Computational and Structural Biotechnology Journal 21 (2023) 238–250

COMPUTATIONAL AND STRUCTURAL BIOTECHNOLOGY JOURNAL

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

Mini review

From sequence to function through structure: Deep learning for protein design

Noelia Ferruz [a,b,*], Michael Heinzinger [c], Mehmet Akdel [d], Alexander Goncearenco [d], Luca Naef [d], Christian Dallago [c,d,e,*]

[a] Institute of Informatics and Applications, University of Girona, Girona, Spain
[b] Department of Biochemistry, University of Bayreuth, Bayreuth, Germany
[c] Department of Informatics, Bioinformatics & Computational Biology, Technische Universität München, 85748 Garching, Germany
[d] VantAI, 151 W 42nd Street, New York, NY 10036, United States
[e] NVIDIA DE GmbH, Einsteinstraße 172, 81677 München, Germany

ARTICLE INFO

Article history:
Received 31 August 2022
Received in revised form 5 November 2022
Accepted 5 November 2022
Available online 19 November 2022

Keywords:
Protein design
Protein prediction
Drug discovery
Deep learning
Protein language models

ABSTRACT

The process of designing biomolecules, in particular proteins, is witnessing a rapid change in available tooling and approaches, moving from design through physicochemical force fields, to producing plausible, complex sequences fast via end-to-end differentiable statistical models. To achieve conditional and controllable protein design, researchers at the interface of artificial intelligence and biology leverage advances in natural language processing (NLP) and computer vision techniques, coupled with advances in computing hardware to learn patterns from growing biological databases, curated annotations thereof, or both. Once learned, these patterns can be used to provide novel insights into mechanistic biology and the design of biomolecules. However, navigating and understanding the practical applications for the many recent protein design tools is complex. To facilitate this, we 1) document recent advances in deep learning (DL) assisted protein design from the last three years, 2) present a practical pipeline that allows to go from *de novo*-generated sequences to their predicted properties and web-powered visualization within minutes, and 3) leverage it to suggest a generated protein sequence which might be used to engineer a biosynthetic gene cluster to produce a molecular glue-like compound. Lastly, we discuss challenges and highlight opportunities for the protein design field.

Contents

1. Introduction . . . 239
2. Moving from physicochemical functions to deep neural networks in protein research . . . 239
3. The deep learning era of protein sequence and structure generation . . . 240
4. An offline pipeline for protein generation and selection through visual exploration of predicted features . . . 244
5. Use case: selecting protein factories for molecular glues through deep learning . . . 244
6. Discussion . . . 245
Availability . . . 247
CRediT authorship contribution statement . . . 247
Declaration of Competing Interest . . . 247

Abbreviations: ADMM, Alternating Direction Method of Multipliers; CNN, Convolutional Neural Network; DL, Deep learning; FNN, fully-connected neural network; GAN, Generative Adversarial Network; GCN, Graph Convolutional Network; GNN, Graph Neural Network; GO, Gene Ontology; GVP, Geometric Vector Perceptron; LSTM, Long-Short Term Memory; MLP, Multilayer Perceptron; MSA, Multiple Sequence Alignment; NLP, Natural Language Processing; NSR, Natural Sequence Recovery; pLM, protein Language Model; VAE, Variational Autoencoder.

* Corresponding authors at: Institute of Informatics and Applications, University of Girona, Girona, Spain (N. Ferruz). Department of Informatics, Bioinformatics & Computational Biology, Technische Universität München, 85748 Garching, Germany (C. Dallago).

E-mail addresses: noelia.ferruz@udg.edu (N. Ferruz), christian.dallago@tum.de (C. Dallago).

https://doi.org/10.1016/j.csbj.2022.11.014
2001-0370/

Acknowledgements . . . 247
Appendix A. Supplementary data . . . 247
References . . . 247

1. Introduction

Proteins take part in nearly every process of life, controlling a wide variety of functions. This functional versatility and the fact that proteins are nanoscopic, biodegradable materials have motivated tremendous efforts toward designing artificial proteins for medical and industrial applications. In fact, human-designed proteins are widely used across medicine, agriculture and manufacturing, constituting the active form of four out of the top five top selling pharmaceuticals in 2021 [1]. To cite a few more examples, protein engineers have improved the thermostability of a malaria invasion protein for use as a vaccine antigen [2] or improved activity of an enzyme capable of hydrolyzing Polyethylene terephthalate (PET), providing a new, green, and scalable route for plastic waste recycling [3]. Despite the clear objective of designing functional properties, most protein design and engineering strategies have traditionally leveraged a structure-first approach [4], i.e. either by re-engineering known proteins or by designing novel stable structures that then could be further tweaked to target desired functions, e.g. binding to other molecules [5]. The reason why the concomitant design of sequence, structure, and function has remained so challenging is due to the computational intractability of the problem [6]: traditionally, protein design has been tackled as a mathematical optimization, where an algorithm, such as Monte Carlo [7], searched the global minima of a multi-dimensional physicochemical energy function [8].

In recent months, however, an explosion of methods leveraging advances in machine learning provide a fresh alternative for *de-novo* protein design, including the design of long (i.e., with many amino acids) functional proteins. Motivated by the enormous success of structure prediction methods [9–12] and the recent availability of large putative protein structure databases [12,13], some works are exploiting a structure-first approach to designing new folds and proteins [14,15]. In contrast, others operate *sequence-first*, by training large generative language models on vast sequence databases [16,17,p. 2,18,p. 2,19]. Advancing the field by exploiting different modalities (sequence, structure and even function) is fundamental, as no one modality may be able to explain all cell phenomena necessary for the design of biologics [20].

For comprehensive reviews of protein engineering or machine learning methods for protein research, we refer to the works by Yang [21] and Defresne [22]. Yet, with the fundamental advances to protein design in a brief period of time, this manuscript attempts to provide an overview of recent work using AI with a focus on the last three years. To showcase the practical uses of the work presented, we engineer a pipeline for the generation of *de novo* protein sequences selectable for tailored properties that may benefit the protein design community and make use of this novel pipeline to discover sequences that may generate natural products.

In particular, in the following:

1. We describe recent advances in protein design, namely those shifting from a physical-based function paradigm to one that uses deep learning architectures for sequence and structure generation. We aim to give practitioners a waymark to novel tools and their intended uses.
2. We offer a novel pipeline capable of generating protein sequences with tailored properties. In short, we couple ProtGPT2 [17], which generates *de novo* sequences, with ProtT5 [23,p. 5], which predicts properties from them in order to discriminate sequences by desired functions.
3. We dig into the pipeline by presenting a use case for the selection of factory proteins with the predicted ability to produce natural products.
4. We discuss challenges to the design of marketable proteins with controllable properties.

2. Moving from physicochemical functions to deep neural networks in protein research

The *de novo* design of proteins was traditionally approached as an optimization problem where an algorithm searched the global minima of an energy function [24]. This function would evaluate an astronomical number of sequences for a given backbone, quickly leading to an NP-hard problem that required turning to heuristic algorithms and static pairwise potential energy functions to limit computational complexity [25,26]. These approaches met enormous success, with a myriad of *de novo* generated proteins in the last 20 years [27]. These designs have remarkably evolved from often short, alpha-helical peptides [28] and bundles [29] to complex multi-domain architectures [30]. Much has been said about physicochemical-based protein design, we refer the readers to these comprehensive reviews [5,8,27].

More recently however, deep learning (DL) approaches have provided a new venue for protein design research by showing high accuracy in prediction tasks. Highly publicized progress from one such tool came from DeepMind's AlphaFold [31] in December 2018 at CASP13 [32], a multi-step pipeline incorporating DL attempted to solve the decade-long problem of protein 3D structure prediction from sequence. The successor AlphaFold 2 [10], an end-to-end engineered solution, promoted even more excitement due to its incredible ability to accurately predict protein structures from sequences *in-silico* [33,34]. End-to-end techniques refer to the case where a single model learns an underlying mathematical function that maps an input to a complex output. These solutions are of particular interest in the protein research realm as they map relationships that are often complex to capture explicitly with other techniques, for example the relationship between an input sequence and an output 3D structure [35,36].

In parallel to AlphaFold, natural language processing (NLP) methods were leveraged to learn novel protein representations by learning the protein *language*, offering an alternative route to the *explicit* extraction of evolutionary information from multiple sequence alignments (MSAs), historically done by collecting statistics on the co-evolution of residues within MSAs [25]. These models achieved an understanding of proteins by tasking multi-million parameter DL architectures to solve millions of cloze tests (see Supplement 4) from large protein sequence datasets, allowing to encode statistics of protein sequences without supervision on physicochemical or evolutionary relationships [23,37]. While at first protein language model (pLM) representations (shorthand: *embeddings*) did not outperform traditional exploitations of direct physicochemical and evolutionary knowledge [38,39], quick advancements in better mechanisms to process embeddings led to competitive predictors for protein function and structure [11,40–50]. It is thus becoming clear that pLM embeddings are rich inputs to *downstream* prediction methods of protein function and

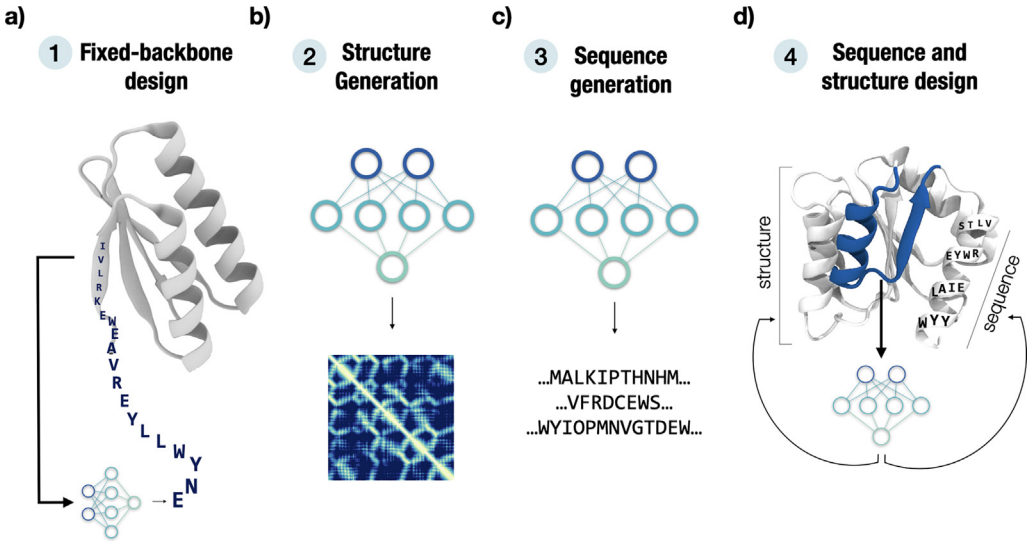

Fig. 1. Leveraging deep learning for different protein design goals. We classify DL approaches in (1) models that try to solve the traditional protein design problem of inverse folding, i.e. find sequences that fold into a desired structure, (2) models capable of generating structure-encoding objects, like contact or distance maps, (3) models that learn to generate protein sequences or (4) concomitantly model sequence and structure to generate either. Categories 1–4 are cross-linked in Table 1.

structure competing with those that exploit MSAs. However, what is encoded in embeddings, including how much *evolutionary information* as defined in MSAs (i.e., residue co-evolution) is implicitly captured, remains a subject of debate, even in light of correlation between MSAs and pLM embeddings in accuracy for 3D structure prediction [41].

