, 2020).
46. J. Rasley, S. Rajbhandari, O. Ruwase, Y. He, DeepSpeed: System Optimizations Enable Training Deep Learning Models with over 100 Billion Parameters. *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 3505–3506 (2020).
47. M. Remmert, A. Biegert, A. Hauser, J. Söding, HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods 2011 92* **9**, 173–175 (2011).
48. N. Ferruz, *et al.*, Identification and Analysis of Natural Building Blocks for Evolution-Guided Fragment-Based Protein Design. *J. Mol. Biol.* **432**, 3898–3914 (2020).

Supporting Information

Fig S1: Histogram of token sizes in the vocabulary. Number of tokens in the vocabulary that show a certain amino acid length. Most tokens consist of tetramers.

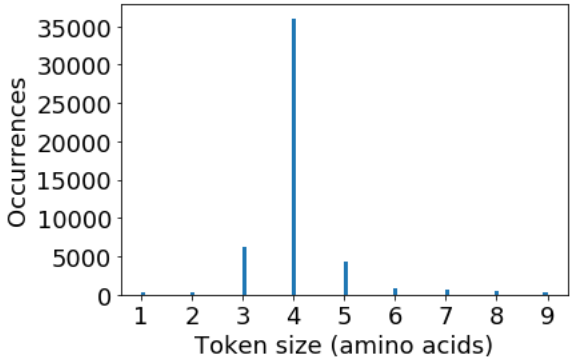

Fig S2: The protein sequence space represented with SCOP95 sequences. Each node in the network represents a protein sequence, which is linked to others when they have an alignment of at least 20 amino acids in length and with a HHsearch probability over 70%. Each of the seven major SCOP classes is depicted in a different color. The graph contains 13,201 vertices and 97,705 edges.