nature machine intelligence

Article

https://doi.org/10.1038/s42256-024-00838-2

Accurate and robust protein sequence design with CarbonDesign

Received: 10 August 2023

Accepted: 10 April 2024

Published online: 23 May 2024

Check for updates

Milong Ren[1,2], Chungong Yu[1,2,3], Dongbo Bu[1,2,3] ✉ & Haicang Zhang[1,2,3] ✉

Protein sequence design is critically important for protein engineering. Despite recent advancements in deep learning-based methods, achieving accurate and robust sequence design remains a challenge. Here we present CarbonDesign, an approach that draws inspiration from successful ingredients of AlphaFold and which has been developed specifically for protein sequence design. At its core, CarbonDesign introduces Inverseformer, which learns representations from backbone structures and an amortized Markov random fields model for sequence decoding. Moreover, we incorporate other essential AlphaFold concepts into CarbonDesign: an end-to-end network recycling technique to leverage evolutionary constraints from protein language models and a multitask learning technique for generating side-chain structures alongside designed sequences. CarbonDesign outperforms other methods on independent test sets including the 15th Critical Assessment of protein Structure Prediction (CASP15) dataset, the Continuous Automated Model Evaluation (CAMEO) dataset and de novo proteins from RFDiffusion. Furthermore, it supports zero-shot prediction of the functional effects of sequence variants, making it a promising tool for applications in bioengineering.

Protein sequence design, also referred to as inverse protein folding, is to identify amino acid sequences that can fold into a given protein backbone structure while exhibiting desired functions. It serves as a crucial step in computational protein design, which has recently made significant advancements in the engineering of therapeutics[1,2], enzymes[3,4] and more applications[5]. Typically in de novo protein design, determining the optimal sequences became essential once the backbone structures are derived from either energy-based methods[6] or recent diffusion generative models[7–9].

Recent advancements in deep learning-based sequence design methods have demonstrated promising results in generating highly accurate candidate sequences[10–15]. These approaches differ from one another in their strategies for encoding the protein structure and decoding the associated sequences. Typically, ProteinMPNN[10] and ESM-IF[11] utilize neural networks to encode the entire backbone structure and subsequently decode the sequences in an end-to-end autoregressive manner. On the other hand, methods such as 3DCNN[12], ABACUS-R[13] and ProDESIGN-LE[14] individually encode the structural context of each residue and iteratively refine the designed sequences, starting from a randomly initialized sequence.

Protein structure prediction and protein sequence design are closely intertwined, with advancements in one field benefiting the other. Inspired by the remarkable success of AlaphFold[16] and RoseTTAFold[17] in addressing the protein folding problem, we adapt their key concepts to the inverse folding and propose CarbonDeisgn, aiming to improve sequence design through enhancing the encoder and decoder architecture, leveraging more efficient features and refining the training strategy.

At its core, CarbonDesign explores a network architecture called Inverseformer to transform three-dimensional structural features into single and pair representations through a series of node updates and triangular edge updates, following a Markov random field (MRF) module for sequence decoding. Intuitively, the Inverseformer inverts the information flow compared to AlphaFold's Evoformer, primarily focusing on learning representations from backbone structures.

[1]SKLP, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. [2]University of Chinese Academy of Sciences, Beijing, China. [3]Central China Institute of Artificial Intelligence, Zhengzhou, China. ✉e-mail: dbu@ict.ac.cn; zhanghaicang@ict.ac.cn