While these pLMs excel at embedding representations of input sequences, other types of pLMs (discussed below) are now capable of directly generating new sequences. With increasing experimental validation of "black box" DL methods, the protein design field now has a unique opportunity to generate sequences, interpret DL representations, and build pipelines aiding at various stages of protein design and drug development, from initial library generation to refinement and optimization.

3. The deep learning era of protein sequence and structure generation

Previous reviews have focused on describing the types of neural network architectures used in protein design [22,51]. We focus on the type of problem that these methods attempt to solve: fixed-backbone design (Fig. 1, Panel 1), structure generation (Fig. 1, Panel 2), sequence generation (Fig. 1, Panel 3), and concomitant structure and sequence design most often via *hallucination* (Fig. 1, Panel 4). We discuss several DL methods (**bold and underlined** to match rows Table 1) from the last three years with a focus on those not using Potts models [52,53], which have been reviewed and analyzed elsewhere [53–55]. A virtual version of this table can be found at https://github.com/hefeda/design_tools, while another comprehensive list of DL methods for protein design can be found at https://github.com/Peldom/papers_for_protein_design_using_DL.

The first category of methods we describe focus on solving the traditional protein design problem, i.e., finding a sequence that optimally adopts a desired backbone (Fig. 1, **Panel 1**). The performance of these methods is usually evaluated by *native sequence recovery* (NSR), i.e., the percentage of wild-type amino acids recovered for an input sequence by the design method. While this metric imposes some limitations, given that the identity percentage does not necessarily correlate with expression or functional levels [56], it is nevertheless a convenient measure to evaluate how well the method recapitulates wild-type sequences. Some of the first attempts came from SPIN [57] and **SPIN2** [58,p. 2], which leveraged three-layered fully-connected neural networks (FNN) to learn from structural features embedded as a 1-dimensional (1D) tensor representing backbone torsion angles, local fragment-derived profiles, and global energy-based features. While SPIN and SPIN2 achieved an NSR of 30 % and 34 %, respectively, they suffered from information loss due to the 1D input representation and the shortcomings of FNNs to encode local and global context. Their successor, **SPROF** [59], tried to remedy information loss by leveraging a 2D input and a 2D Convolutional Neural Network (CNN), which brought NSR to 39.8 %. 2D CNN models excelled for years in computer vision tasks as they handle local and global information better than FNNs [60]. In the protein context, these may be spatial and geometric features of 3D structure, which can be extracted in an unsupervised fashion through CNNs, given that protein 3D structures can be proxied accurately (up to symmetry) by 2D representations like contact- or distance maps. For instance, **ProDCoNN** [61] learned to reconstruct sequences by representing atomic discretized environments through voxelized 18 Å-cubic boxes, leading to a NSR of over 45 %. **Anand** [14]'s tool designed sidechain rotamers besides recovering sequence identity at a given input backbone position, reaching an NSR of 57 %. Three of the designs were validated with X-ray crystallography. Meanwhile, Deep Learning researchers attempted to solve training challenges of CNNs, such as the elevated computational cost associated with increasing neural network depth. For instance, DenseNet [62] expanded on the concept of residual connections, i.e. wiring a layer's input to all subsequent *residual blocks*, which inspired **DenseCPD** [63]. Other attempts tried to encode more information in the input to CNNs, e.g. **CNN protein landscape** [64], which amongst others encoded side chain atoms, partial charges and solvent accessibility reaching 60 % NSR, and **TIMED** [65], which besides included reimplementations of several CNN-based protein design methods.

Protein 3D structures can also be represented by graphs where nodes are residues (or atoms) and edges represent structural proximity. Graph neural networks (GNN) directly harness graph repre-

Table 1
Machine-learning-based protein design methods. Methods ordered by their release date, accounting for the date of pre-prints when available. Class (1–4) captured in Fig. 1. Expanded method detail in main text. Unnamed methods are referenced by the first name of the first author. For publications exploring several models/datasets we include the largest. Online and extended version at: https://github.com/hefeda/design_tools.

Method	Input	Output	Architecture	E2E	Dataset	Params	yy/mm
Class 1: 'Fixed-backbone' sequence design							
SPIN2 [58]	3D structure	Sequence	FNN	yes	1,532 X-ray structures	~105 k	18/02
SPROF [59]	3D structure	Sequence	CNN-LSTM	yes	1,532 X-ray structures	–	19/08
Ingraham [35]	3D structure	Sequence	modified Transformer	yes	CATH 4.2 @40 % redundancy sequences/structures	>3k	19/12
ProDCoNN [61]	3D Structure	Sequence	CNN	yes	17,044 X-ray structures	>28 k	19/12
Anand [14]	3D structure	Amino acid and side chain conformation	CNN	yes	53,414 CATH domain structures	–	20/01
DenseCPD [63]	3D structure	Sequence	CNN	yes	11,227 X-ray structures	3 M	20/01
Protein Solver [68]	3D Structure	Sequence	GNN	yes	72,464,122 sequences and adjacency matrices	–	20/03
Norn [105]	Distance map	Sequence	CNN	no	N/A	N/A	20/07
GVP-GNN [72]	3D structures	Sequence	GVP	yes	CATH 4.2 @40 % redundancy sequences/structures	–	20/09
Fold2Seq [75]	3D structure	Sequence	Transformer	yes	CATH 4.2 @100 % redundancy	–	21/06
CNN protein landscape [64]	3D structure	Sequence	CNN	yes	16,569 PDB chains	>10 M	21/08
Orellana [73]	3D structure	Sequence	GCN	yes	CATH 4.2 @40 % redundancy sequences/structures	–	21/11
ABACUS-R [69]	3D structure	Sequence	Transformer	yes	25,234 CATH 4.2 X-ray structures	152 M	22/02
ESM-IF1 [76]	3D structure	Sequence	Modified Transformer	yes	16 k X-ray structures + 1.2 M AF2 predictions	142 M	22/04
McPartlon [74]	3D structures	Sequence	Modified Transformer	yes	37 k 3D structures from the BC40 dataset	–	22/04
TERMinator [66]	3D structures	Potts model	GNN	yes	CATH 4.2 @40 % redundancy sequences/structures	–	22/04
MIF [70]	3D structures	Potts model	structured GNN	yes	CATH 4.2	6.8 M	22/05
Protein MPNN [15]	3D structure	Sequence	Message-passing neural network	yes	CATH 4.2 40 % structure/sequence	1.8 M	22/06
ProDESIGN-LE [71]	3D structure	sequence	Transformer + FNN	yes	5,867,488 residues from PDB40	–	22/07
TIMED [65]	3D structure	Sequence	CNN	yes	32 K structures from the PISCES server	3 M	22/08
PiFold [77]	3D structure	sequence	GNN	yes	–	–	22/09
Class 2: Methods generating structures (contact & distance maps and 3D coordinates)							
64GAN[1] [78]	–	Protein backbone coordinates (via ADMM)	GAN	no	427,659 contact maps	–	18/12
Anand2[1] [79]	–	Protein backbone coordinates (via CNN)	GAN	no	800,000 distance maps	–	19/03
RamaNet [83]	–	Sequence of φ and ψ angles	LSTM	yes	607 helical structures	>2k	19/06
DECO-VAE [84]	Structures represented as graphs (Cα as nodes)	Contact graph	VAE	yes	>650,000 contact graphs	–	20/04
SCUBA2 [85]	Secondary structure motif	Back bone	NC-NN (neighbor counting + neural networks)	yes	12,465 structures	~20 k	22/02
Ig-VAE [81]	–	Protein backbone coordinates	VAE	yes	10,768 individual immunoglobulin domains	–	22/02
GENESIS[2] [86]	Secondary structure motif sketches	Contact map	VAE	no	40,726 backbones with remodeled loops.	–	22/03
ProtDiff & SMCDiff [89]	Optional: structural motif	coordinates	EGNN	yes	4,269 PDB structures	–	22/06
Lai [82]	topology	Protein backbone coordinates	VAE	yes	CATH 4.2 40 % sequences/structures	–	22/07
ProteinSGM [87]	Optional: masked matrices	Distance and torsional angle matrices	SDE + CNN	yes	10,361 structure from CATH 4.3 @ 95 %	–	22/07
FoldingDiff [88]	–	Internal angles	Transformer	yes	CATH 4.2 40 % structures	–	22/09

(*continued on next page*)

Table 1 (*continued*)

Class 1: 'Fixed-backbone' sequence design							
Method	Input	Output	Architecture	E2E	Dataset	Params	yy/mm
Class 3: Methods generating sequences							
ProteinGAN [93]	–	Sequence	GAN	yes	16,706 sequences	60 M	19/10
ProGen [16]	Optional: Sequence or functional label	Sequence	Transformer	yes	280 M sequences	1.2B	20/03
ProtTrans (ProtT5) [23]	Optional: sequence	Sequence	Transformer	yes	BFD100	11B	20/07
EVE [100]	MSA	Sequences	VAE	yes	3,219 MSAs	–	20/12
DARK3 [19]	Optional: sequence	Sequence	Transformer	yes	615,000 sequences	110 M	22/01
ReLSO [101]	sequence	Sequence and predicted value for label	Modified transformer	yes	Directed evolution datasets	–	22/02
ProtGPT2 [17]	Optional: sequence	Sequence	Transformer	yes	44,900,000 sequences	738 M	22/03
RITA [98]	Optional: sequence	Sequence	Transformer	yes	UniRef100	1.2B	22/05
Tranception [99]	Optional: sequence	Sequence	Transformer	yes	Uniref100	700 M	22/05
ProGen2 [18,p.2]	Optional: Sequence or functional label	Sequence	Transformer	yes	UniRef90 + BFD30	6.4B	22/06
Class 4: Concomitant design of sequence and structure							
Hallucination [102]	Random sequence	Sequence	CNN (trRosetta)	no	N/A	N/A	20/07
Constrained hallucination [104]	Sequence/structure	Sequence and structure	CNN (trRosetta)	yes	N/A	N/A	20/11
Constrained Hallucination [104]	Sequence and/or structure	Sequence and/or structure	CNN RoseTTAFold	yes	N/A	N/A	21/11
RFjoint [106]	Sequence and/or structure	Sequence and/or structure	RoseTTAFold	yes	Finetuned with 25 % PDB version 02/2020 + 75 % AF2 structures	N/A	21/11
Protein Diffusion [109]	Secondary structure motif sketches	Sequence/structure	Diffusion	yes	53,414 3D structures (95 % CATH 4.2 S95)	–	22/05
Roney [108]	Random sequence	sequence/structure	AlphaFold2	no	N/A	–	22/06

Architecture: The architecture of the deep learning model; E2E: an end-to-end differentiable solution; Input: the input to run (infer from) the model, e.g. a contact map; Output: the output, e.g. a protein sequence; Dataset: the number and type of samples used to train the method; EGNN: equivariant graph neural network. Params: the exact or estimated number of parameters of the model; SDE: stochastic differential equations; [1]: the 3D recovery was performed in an external, second step; [2]: conditioned generation; –: no input required for generation.

sentations, and thus attempts like **TERMinator** [66,67] or **ProteinSolver** [68], a GNN inspired by Sudoku problems, emerged. ProteinSolver generated sequences from four different folds, two of which were experimentally characterized with circular dichroism. **Ingraham** [35] trained a encoder-decoder *Structured Transformer*, where the GNN encoder learnt protein structures represented by graphs, while the decoder sampled sequences conditioned on the encoder-learn structure representations. Another encoder-decoder, **ABACUS-R** [69], took backbone structural features and sidechain types for surrounding residues of a residue as input to an encoder, and employed a decoder to output the sidechain type for the given residue. **MIF** [70] adapted Ingraham's [35] architecture to a bidirectional denoising model. **ProDESIGN-LE** [71] inputs structural local environments to three encoder layers to output a distribution over the 20 residue types. Graph Vector Perceptrons (**GVP-GNN**) [72] which replaced multilayer perceptrons (MLPs) in GNNs improved performance in protein design and model quality assessment. Through modifications of the GVPs architecture, **Orellana** [73] improved the median sequence recovery from 40.2 % to 44.7 %. In recent work, **McPartlon** [74] introduced a partial masking scheme and side-chain conformation loss to GNNs achieving an NSR of 50.5 % on independent CASP, CATH and TS50 test sets. Other recent studies leveraged encoder-decoder Transformer architectures, e.g. **Fold2seq** [75] learned protein representations jointly from sequence and structure through two encoder modules connected to a decoder module tasked with sampling sequences from the learnt sequence and structure representations. **ESM-IF1** [76] used GVP-GNN encoder layers to represent geometric features, followed by a generic autoregressive encoder-decoder, improving performance over previous GVP-GNN networks. **ProteinMPNN** [15] implemented a Transformer whose encoder embeds the protein backbone coordinates, whilst the decoder outputs suitable sequences. The method was experimentally validated, showing high expression yields (in some cases rates over 88 %) across different tasks. One design was crystallized, showing a more complex fold than most *de novo* proteins up to date [15]. GNNs may however suffer from slow inference times for long sequences; **PiFold** [77] addressed this issue by introducing a novel "*residue featurizer*", achieving up to 70 times speedup while providing a NSR of 51.7 %.

Most of the early work in the second category of methods generating structure encoding objects (Fig. 1, Panel 2) came from the Po-Ssu Huang lab, which initially trained generative adversarial networks (GANs) on contact maps used as input to an convex optimization algorithm (alternating direction method of multipliers algorithm, ADMM) to recover 3D coordinates [78]. Later, the network was improved in **Anand2** to learn from distance maps, and

included a learned coordinate recovery module replacing ADMM [79]. In these methods, sequences were generated from the designed backbones using Rosetta [80]. GANs were subject to a lack of satisfactory accuracy in the coordinate recovery process and the loss of resolution when inputting structures as either contact or distance maps, which led to missing biochemical features and unrealistic designs [81]. **IG-VAE** [81] addressed some of these shortcomings by training a variational autoencoder (VAE) that directly generated 3D coordinates of backbone atoms for class-specific Immunoglobulin proteins. The VAE implemented in **Lai** [82] instead outputs a conformational ensemble of protein structures. Similar methods with the goal of generating structures are **RamaNet** [83], a long-short term memory network (LSTM), trained in an autoregressive manner to output a sequence of φ and ψ angles to design alpha-helical structures, and **DECO-VAE** [84], based on VAEs. As opposed to the fixed-backbone methods (Fig. 1, Panel 1), these generative structure methods allowed to explore novel, unseen topologies, which could host novel functions. Nevertheless, it is often desirable to control the design process, i.e., to condition the sampling towards aspects of function, structure, or sequence. Two methods (SCUBA and GENESIS) allow to do so through inputting a series of secondary structure rules of sketches (such as 'helix-loop-helix'), which guide the network. **SCUBA** [85] used statistical representations of backbones by tasking FNNs to learn from radial kernels encoding different representations of 3D structure. Designs from SCUBA were experimentally evaluated through X-ray crystallography, leading to the discovery of three novel topologies. **GENESIS** [86] implemented a VAE that takes secondary structure sketches and outputs contact maps with finer definitions of secondary structural elements. **ProteinSGM** [87] uses stochastic differential equations (SDE) to generate matrices that capture distance and torsional angles, which are then passed to Rosetta to produce 3D folded structures. **FoldingDiff** [88] merges this two-step approach by using a set of six internal angles, directly producing good quality structures without needing other methods like Rosetta for refinement. **ProtDiff** and **SMCDiff** [89] adopt a similar approach, producing 3D coordinates directly as an output.

Another emerging protein design branch focuses on sequence generation (Fig. 1, Panel 3), mostly inspired by the impressive advances in natural language processing (NLP) over the last few years [51,90]. Language models have been extensively applied to protein sequences (e.g., ESM [37,41] or UniRep [91], focused on protein representation learning). Although many of these models could generate protein sequences by, for example, sampling single-site substitutions using a Monte Carlo approach [92], we do not include them here since they were not evaluated for this objective. Possibly the first advance in the area of models trained specifically for sequence generation came from **ProteinGAN** [93], a GAN trained on the family of malate dehydrogenases (MDH) capable of producing novel functional MDH sequences with as low as 66 % identity to natural protein sequences. Since then, a myriad of autoregressive protein language models (pLMs) with generative capabilities, often leveraging the successful Transformer architecture [94], have followed. Since the applications of autoregressive Transformers to create protein language models have extensively been reviewed [51], we will only briefly mention these. **ProGen** [16] was the first reported decoder-only model specifically trained for protein sequence design and included over 1,100 UniProt [95] control tags (*keywords*). These tags could be used to control the generation process, e.g. by selecting for acyltransferase activity. Transformer models can also be "*fine-tuned*" to achieve a desired goal, practically an alternative technique to using tags for controllable generation. ProGen was employed for the generation of Lysozimes using fine-tuning on five fold diverse Lysozimes families resulting in about 50,000 sequences. Generated sequences showed enzymatic activity in the range of natural counterparts, and one sequence was purified and its structure resolved via X-ray crystallography [96]. **ProtTrans** [23], an extensive probe into the ability of six transformer-based architectures to encode protein sequence knowledge, included the training of autoregressive models (ProtXL, ProtXLNet, ProtElectra-Generator-BFD, and ProtT5) which have, in principle, sequence generation ability, although they were not originally tested for this task. **DARK3** [19] is a decoder-only model with 100 M parameters trained on synthetic sequences. Following the principles of DARK3, **ProtGPT2** leveraged a GPT2-like model [97] an trained on the UniRef50 dataset [95], leading to a model able to generate proteins in unexplored regions of the natural protein space, while presenting natural-like properties [17]. **RITA** [98] included a study on the scalability of generative Transformer models with several model-specific (e.g. perplexity) and application-specific (e.g. sequence properties) benchmarks, similar to **ProGen2** [18,p. 2], which was accompanied by the release of all pre-trained weights and architectures, ranging from 151 M to 6.4B parameter models. Similar to these models, **Tranception** [99] exploited the autoregressive objective, but with a novel attention mechanism aimed at extracting subsequence k-mers, which proved to be very effective for protein modeling. The work included ProteinGym, a benchmark to assay the performance of fitness predictors. Another model with generative ability is **EVE** [100], a VAE used to predict the pathogenicity of protein variants. A particularly interesting application of language models came with **ReLSO** [101], which used a transformer autoencoder paired with function prediction, inferring protein functionality from sequence embeddings. This model can also be used to generate new sequences by optimizing the latent space with gradient ascent.

The last category of design methods encompasses those capable of concomitantly designing sequence and structure (Fig. 1, Panel 4). Possibly the first method in this class, **Hallucination** [102] allowed to "*hallucinate*" 100-amino acid long de-novo protein sequences leveraging trRosetta. The term *hallucination* was inspired by DeepDream [103], a CNN capable of generating mesmerizing, psychedelic images by combining input patterns iteratively (e.g. generating a house from eyes and faces). Protein hallucination works similarly: random sequences (which only have arbitrary local structural patterns) are passed to a structure prediction method, such as trRosetta, which predicts a distance map. The difference between this map and a background distribution trained on high-resolution natural structures is iteratively minimized by mutating the sequence one mutation at the time and recomputing its distance map in order to ultimately reach a minimal (or optimal) distance between background distribution and distance map. Iterating over this process for 40,000 steps using Monte Carlo led to sharply defined distance maps. 129 designs were expressed in E.coli, of which 27 were monomeric and well-folded, with three being validated through crystallization [102]. Similar to other applications, it is often paramount to gain control over specific properties, such as building a scaffold around a particular structural motif, like a binding pocket. **Constrained hallucination** [104] modified the hallucination process in two ways: first by using a composite loss function, combining the losses from **Norn** [105] and Anishchenko [102], which allowed to create a model to concomitantly find a sequence for the structural motif, while hallucinating the scaffold around it; second, the Monte Carlo sampling procedure was replaced by a gradient-based sequence optimization, leading to an 18-fold decrease in sampling time (from 90 to 5 min). In **RFjoint** [106], this approach was further

enhanced by employing RoseTTAFold [107] instead of trRosetta, thus leading to finding a minimum for 3D structure differences instead of distance maps [106] and in **Roney** [108], AlphaFold2 was used as the core model for the hallucination. However, this constrained hallucination schema was still too computationally expensive, and thus the authors fine-tuned a RoseTTAFold variant with a three-term loss which allowed RFjoint to inpaint missing sequences and structures in a few seconds [106]. The method was extensively evaluated experimentally [106]. Lastly, **Protein Diffusion** [109] leveraged diffusion models [110–112] popularized by generative image approaches [113] to train a model capable of generating sequence and structure based on a set of secondary structure input constraints.