https://doi.org/10.1038/s42256-024-00838-2

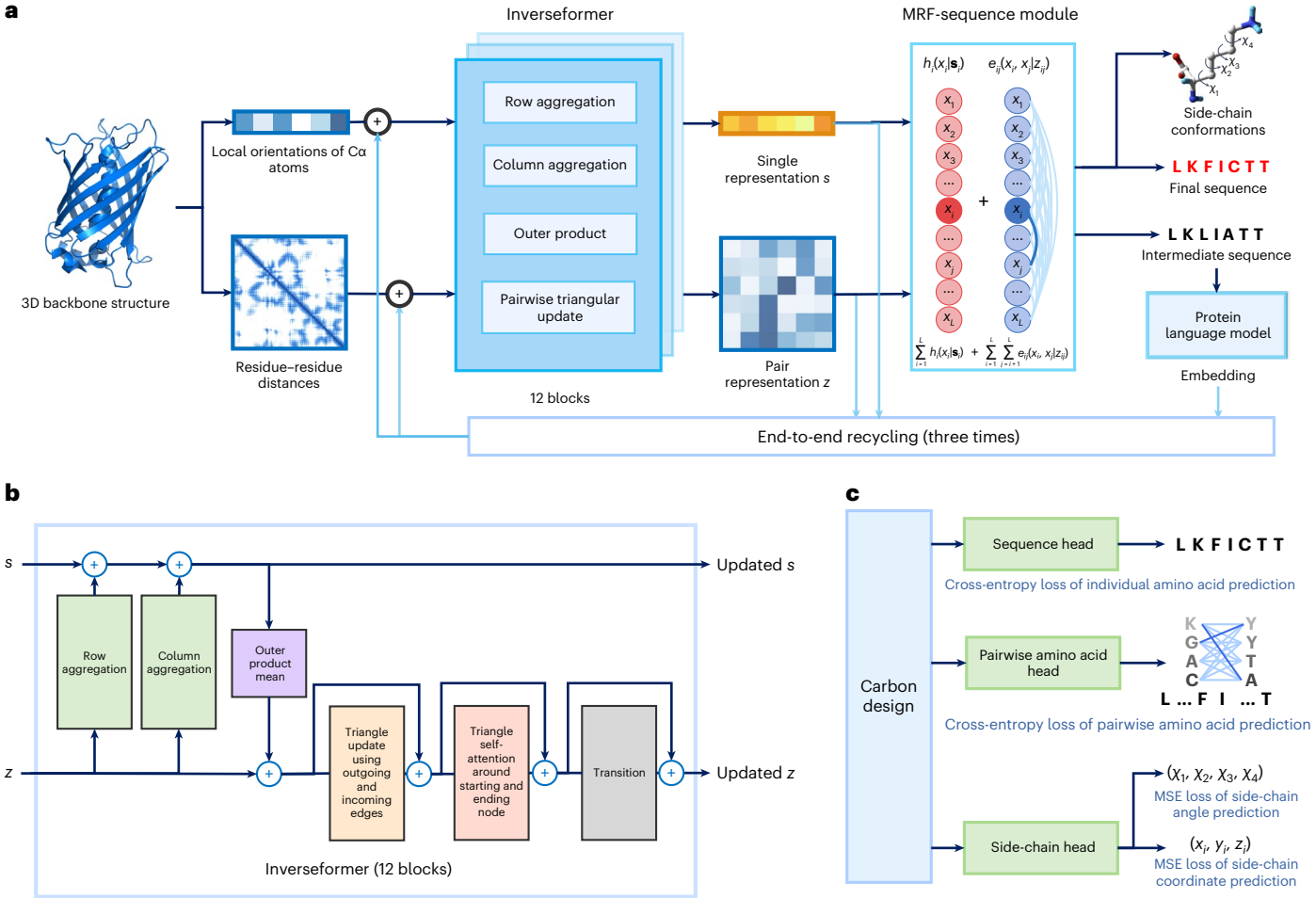

Fig. 1 | CarbonDesign architecture. a, The arrows illustrate the flow of information in the network, designing a one-dimensional protein sequence from a three-dimensional (3D) backbone structure. **b**, The Inverseformer blocks update the single and pair representations through node aggregation and triangular edge update layers. **c**, CarbonDesign employs multitask learning with various training losses, including single and pair amino acid losses and losses for side-chain structures. χ represents the side chain torsion angles.

We also introduce two other crucial concepts. First, we adopt the network recycling strategy[16,18,19] to recycle the entire network with shared weights in an end-to-end manner. During the recycling stages, we incorporate sequence embedding from the protein language model ESM2 (ref. 20), enabling CarbonDesign to fuse evolutionary and structural constraints effectively. Second, we leverage multitask learning with several auxiliary losses to directly guide the learning of single and pair representations and predict the sequences and corresponding side-chain structures.

We extensively evaluate CarbonDesign using diverse datasets, including the Continuous Automated Model EvaluatiOn (CAMEO) dataset[21], the 15th Critical Assessment of protein Structure Prediction (CASP15) dataset[22] and the predicted structures from AlphaFold. Additionally, in the context of de novo protein design, we further assess the utility of CarbonDesigin in reconstructing sequence for the de novo structures derived from diffusion generative methods such as RFdiffusion[7] and FrameDiff[8]. Furthermore, we demonstrate that CarbonDesgin serves as a reliable zero-shot predictor of mutational effects on protein function, with its performance evaluated using deep mutational scanning datasets encompassing millions of missense variants.

Results

Model architecture

CarbonDesign improves protein sequence design by incorporating Inverseformer neural network architectures and training procedures based on evolutionary and structural constraints. To convert protein three-dimensional structures to one-dimensional sequences, we invert and adapt the network architecture employed in AlphaFold, which was originally developed for three-dimensional structure prediction from one-dimensional sequences (Fig. 1 and Table 1).