4. An offline pipeline for protein generation and selection through visual exploration of predicted features

Despite significant advances, the field appears to remain distant from tools allowing the seamless design of proteins that fulfill specific properties in an end-to-end fashion, and even farther for *the end* to be marketable devices primed to ace clinical trials or pass environmental regulations. A step towards this goal would be DL models that prompted with a set of biological mechanism and industry-relevant properties (e.g., desired thermostability, aggregate viscosity, sequence length, subcellular localization, catalytic capabilities, or binding partners) output a sequence or structure satisfying the selected criteria with high precision in a timely fashion. Whilst this may not yet be possible, we present an offline attempt (i.e., not an end-to-end solution) by combining a generative protein sequence model (Fig. 1, Panel 3) [17] with an oracle discriminator DL model helping to query generated sequences for desired properties [23].

In particular, we unconditionally (i.e., without priors on family, function or structure) generated a set of 100,000 protein sequences using ProtGPT2, and predicted secondary structure [23], Gene Ontology (GO) terms [45], residue ability to bind small molecules, nucleotides or metals [48], protein subcellular localization [47], transmembrane topology [42], residue conservation [43], residue disorder [44] and CATH family [46]. Remarkably, this generated a repertoire of 100,000 protein sequences with ~12 predicted features of structure and function from a single script in approximately 3.5 h (Supplement 1). To analyze whether the generated sequences resemble natural ones, we prepared two more sets to compare against: 1) 100,000 sampled protein sequences from UniRef50 [95] which we call "*U50*" and 2) a *nonsense* sequence set created by shuffling residues within the sequences of the sampled UniRef50 set (e.g. "SEQVENCE" may become "QEEVNCSE") which we call "*random*". We then predict structure and function using the same methods and script applied to the generated sequences for both sets.

Qualitatively comparing distributions of several predicted features for the three sets (Fig. 2) suggests that generated protein sequences resemble natural sequences more than random sequences. When comparing distributions of predicted features between generated and natural sequences, p-values were larger than when comparing the same distributions for either natural or generated against random sequences, suggesting that the newly generated sequences are closer to natural ones than to the random background (analyses included in our Notebooks; see Availability). However, pairwise t-Tests on predicted features with multiple-testing correction failed to give a clear indication about whether generated and natural predicted feature distributions are more similar to each other than those for random. As generated sequences are accompanied by a wealth of interpretable predicted features, we provide a simple Jupyter Notebook (https://github.com/hefeda/PGP) that allows to query the generated protein set in *quasi*-natural language via the predicted features. To cite two examples, a user could shortlist protein sequences with more than 200 residues that have at least 30 % of residues involved in alpha-helical secondary structure and that locate on the outer cell membrane according to GO annotations, or select for sequences shorter than 100 residues with transmembrane strand content and binding to small molecules. The resulting sequences filtered by the desired query displayed in the Jupyter Notebook are linked to LambdaPP [114], a web-server for visual exploration of predicted features that also predicts and displays protein 3D structure using ColabFold [115] and allows 3D structure comparisons through FoldSeek [116].

Whilst not an end-to-end solution, carefully crafted combinations of lightning-fast DL generators and predictors, easy query mechanisms, and rich visual tools allow to push the envelope in discovery of novel candidates, as explored in the following use case.

5. Use case: selecting protein factories for molecular glues through deep learning

Proteins can serve as factories increasingly engineered to manufacture other products, such as the triterpene squalene [117]. Often protein factories are produced by genes in biosynthetic gene clusters (BGC) found in many microorganisms. Protein language model-based deep learning was used to create a tool able to scan genomes for BGCs, assess their functional classification according to Pfam [118], and predict their output product [119]. These protein factories are also the source of many therapeutically used natural products, of which an estimated 50 % of all FDA approved drugs are derived [120]. It was discovered that many of these natural products act through a newly established mechanism [121] used by microbial, plant, fungal and animal cells to regulate protein interaction networks at the proteome scale [121], such as plant hormones Auxin and Jasmonate [122]. These small molecules are now aptly named *molecular glues* as they work by gluing a protein of interest to a regulating protein, often an enzyme called the *effector* protein. Multiple approved drugs with previously unknown mechanisms of action have now been identified to work through this mechanism [123,124]. Molecular glues hold tremendous promise, allowing to drug protein classes that are traditionally considered undruggable, as they don't require binding enzymatically active sites, can work in shallow, featureless pockets [125], as well as on intrinsically disordered proteins which can become ordered at the stabilized protein interface [126,127]. Approved molecular glues, however, have all been discovered through serendipity, and it appears that nature has been much more successful in designing this class systematically.

To show the therapeutic promise of artificially designed proteins, we mine unconditionally ProtGPT2-generated protein sequences for their potential to act as factories to produce novel molecular glues. These sequences could be used as a starting pool in a directed evolution approach. Generating sequences, in contrast to the typically used point mutation and shuffling strategies, can achieve a better balance between exploration of sequence diversity and maintaining reaction stability. To do so, we filtered the set of ProtGPT2-generated sequences discussed previously using embedding-based annotation transfer (EAT). First, we filter sequences by those with the Gene Ontology (GO) annotation "*secondary metabolite biosynthetic process*" (GO:0044550) or its children terms, using a cutoff of embedding euclidean distance of up to 0.6, resulting in 1,345 sequences (we use a more relaxed embedding distance cutoff compared to Littmann et al. [45] in order to derive a sizable set of candidate sequences). Secondly, we compute

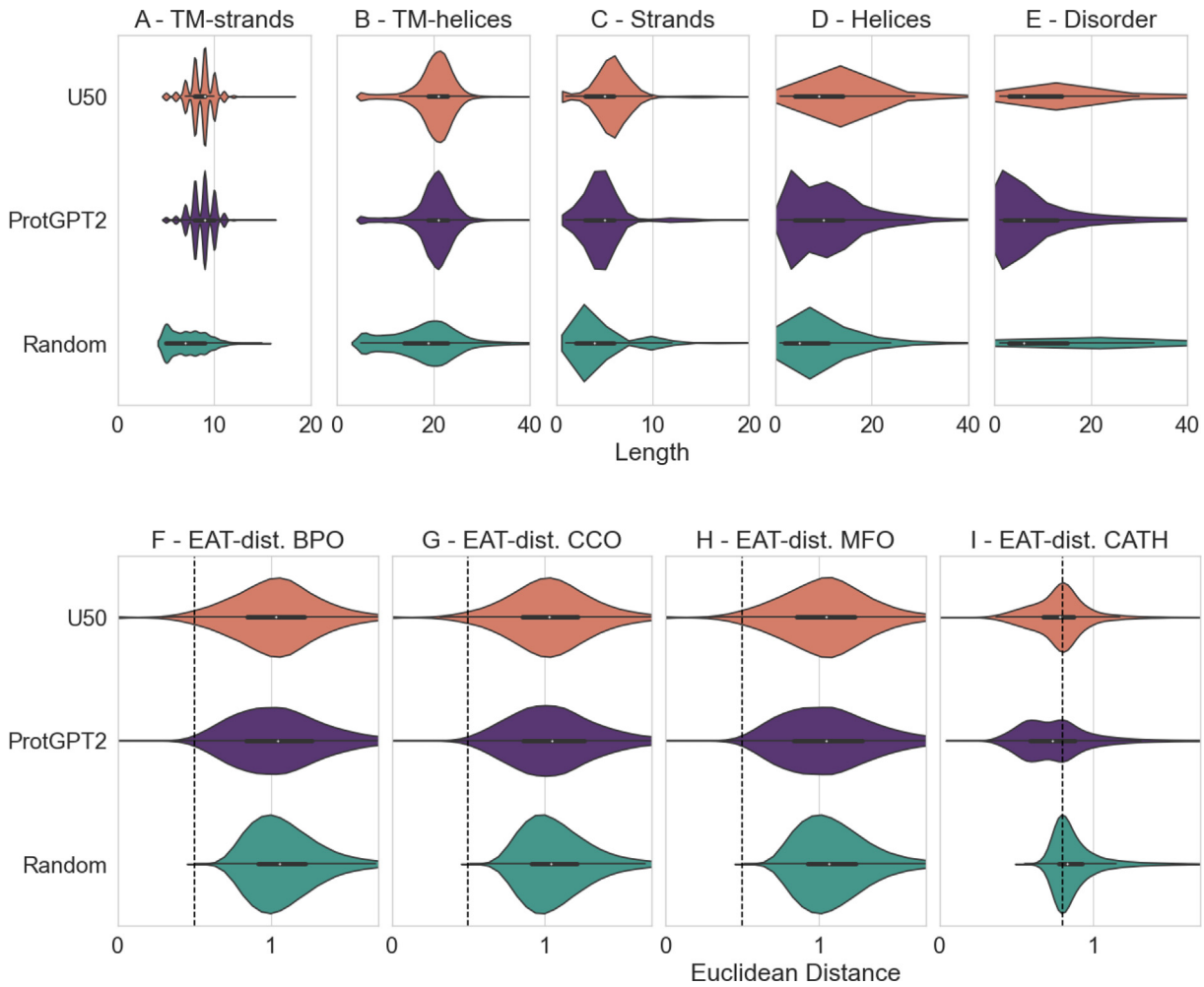

Fig. 2. Distributions of predicted structural and functional features suggest generated proteins closer to natural sequences than random. *Legend*: U50/orange: 100 k sequences sampled from UniRef50; ProtGPT2/violetviolet: 100 k Dl-generated sequences; Random/greengreen: sequences created by randomly shuffling the residues within the U50 set; Length: length of the stretch, e.g. 10 consecutive residues; Euclidean Distance: the Euclidean distance in embedding space to the first sequence in an annotated set. Structure at residue resolution: consecutive residues forming stretches falling into predicted transmembrane strands (A), helices (B), or general strands (C), helices (D) or disorder (E) follow similar stretch length distribution between U50 (orange) and ProtGPT2 (violet), but differ from random (green). For instance, random sequences have a preference for short stretches (A-D, left of distribution) and are overall less likely to be predicted as part of a TM-strand or -helix (A-B). Function at sequence resolution: embedding-based annotation transfer (EAT) distribution of distance between proteins in U50, ProtGPT2 and Random to the closest protein with existing GO annotations taken from SwissProt of either biological processes (BPO/F), cellular compartment (CCO/G) or molecular function (MFO/H), or to sequences with CATH annotations (I). The distance distribution in the high confidence interval (up to Euclidean distance of 0.5 for F-H or up to 0.8 for I, indicated by black, dashed line; thresholds differ due to different embedding spaces and tasks: F-H use ProT5 embedding space to infer function [45], while "I" uses ProtTucker [46] embedding space to infer structure) is similar for U50 (orange) and ProtGPT2 (violet), but differs for random (green). Comparing the three sets suggests that annotation transfers at Euclidean distance greater than the respective thresholds in embedding space (right of vertical dashed line, F-I) invite uncertainty, whilst transfers between proteins at Euclidean embedding distance lower than 0.6 may be considered reliable.

embedding euclidean distances between these 1,345 sequences and sequences in the biosynthetic gene cluster dataset (MiBIG) [128], and remove all generated sequences which fall below a distance of 0.6 to any sequence in MiBIG. We discover that even with no further optimization, 234 generated, filtered sequences are *similar* (up to an embedding euclidean distance of 0.6 to both relevant GO annotations and annotated sequences with MiBIG) to naturally occurring BGC proteins (Fig. 3).

We focus on one generated protein (sequence in Supplement 2) which is close to a short chain dehydrogenase BGC constituent involved in producing fusicoccin A (one of the molecules depicted in Supplement 3, Fig. S1, panel A). Fusicoccin A is a phytotoxic molecular glue which was originally discovered to stabilize the interface between the scaffolding protein 14-3-3 and H+-ATPases from Arabidopsis thaliana (AHA2) and Nicotiana plumbaginifolia (PMA2) [129,130], but has since then been expanded semi-synthetically as an anti-cancer drug [131]. By searching for similar proteins in embedding space for different datasets (Fig. 3, panel B and C) and predicted fold space, by querying the AlphaFold database [13] through FoldSeek [116] using the ColabFold [115] predicted 3D structure for the designed sequence (Fig. 3, panel D), three natural proteins emerged. All three predicted structures belong to fungal, short-chain dehydrogenases with very similar folds, but are involved in the production of completely different natural products, as expected from different BGCs. This demonstrates a potential *in-silico* workflow of generating sequences, with the aim of evolving specific BGCs to produce desired molecules after further *in-vitro* confirmation, for instance by applying directed evolution [132].

6. Discussion

Deep learning ushers in a new wave of tools for protein design, engineering, prediction, and optimization. The efficient combination of *in-silico* and *in-vitro* approaches in multi-stage

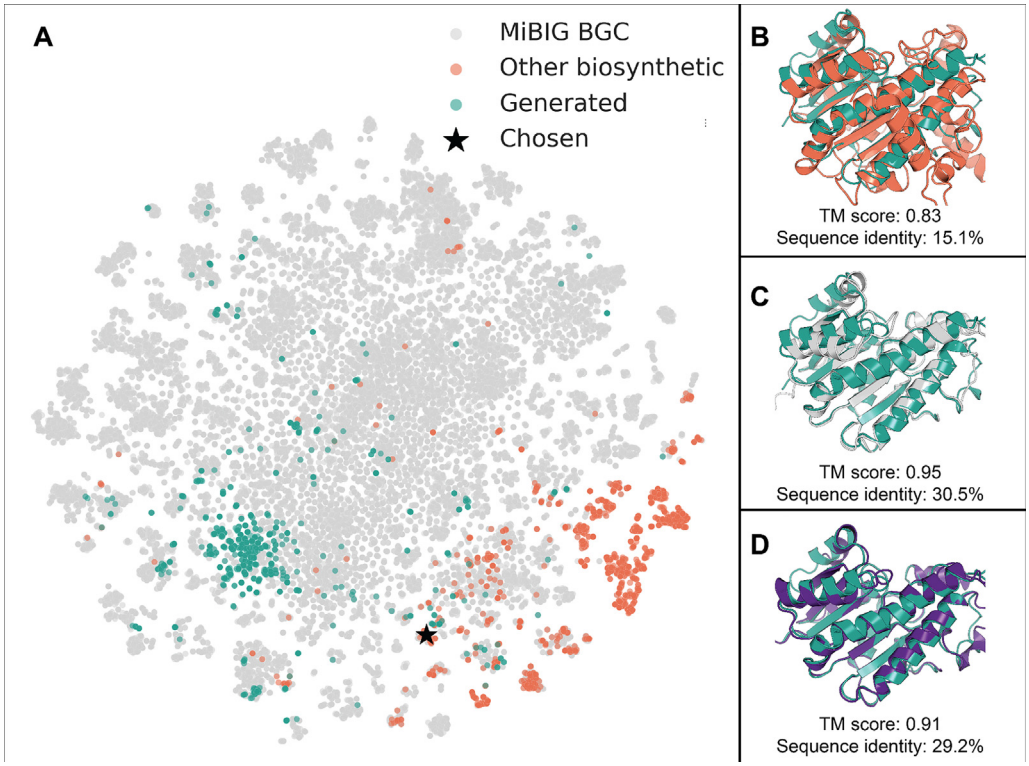

Fig. 3. Unconditionally generated protein sequences close to natural sequences with desired function. *t*-SNE projection of protein embeddings for enzymes in biosynthetic gene clusters (as taken from MiBIG [128]; gray), other biosynthetic enzymes (from UniProt; orange), and generated sequences in green (panel A). On the right the structural superposition of the chosen protein (★ in panel A and green in B-D, predicted by ColabFold [106]) with the closest biosynthetic protein from UniProt (accession: D7UTD0 - obtained from the Alpha Fold database [13]; orange; panel B), the closest BGC protein from MiBig (UniProt: D2IKP3 - obtained from the AlphaFold database, MiBIG: BGC0000304|9); gray; panel C), and the closest protein structure from the AlphaFold database found via FoldSeek [116] (id: AF-Q00674; purple; panel D).

pipelines, such as in drug discovery, remains complex, costly and often requires many types of expertise. However, practitioners now have access to an increasing collection of *in-silico* DL tools which particularly address the first stages of these pipelines (Table 1). DL methods have started a revolution in protein design, a field that has undergone a paradigm shift over the last few years, moving away from traditional physico-chemical energy functions. As we highlight in the use case, generative sequence models can be employed to create large libraries of plausible sequences for which oracle models predict structure and function properties in a matter of hours (Fig. 2). This process requires no technical DL expertise and provides data from which experts can select and refine to one or a set of promising candidates that constitute starting points for *in-vitro* experiments (Fig. 3).

End-to-end protein design for marketable biologics remains remarkably complex. Despite advances in end-to-end DL models for protein design, except for a few recent breakthroughs [133], bridging the gap from *computational* to *marketable* (i.e., using a single computational system to generate devices that can be put on pharmacy shelves) remains difficult, an unfortunate reality since proteins would have the potential to tackle many emerging biomedical and environmental challenges [134].

Two interesting aspects of DL solutions are their *end-to-end* nature and the ability to combine different losses, i.e. to condition the output (e.g. sequence) on the input (e.g. structure) using a mathematical formulation which takes into account aspects (e.g. thermostability) interesting in the application context (e.g. biopharma). DL models for protein design are optimized end-to-end on large biological collections, which are often limited by experimental shortcomings, for instance capturing a static notion of protein 3D structure, as opposed to its dynamic nature (e.g., conformational modifications during protein–protein interactions). Thus, while DL models for protein design may encode biological mechanisms required to propose viable biologics, they cannot yet be blindly utilized without experimental validation, and do not yet explicitly account for marketable variables. Similarly, as protein function lacks rigorous definition (e.g., may be binding or localization, or both) and scale [135], designing around protein function remains more challenging than designing around sequence or structure, and selecting for function in production is often better validated through targeted functional assays.

An opportunity to connect *in-silico* with *in-vitro* approaches, where physical experiments, accounting for all variables, inform DL models in a differentiable manner may come from advancements in automated labs [136,137] and a new frontier of cloud labs [138]. However, a bottleneck from combining DL models with physical experimentation is high-throughput [135], as digital systems often vastly outpace physical counterparts in efficiency and scale. Yet, digital copies of physical labs (or digital twins [139]), which have already been proven useful in boosting manufacturing [140,141] provide an opportunity for the future. A DL solution aiming at outputting marketable devices should also account for non-biological variables, such as the ones influencing production costs or clinical response. Data necessary to inform such models is however lacking, although improvements are underway, such as for clinical trial design [142].

Looking forward: protein design community on a rigorous journey to establish goals. In the first half of the 1990s, at a time when having *solved the protein folding challenge* sporadically made headlines, the Critical Assessment of Structure Prediction (CASP) [32] set the standard against which *in-silico* predictive methods for protein structure needed to prove advancement. In combination with establishing data standardization and a single data repository, this led to multiple revolutionary approaches, from early

uses of single-frequency models (i.e., positional scoring matrices), to complex co-evolutionary representations through direct-coupling analysis [25], all the way to end-to-end DL solutions like AlphaFold 2 [10]. Arguably, the success of structure prediction is a combination of many factors, including technical, biological understanding and intuition, and more complex and principled statistical methods. Fundamentally, however, structure prediction through CASP sets an example of how innovation can be fostered.

As the protein design field moves to more complex DL approaches, benchmarks, which promoted the success of structure prediction tools, appear to be lacking. Arguably, this is due to the underlying complexity of defining, in principle, what protein design is supposed to be, especially as it is moving from theoretical exercise to practical applications. Is it sequence design? Is it structure design? Is it sequence and structure that culminate in function? In fact, function would most likely be the design goal from a practical standpoint. Attempts at measuring advancements in protein function prediction exist, e.g. CAFA [143] and CAGI [144], however they focus on scoring how *in-silico* tools predict known function from sequence, rather than their ability to infer proteins (sequences or structures) that perform a desired, sometimes non-naturally found function. Conversely, model developers score their tools by metrics like Natural Sequence Recovery (NSR), which validate a model's ability to link structure to sequence, but often not a model's ability to generate diverse sequences fitting a desired structure [56]. A recent push in benchmarks scoring models' ability to engineer proteins [99,145–148] highlights three aspects that protein design tools should strive to solve: 1) emulate laboratory conditions, i.e., extrapolate from very little available data; 2) set multiple function generalization goals, i.e, measure different aspects of function, with the intent of finding an optimal solution rather than maximizing any one metric; 3) focus on the ability to address out of observed data distributions, i.e., design proteins that achieve functions not observed in nature.

Ultimately, by overcoming the challenges of measuring advances, deep learning is set to enable protein engineers to design sequence, structure, and function with controllable properties.

Availability

pLM generated and UniRef50 sampled sequence sets and predictions are available at http://data.bioembeddings.com/public/design. Code-base and Notebooks for analysis are available at https://github.com/hefeda/PGP. An online version of Table 1 can be found at https://github.com/hefeda/design_tools.

CRediT authorship contribution statement

Noelia Ferruz: Conceptualization, Methodology, Software, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Michael Heinzinger:** Conceptualization, Methodology, Software, Formal analysis, Writing – original draft, Writing – review & editing, Visualization. **Mehmet Akdel:** Validation, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Alexander Goncearenco:** Validation, Writing – original draft, Visualization. **Luca Naef:** Conceptualization, Validation, Writing – original draft, Visualization. **Christian Dallago:** Conceptualization, Methodology, Validation, Resources, Writing – original draft, Writing – review & editing, Visualization.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: CD was employed by VantAI and NVIDIA at different periods during the time of writing. MA, AG, LN are employees of VantAI. NVIDIA and VantAI had no influence on the contents of this manuscript.

Acknowledgements

The authors wish to thank the anonymous reviewers for their invaluable support in improving the manuscript. We wish to thank Burkhard Rost for inspiring conversations and hardware support. We extend gratitude to all who deposit experimental data in public databases, to those who maintain these databases, and those who make analytical and predictive methods freely available. Additionally, thanks to Simon Duerr, Gina El Nesr, Sergey Ovchinnikov, Kevin K. Yang and all those curating scientific knowledge into easy to parse lists. NF acknowledges support from a Beatriu de Pinos MSCA-COFUND Fellowship (project 2020-BP-00130). CD acknowledges support by the Bundesministerium für Bildung und Forschung (BMBF) through the program "Software Campus 2.0 (TU München)" – project number 01IS17049, and the library of the Technical University of Munich for support in open access costs.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.csbj.2022.11.014.

References

[1] Buntz B. 50 of 2021's best-selling pharmaceuticals accessed Aug. 15, 2022. Drug Discov Dev 2022;29. https://www.drugdiscoverytrends.com/50-of-2021s-best-selling-pharmaceuticals/.
[2] Campeotto I et al. One-step design of a stable variant of the malaria invasion protein RH5 for use as a vaccine immunogen. Proc Natl Acad Sci 2017;114 (5):998–1002. https://doi.org/10.1073/pnas.1616903114.
[3] Lu H et al. Machine learning-aided engineering of hydrolases for PET depolymerization. Nature 2022;604(7907):662–7. https://doi.org/10.1038/s41586-022-04599-z.
[4] Scheibenreif L, Littmann M, Orengo C, Rost B. FunFam protein families improve residue level molecular function prediction. BMC Bioinf 2019;20 (1):400. https://doi.org/10.1186/s12859-019-2988-x.
[5] Woolfson DN. A brief history of De Novo protein design: minimal, rational, and computational. J Mol Biol 2021;433(20):. https://doi.org/10.1016/j.jmb.2021.167160167160.
[6] Pierce NA, Winfree E. Protein design is NP-hard. Protein Eng Des Sel 2002;15 (10):779–82. https://doi.org/10.1093/protein/15.10.779.
[7] Metropolis N, Ulam S. The Monte Carlo method. J Am Stat Assoc 1949;44 (247):335–41. https://doi.org/10.1080/01621459.1949.10483310.
[8] Kuhlman B, Bradley P. Advances in protein structure prediction and design. Nat Rev Mol Cell Biol 2019;20(11):681–97. https://doi.org/10.1038/s41580-019-0163-x.
[9] Ahdritz G, Bouatta N, Kadyan S, Xia Q, Gerecke W, AlQuraishi M. OpenFold. Zenodo 2021. https://doi.org/10.5281/ZENODO.5709539.
[10] Jumper J et al. Highly accurate protein structure prediction with AlphaFold. Nature 2021;596(7873):583–9. https://doi.org/10.1038/s41586-021-03819-2.
[11] Wu R et al. High-resolution de novo structure prediction from primary sequence. bioRxiv 2022. https://doi.org/10.1101/2022.07.21.500999.
[12] Humphreys IR, et al., Computed structures of core eukaryotic protein complexes. Science, vol. 374, no. 6573, eabm4805, doi: 10.1126/science.abm4805.
[13] M. Varadi et al., AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models, Nucleic Acids Res., vol. 50, no. D1, pp. D439–D444, Jan. 2022, doi: 10.1093/nar/gkab1061.
[14] Anand N et al. Protein sequence design with a learned potential. Nat Commun 2022;13(1):746. https://doi.org/10.1038/s41467-022-28313-9.
[15] Dauparas J, et al., Robust deep learning based protein sequence design using ProteinMPNN. bioRxiv, Jun. 04, 2022. doi: 10.1101/2022.06.03.494563.
[16] Madani A, et al., ProGen: Language Modeling for Protein Generation. arXiv, Mar. 07, 2020. Accessed: Jul. 28, 2022. [Online]. Available: http://arxiv.org/abs/2004.03497.
[17] Ferruz N, Schmidt S, Höcker B. ProtGPT2 is a deep unsupervised language model for protein design. Nat Commun 2022;13(1):4348. https://doi.org/10.1038/s41467-022-32007-7.
[18] Nijkamp E, Ruffolo J, Weinstein EN, Naik N, Madani A, ProGen2: exploring the boundaries of protein language models. arXiv, Jun. 27, 2022. Accessed: Jul. 28, 2022. [Online]. Available: http://arxiv.org/abs/2206.13517.

[19] Moffat L, Kandathil SM, Jones DT. Design in the DARK: learning deep generative models for De Novo protein design. bioRxiv 2022. https://doi.org/10.1101/2022.01.27.478087.

[20] Lowe D, Why AlphaFold won't revolutionise drug discovery, *Chem World*, 2022. https://www.chemistryworld.com/opinion/why-alphafold-wont-revolutionise-drug-discovery/4016051.article (accessed Aug. 07, 2022).

[21] Yang KK, Wu Z, Arnold FH. Machine-learning-guided directed evolution for protein engineering. Nat Methods 2019;16(8):687–94. https://doi.org/10.1038/s41592-019-0496-6.

[22] Defresne M, Barbe S, Schiex T. Protein design with deep learning. Int J Mol Sci 2021;22(21):11741. https://doi.org/10.3390/ijms222111741.

[23] Elnaggar A, et al., ProtTrans: Towards cracking the language of lifes code through self-supervised deep learning and high performance computing, *IEEE Trans Pattern Anal Mach Intell*, 2021;1-1, doi: 10.1109/TPAMI.2021.3095381.

[24] Gainza P, Nisonoff HM, Donald BR. Algorithms for protein design. Curr Opin Struct Biol 2016;39:16–26. https://doi.org/10.1016/j.sbi.2016.03.006.

[25] Morcos F et al. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. Proc Natl Acad Sci 2011;108 (49). https://doi.org/10.1073/pnas.1111471108.

[26] Das R, Baker D. Macromolecular modeling with Rosetta. Annu Rev Biochem 2008;77(1):363–82. https://doi.org/10.1146/annurev.biochem.77.062906.171838.

[27] Huang P-S, Boyken SE, Baker D. The coming of age of de novo protein design. Nature 2016;537(7620):320–7. https://doi.org/10.1038/nature19946.

[28] Hill CP, Anderson DH, Wesson L, DeGrado WF, Eisenberg D. Crystal structure of alpha 1: implications for protein design. Science 1990;249(4968):543–6. https://doi.org/10.1126/science.2382133.

[29] Lovejoy B, Choe S, Cascio D, McRorie DK, DeGrado WF, Eisenberg D. Crystal structure of a synthetic triple-stranded alpha-helical bundle. Science 1993;259(5099):1288–93. https://doi.org/10.1126/science.8446897.

[30] Courbet A et al. Computational design of mechanically coupled axle-rotor protein assemblies. Science 2022;376(6591):383–90. https://doi.org/10.1126/science.abm1183.

[31] Senior AW et al. Improved protein structure prediction using potentials from deep learning. Nature 2020;577(7792):706–10. https://doi.org/10.1038/s41586-019-1923-7.

[32] Kryshtafovych A, Schwede T, Topf M, Fidelis K, Moult J. Critical assessment of methods of protein structure prediction (CASP)—Round XIII. Proteins Struct Funct Bioinf 2019;87(12):1011–20. https://doi.org/10.1002/prot.25823.

[33] AlQuraishi M. A watershed moment for protein structure prediction. Nature 2020;577(7792):627–8. https://doi.org/10.1038/d41586-019-03951-0.

[34] Method of the Year 2021: Protein structure prediction," *Nature*. https://www.nature.com/collections/dfejabhghd (accessed Aug. 05, 2022).

[35] Ingraham J, Garg V, Barzilay R, Jaakkola T, Generative models for graph-based protein design, in *Advances in neural information processing systems*, 2019, vol. 32. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/f3a4ff4839c56a5f460c88cce3666a2b-Paper.pdf.

[36] Ingraham J, Riesselman A, Sander C, Marks D, Learning protein structure with a differentiable simulator. In *International conference on learning representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Byg3y3C9Km.

[37] Rives A et al. **Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences**. Proc. Natl. Acad. Sci. Apr. 2021;118(15):. https://doi.org/10.1073/pnas.2016239118e2016239118.

[38] Heinzinger M et al. Modeling aspects of the language of life through transfer-learning protein sequences. BMC Bioinf 2019;20(1):723.

[39] Rao R, et al., Evaluating protein transfer learning with TAPE. In: *Advances in Neural Information Processing Systems 32*, 2019, pp. 9689–9701. Accessed: Mar. 21, 2020. [Online]. Available: http://papers.nips.cc/paper/9163-evaluating-protein-transfer-learning-with-tape.pdf.

[40] Meier J, Rao R, Verkuil R, Liu J, Sercu T, Rives A, Language models enable zero-shot prediction of the effects of mutations on protein function. *Adv Neural Inf Process Syst*, 2021;34:29287–303. [Online]. Available: https://proceedings.neurips.cc/paper/2021/file/f51338d736f95dd42427296047067694-Paper.pdf.

[41] Lin Z et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. BioRxiv 2022. https://doi.org/10.1101/2022.07.20.500902.

[42] Bernhofer M, Rost B. TMbed: transmembrane proteins predicted through language model embeddings. BMC Bioinf 2022;23(1):326. https://doi.org/10.1186/s12859-022-04873-x.

[43] Marquet C et al. Embeddings from protein language models predict conservation and variant effects. Hum Genet 2021. https://doi.org/10.1007/s00439-021-02411-y.

[44] Ilzhoefer D, Heinzinger M, Rost B. SETH predicts nuances of residue disorder from protein embeddings. BioRxiv 2022. https://doi.org/10.1101/2022.06.23.497276.

[45] Littmann M, Heinzinger M, Dallago C, Olenyi T, Rost B. Embeddings from deep learning transfer GO annotations beyond homology. Sci Rep 2021;11 (1):1–14.

[46] Heinzinger M, Littmann M, Sillitoe I, Bordin N, Orengo C, Rost B. Contrastive learning on protein embeddings enlightens midnight zone. NAR Genomics Bioinf 2022;4(2). https://doi.org/10.1093/nargab/lqac043.

[47] Stärk H, Dallago C, Heinzinger M, Rost B. Light attention predicts protein location from the language of life. Bioinf Adv 2021;1(1). https://doi.org/10.1093/bioadv/vbab035.

[48] Littmann M, Heinzinger M, Dallago C, Weissenow K, Rost B. Protein embeddings and deep learning predict binding residues for various ligand classes. Sci Rep 2021;11(1):1–15.

[49] V. Thumuluri, J.J. Almagro Armenteros, A.R. Johansen, H. Nielsen, O. Winther. DeepLoc 2.0: multi-label subcellular localization prediction using protein language models. Nucleic Acids Res. doi: 10.1093/nar/gkac278.

[50] M.H. Høie et al., NetSurfP-3.0: accurate and fast prediction of protein structural features by protein language models and deep learning, Nucleic Acids Res., vol. 50, no. W1, pp. W510–W515, Jun. 2022, doi: 10.1093/nar/gkac439.

[51] Ferruz N, Höcker B. Controllable protein design with language models. Nat Mach Intell 2022;4(6):521–32. https://doi.org/10.1038/s42256-022-00499-z.

[52] Wang H, Feng S, Liu S, Ovchinnikov S, Disentanglement of entropy and coevolution using spectral regularization. bioRxiv, Mar. 07, 2022. doi: 10.1101/2022.03.04.483009.

[53] McGee F et al. The generative capacity of probabilistic protein sequence models. Nat Commun 2021;12(1):1. https://doi.org/10.1038/s41467-021-26529-9.

[54] Wilburn GW, Eddy SR. Remote homology search with hidden Potts models. PLOS Comput Biol 2020;16(11):e1008085. https://doi.org/10.1371/journal.pcbi.1008085.

[55] Levy RM, Haldane A, Flynn WF. Potts Hamiltonian models of protein co-variation, free energy landscapes, and evolutionary fitness. Curr Opin Struct Biol 2017;43:55–62. https://doi.org/10.1016/j.sbi.2016.11.004.

[56] Castorina LV, Petrenas R, Subr K, Wood CW. PDBench: evaluating computational methods for protein sequence design. arXiv 2021.. https://doi.org/10.48550/arXiv.2109.07925.

[57] Li Z, Yang Y, Faraggi E, Zhan J, Zhou Y. Direct prediction of profiles of sequences compatible with a protein structure by neural networks with fragment-based local and energy-based nonlocal profiles. Proteins 2014;82 (10):2565–73. https://doi.org/10.1002/prot.24620.

[58] O'Connell J et al. SPIN2: Predicting sequence profiles from protein structures using deep neural networks. Proteins Struct Funct Bioinf 2018;86(6):629–33. https://doi.org/10.1002/prot.25489.

[59] Chen S et al. To Improve protein sequence profile prediction through image captioning on pairwise residue distance map. J Chem Inf Model 2020;60 (1):391–9. https://doi.org/10.1021/acs.jcim.9b00438.

[60] Krizhevsky A, Sutskever I, Hinton GE, ImageNet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, 2012, vol. 25. Accessed: Aug. 28, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

[61] Zhang Y et al. ProDCoNN: Protein design using a convolutional neural network. Proteins Struct Funct Bioinf 2020;88(7):819–29. https://doi.org/10.1002/prot.25868.

[62] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ, Densely connected convolutional networks. In: *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, Jul. 2017, pp. 2261–2269. doi: 10.1109/CVPR.2017.243.

[63] Qi Y, Zhang JZH. DenseCPD: improving the accuracy of neural-network-based computational protein sequence design with DenseNet. J Chem Inf Model 2020;60(3):1245–52. https://doi.org/10.1021/acs.jcim.0c00043.

[64] Kulikova AV, Diaz DJ, Loy JM, Ellington AD, Wilke CO. Learning the local landscape of protein structures with convolutional neural networks. J Biol Phys 2021;47(4):435–54. https://doi.org/10.1007/s10867-021-09593-6.

[65] Castorina LV, Subr K, Wood CW. TIMED-design: efficient protein sequence design with deep learning. Zenodo 2022. https://doi.org/10.5281/zenodo.6997495.

[66] Li AJ, Sundar V, Grigoryan G, Keating AE. TERMinator: A neural framework for structure-based protein design using tertiary repeating motifs. arXiv 2022;27. https://doi.org/10.48550/arXiv.2204.13048.

[67] Li AJ, Lu M, Desta I, Sundar V, Grigoryan G, Keating AE, Neural network-derived potts models for structure-based protein design using backbone atomic coordinates and tertiary motifs. bioRxiv, p. 2022.08.02.501736, 2022. doi: 10.1101/2022.08.02.501736.

[68] Strokach A, Becerra D, Corbi-Verge C, Perez-Riba A, Kim PM. Fast and flexible protein design using deep graph neural networks. Cell Syst 2020;11 (4):402–411.e4. https://doi.org/10.1016/j.cels.2020.08.016.

[69] Liu Y et al. Rotamer-free protein sequence design based on deep learning and self-consistency. Nat Comput Sci 2022;2(7):7. https://doi.org/10.1038/s43588-022-00273-6.

[70] Yang KK, Zanichelli N, Yeh H. Masked inverse folding with sequence transfer for protein representation learning. bioRxiv 2022. https://doi.org/10.1101/2022.05.25.493516.

[71] Huang B et al. Accurate and efficient protein sequence design through learning concise local environment of residues. bioRxiv 2022. https://doi.org/10.1101/2022.06.25.497605.

[72] Jing B, Eismann S, Suriana P, Townshend RJL, Dror R, Learning from protein structure with geometric vector perceptrons. arXiv, 2021. doi: 10.48550/arXiv.2009.01411.

[73] Orellana GA, Caceres-Delpiano J, Ibañez R, Dunne MP, Alvarez L. Protein sequence sampling and prediction from structural data. bioRxiv 2021. https://doi.org/10.1101/2021.09.06.459171.

[74] McPartlon M, Lai B, Xu J, A Deep SE(3)-equivariant model for learning inverse protein folding. bioRxiv, p. 2022.04.15.488492, Apr. 16, 2022. doi: 10.1101/2022.04.15.488492.

[75] Cao Y, Das P, Chenthamarakshan V, Chen P-Y, Melnyk I, Shen Y. Fold2Seq: A joint sequence (1D)-Fold (3D) embedding-based generative model for protein design. arXiv 2021. https://doi.org/10.48550/arXiv.2106.13058.

[76] Hsu C, et al., Learning inverse folding from millions of predicted structures. bioRxiv, 2022;2022.04.10.487779. doi: 10.1101/2022.04.10.487779.

[77] Gao Z, Tan C, Li SZ. PiFold: Toward effective and efficient protein inverse folding. arXiv 2022. https://doi.org/10.48550/arXiv.2209.12643.

[78] Anand N, Huang P, Generative modeling for protein structures. In: *Advances in Neural Information Processing Systems*, 2018, vol. 31. Accessed: Aug. 08, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2018/hash/afa299a4d1d8c52e75dd8a24c3ce534f-Abstract.html.

[79] Anand N, Eguchi R, Huang P-S, Fully differentiable full-atom protein backbone generation, Jul. 2022, Accessed: Aug. 22, 2022. [Online]. Available: https://openreview.net/forum?id=SJxnVL8YOV.

[80] Alford RF et al. The Rosetta all-atom energy function for macromolecular modeling and design. J Chem Theory Comput 2017;13(6):3031–48. https://doi.org/10.1021/acs.jctc.7b00125.

[81] Eguchi RR, Choe CA, Huang P-S. Ig-VAE: Generative modeling of protein structure by direct 3D coordinate generation. PLOS Comput Biol 2022;18(6):e1010271. https://doi.org/10.1371/journal.pcbi.1010271.

[82] Lai B, McPartlon M, Xu J, End-to-End deep structure generative model for protein design. bioRxiv, 2022;2022.07.09.499440. doi: 10.1101/2022.07.09.499440.

[83] Sabban S, Markovsky M. RamaNet: Computational de novo helical protein backbone design using a long short-term memory generative neural network. bioRxiv 2020. https://doi.org/10.1101/671552.

[84] Guo X, Du Y, Tadepalli S, Zhao L, Shehu A. Generating tertiary protein structures via interpretable graph variational autoencoders. Bioinforma Adv 2021;1(1):vbab036. https://doi.org/10.1093/bioadv/vbab036.

[85] Huang B et al. A backbone-centred energy function of neural networks for protein design. Nature 2022;602(7897):523–8. https://doi.org/10.1038/s41586-021-04383-5.

[86] Harteveld Z, et al., Deep sharpening of topological features for de novo protein design. In: presented at the ICLR2022 Machine Learning for Drug Discovery, May 2022. Accessed: Aug. 12, 2022. [Online]. Available: https://openreview.net/forum?id=DwN81YIXGQP.

[87] Lee JS, Kim PM. ProteinSGM: Score-based generative modeling for de novo protein design. bioRxiv 2022. https://doi.org/10.1101/2022.07.13.499967.

[88] Wu KE, Yang KK, van den Berg R, Zou JY, Lu AX, Amini AP. Protein structure generation via folding diffusion. arXiv 2022. https://doi.org/10.48550/arXiv.2209.15611.

[89] Trippe BL, et al., Diffusion probabilistic modeling of protein backbones in 3D for the motif-scaffolding problem. arXiv, 2022. doi: 10.48550/arXiv.2206.04119.

[90] Ofer D, Brandes N, Linial M. The language of proteins: NLP, machine learning & protein sequences. Comput Struct Biotechnol J 2021;19:1750–8. https://doi.org/10.1016/j.csbj.2021.03.022.

[91] Alley EC, Khimulya G, Biswas S, AlQuraishi M, Church GM. Unified rational protein engineering with sequence-based deep representation learning. Nat Methods 2019;16(12):1315–22. https://doi.org/10.1038/s41592-019-0598-1.

[92] Biswas S, Khimulya G, Alley EC, Esvelt KM, Church GM. Low-N protein engineering with data-efficient deep learning. Nat Methods 2021;18(4):389–96. https://doi.org/10.1038/s41592-021-01100-y.

[93] Repecka D et al. Expanding functional protein sequence spaces using generative adversarial networks. Nat Mach Intell 2021;3(4):324–33. https://doi.org/10.1038/s42256-021-00310-5.

[94] Vaswani A, et al., Attention is all you need. arXiv, 2017. doi: 10.48550/arXiv.1706.03762.

[95] The UniProt Consortium. UniProt: the universal protein knowledgebase in 2021. Nucleic Acids Res 2021. https://doi.org/10.1093/nar/gkaa1100.

[96] Madani A et al. Deep neural language modeling enables functional protein generation across families. bioRxiv 2021. https://doi.org/10.1101/2021.07.18.452833.

[97] Better language models and their implications, OpenAI, Feb. 14, 2019. https://openai.com/blog/better-language-models/ (accessed Aug. 20, 2022).

[98] Hesslow D, Zanichelli N, Notin P, Poli I, Marks D, RITA: a study on scaling up generative protein sequence models. arXiv, 2022. doi: 10.48550/arXiv.2205.05789.

[99] Notin P, et al., Tranception: protein fitness prediction with autoregressive transformers and inference-time retrieval. In: *Proceedings of the 39th international conference on machine learning*, Jun. 2022, pp. 16990–17017. Accessed: Aug. 05, 2022. [Online]. Available: https://proceedings.mlr.press/v162/notin22a.html.

[100] Frazer J et al. Disease variant prediction with deep generative models of evolutionary data. Nature 2021;599(7883):91–5. https://doi.org/10.1038/s41586-021-04043-8.

[101] Castro E, Godavarthi A, Rubinfien J, Givechian K, Bhaskar D, Krishnaswamy S. Transformer-based protein generation with regularized latent space optimization. Nat Mach Intell 2022;4(10):840–51. https://doi.org/10.1038/s42256-022-00532-1.

[102] Anishchenko I et al. De novo protein design by deep network hallucination. Nature 2021;600(7889):547–52. https://doi.org/10.1038/s41586-021-04184-w.

[103] Szegedy C, et al., Going deeper with convolutions. arXiv, 2014. doi: 10.48550/arXiv.1409.4842.

[104] Tischer D et al. Design of proteins presenting discontinuous functional sites using deep learning. bioRxiv 2020. https://doi.org/10.1101/2020.11.29.402743.

[105] Norn C et al. Protein sequence design by conformational landscape optimization. Proc Natl Acad Sci 2021;118(11):. https://doi.org/10.1073/pnas.2017228118e2017228118.

[106] Wang J et al. Scaffolding protein functional sites using deep learning. Science 2022;377(6604):387–94. https://doi.org/10.1126/science.abn2100.

[107] Baek M et al. Accurate prediction of protein structures and interactions using a three-track neural network. Science 2021;373(6557):871–6. https://doi.org/10.1126/science.abj8754.

[108] Roney JP, Ovchinnikov S. State-of-the-art estimation of protein model accuracy using AlphaFold. bioRxiv 2022. https://doi.org/10.1101/2022.03.11.484043.

[109] Anand N, Achim T, Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. arXiv, 2022. doi: 10.48550/arXiv.2205.15019.

[110] Sohl-Dickstein J, Weiss EA, Maheswaranathan N, Ganguli S, Deep unsupervised learning using nonequilibrium thermodynamics. arXiv, 2015. doi: 10.48550/arXiv.1503.03585.

[111] Ho J, Jain A, Abbeel P, Denoising diffusion probabilistic models. arXiv, 2020. doi: 10.48550/arXiv.2006.11239.

[112] Song Y, Ermon S, Generative modeling by estimating gradients of the data distribution. arXiv, 2020. doi: 10.48550/arXiv.1907.05600.

[113] Ramesh A, Dhariwal P, Nichol A, Chu C, Chen M, Hierarchical text-conditional image generation with CLIP latents. arXiv, 2022. Accessed: Aug. 28, 2022. [Online]. Available: http://arxiv.org/abs/2204.06125.

[114] Olenyi T, et al., LambdaPP: Fast and accessible protein-specific phenotype predictions. bioRxiv, 2022;2022.08.04.502750. doi: 10.1101/2022.08.04.502750.

[115] Mirdita M, Schütze K, Moriwaki Y, Heo L, Ovchinnikov S, Steinegger M. ColabFold: making protein folding accessible to all. Nat Methods 2022;19(6):6. https://doi.org/10.1038/s41592-022-01488-1.

[116] van Kempen M, et al., Foldseek: fast and accurate protein structure search. bioRxiv, 2022;2022.02.07.479398. doi: 10.1101/2022.02.07.479398.

[117] Gohil N, Bhattacharjee G, Khambhati K, Braddick D, Singh V, Engineering strategies in microorganisms for the enhanced production of squalene: advances, challenges and opportunities, *Front Bioeng Biotechnol*, 2022;7. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fbioe.2019.00050.

[118] El-Gebali S et al., The Pfam protein families database in 2019, Nucleic Acids Res., vol. 47, no. D1, pp. D427–D432, Jan. 2019, doi: 10.1093/nar/gky995.

[119] Rios-Martinez C, Bhattacharya N, Amini AP, Crawford L, Yang KK, Deep self-supervised learning for biosynthetic gene cluster detection and product classification. bioRxiv, 2022:2022.07.22.500861. doi: 10.1101/2022.07.22.500861.

[120] Newman DJ, Cragg GM. Natural Products as Sources of New Drugs from 1981 to 2014. J Nat Prod 2016;79(3):629–61. https://doi.org/10.1021/acs.jnatprod.5b01055.

[121] Schreiber SL. The rise of molecular glues. Cell Jan. 2021;184(1):3–9. https://doi.org/10.1016/j.cell.2020.12.020.

[122] Yao L, Zheng Y, Zhu Z. Jasmonate suppresses seedling soil emergence in Arabidopsis thaliana. Plant Signal Behav 2017;12(6):e1330239. https://doi.org/10.1080/15592324.2017.1330239.

[123] Sievers QL et al. Defining the human C2H2 zinc finger degrome targeted by thalidomide analogs through CRBN. Science 2018;362(6414):eaat0572. https://doi.org/10.1126/science.aat0572.

[124] Fischer ES, Park E, Eck MJ, Thomä NH. SPLINTS: Small-molecule protein ligand interface stabilizers. Curr Opin Struct Biol 2016;37:115–22. https://doi.org/10.1016/j.sbi.2016.01.004.

[125] Shigdel UK et al. Genomic discovery of an evolutionarily programmed modality for small-molecule targeting of an intractable protein surface. Proc Natl Acad Sci 2020;117(29):17195–203. https://doi.org/10.1073/pnas.2006560117.

[126] Bier D et al. The molecular tweezer CLR01 stabilizes a disordered protein-protein interface. J Am Chem Soc 2017;139(45):16256–63. https://doi.org/10.1021/jacs.7b07939.

[127] Rudolph J, Settleman J, Malek S. Emerging trends in cancer drug discovery-from drugging the 'undruggable' to overcoming resistance. Cancer Discov 2021;11(4):815–21. https://doi.org/10.1158/2159-8290.CD-21-0260.

[128] Kautsar SA, et al., MIBiG 2.0: a repository for biosynthetic gene clusters of known function, Nucleic Acids Res., vol. 48, no. D1, pp. D454–D458, Jan. 2020, doi: 10.1093/nar/gkz882.

[129] Piotrowski M, Morsomme P, Boutry M, Oecking C. Complementation of the Saccharomyces cerevisiae plasma membrane H+-ATPase by a plant H+-ATPase generates a highly abundant fusicoccin binding site. J Biol Chem 1998;273(45):30018–23. https://doi.org/10.1074/jbc.273.45.30018.

[130] Jahn T et al. The 14–3-3 protein interacts directly with the C-terminal region of the plant plasma membrane H(+)-ATPase. Plant Cell 1997;9(10):1805–14. https://doi.org/10.1105/tpc.9.10.1805.

[131] Marra M, Camoni L, Visconti S, Fiorillo A, Evidente A. The surprising story of fusicoccin: A wilt-inducing phytotoxin, a tool in plant physiology and a 14-3-3-targeted drug. Biomolecules 2021;11(9):1393. https://doi.org/10.3390/biom11091393.

[132] Arnold FH. Design by directed evolution. Acc Chem Res 1998;31(3):125–31. https://doi.org/10.1021/ar960017f.

[133] Hunt AC et al. Multivalent designed proteins protect against SARS-CoV-2 variants of concern. bioRxiv 2021. https://doi.org/10.1101/2021.07.07.451375.

[134] Cirino PC, Arnold FH. Exploring the diversity of heme enzymes through directed evolution. In: Brakmann S, Johnsson K, editors. Directed molecular evolution of proteins. Weinheim, FRG: Wiley-VCH Verlag GmbH & Co. KGaA; 2002. p. 215–43. https://doi.org/10.1002/3527600647.ch10.
[135] De Crécy-lagard V et al. A roadmap for the functional annotation of protein families: a community perspective, Database Jan. 2022;2022:baac062. https://doi.org/10.1093/database/baac062.
[136] Check Hayden E. The automated lab. Nature 2014;516(7529):7529. https://doi.org/10.1038/516131a.
[137] Segal M. An operating system for the biology lab. Nature 2019;573(7775): S112–3. https://doi.org/10.1038/d41586-019-02875-z.
[138] Arnold C. Cloud labs: where robots do the research. Nature 2022;606 (7914):612–3. https://doi.org/10.1038/d41586-022-01618-x.
[139] NVIDIA Omniverse for Digital Twins, *NVIDIA*. https://www.nvidia.com/en-us/omniverse/solutions/digital-twins/ (accessed Aug. 23, 2022).
[140] Tao F, Qi Q. Make more digital twins. Nature 2019;573(7775):490–1. https://doi.org/10.1038/d41586-019-02849-1.
[141] El Saddik A. Digital twins: the convergence of multimedia technologies. IEEE Multimed 2018;25(2):87–92. https://doi.org/10.1109/MMUL.2018.023121167.
[142] Krittanawong C, The next step in deep learning-guided clinical trials, *Nat Cardiovasc Res*, 2022;1(4):4, doi: 10.1038/s44161-022-00044-6.
[143] Zhou N et al. The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. Genome Biol 2019;20(1):244. https://doi.org/10.1186/s13059-019-1835-8.
[144] The Critical Assessment of Genome Interpretation Consortium, CAGI, the Critical Assessment of Genome Interpretation, establishes progress and prospects for computational genetic variant interpretation methods. arXiv, 2022. Accessed: Aug. 28, 2022. [Online]. Available: http://arxiv.org/abs/2205.05897.
[145] Petti S, Eddy SR. Constructing benchmark test sets for biological sequence analysis using independent set algorithms. PLOS Comput. Biol. 2022;18(3):. https://doi.org/10.1371/journal.pcbi.1009492e1009492.
[146] Lorello LS, Galassi A, Torroni P, BANANA: a Benchmark for the Assessment of Neural Architectures for Nucleic Acids, 2021, Accessed: Aug. 07, 2022. [Online]. Available: https://openreview.net/forum?id=Pobz_8y2Q2_.
[147] Dallago C, et al., FLIP: Benchmark tasks in fitness landscape inference for proteins. In: Presented at the thirty-fifth conference on neural information processing systems datasets and benchmarks track (round 2), Jan. 2022. Accessed: Aug. 07, 2022. [Online]. Available: https://openreview.net/forum?id=p2dMLEwL8tF.
[148] Zhang Z, et al., Protein representation learning by geometric structure pretraining. arXiv, 2022. Accessed: Jul. 28, 2022. [Online]. Available: http://arxiv.org/abs/2203.06125.