The network comprises two main stages. First, we use an Inverseformer module to progressively update the single and pair representations, which are initialized with local orientations and residue–residue distances. Second, we use a Markov random fields (MRF)-sequence module to decode the sequence, with its pair coupling terms and site bias terms parameterized based on the learned pair and single representations, respectively (Methods).

Inverseformer aims to learn the single and pair representations from which single-site and pairwise amino acids can be decoded (Fig. 1b). Single and pair representations interact and undergo refinement through a series of blocks. Specifically, single representations are updated through row aggregation and column aggregation layers with pair presentations as inputs, enabling information flows from two-dimensional to one-dimensional representations. Subsequently, pairwise representations are revised through an outer product layer and four triangular attention layers.

In protein structure prediction, triangular edge updates are intuitively motivated by the need to satisfy the triangle inequality constraints on residue–residue distances. On the other hand, for sequence design, we establish an intuitive connection between

Table 1 | Key concepts of CarbonDesign inspired by AlphaFold

Method	AlphaFold	CarbonDesign
Direction of information flow	One dimension to three dimensions	Three dimensions to one dimension
Architecture	Evoformer and structure modules	Inverseformer and Markov random fields-sequence module
Additional features in recycling stage	Distance map of predicted structures	Embeddings of intermediate sequence from language model ESM2
Multitask learning	Folding head, distogram head, confidence head and so on.	Sequence head, pairwise amino acid head, side-chain head.

Inverseformer's triangular updates and the edge message updates in the Belief Propagation (BP) algorithm, which is commonly used for learning and inference in probabilistic graphical models such as MRF and Bayesian networks[23,24]. In the BP algorithm, node and edge messages are updated alternately to aggregate probability mass from neighbouring variable nodes. Each edge message *ij* is updated through a triangular edge updates operation, involving all other edge messages *jk* related to variable node *j* (Supplementary Fig. 1). Based on this intuition, we hypothesize that the triangular edge updates encourage representations that generate sequences with higher likelihoods under the MRF model in the following MRF-sequence module.

The MRF-sequence module constructs a probabilistic model for the sequences conditioned on learned single and pair representations. MRFs are widely utilized in direct coupling analysis to model sequence likelihoods[25–27]. In the context of CarbonDesign, the learned single and pair representations naturally parameterize the coupling and site bias terms in MRF. Subsequently, a simple ad hoc algorithm is used to sample the candidate amino acid sequences from the MRF model (Methods).

End-to-end network recycling with a protein language model

The end-to-end network recycling technique enhances model capacity by stacking and reusing the same model architecture with shared weights. Rather than making direct predictions in a single step, this technique employs a self-correcting mode to progressively refine an initial solution by incorporating feedback from error predictions. It has been successfully applied within the field of computer vision[18,19], as well as in AlphaFold for protein structure prediction.

Network recycling enables the model to extract additional features as error feed-backs from the intermediate predictions. In the case of CarbonDesign, learned single and pair representations from the previous recycling rounds serve as features for the next round.

Furthermore, the recycling technique enables CarbonDesign to leverage evolutionary constraints encoded in protein language models such as ESM2 in an end-to-end manner. Specifically, the intermediate sequence is first predicted using the single representations, and its embedding is extracted from the language model ESM2 as additional recycling features. Protein language models can learn efficient representations from millions of sequences and have been successfully applied in predicting protein functions and structures[20]. In the context of CarbonDesign, the language model serves as a prior for the generated sequences.

Multitask learning with sequence design

We employ a cross-entropy loss for individual amino acids and an auxiliary cross-entropy loss for pairwise amino acid identities to directly guide the learning of the single and pair representations, respectively. To approximate the exact likelihood of the sequences in the MRF model[25], we utilize a composite likelihood during training. Moreover, we incorporate a side-chain torsional angle loss and a side-chain structure loss in training[16], enabling CarbonDesign to predict both the sequences and the corresponding side-chain structures (Fig. 1c).

Evaluating CarbonDesign on independent testing sets

We extensively evaluated CarbonDesign on two prominent datasets: the CAMEO test set[21] and the CASP15 test set. We compared our approach with representative methods in protein sequence design, including ProteinMPNN[10], ESM-IF[11], ABACUS-R[13], Rosetta software[28] and ProDESIGN-LE[14].

We evaluated the performance of CarbonDesign using two key metrics: sequence recovery rate and the BLOcks SUbstitution Matrix (BLOSUM) score[29]. The sequence recovery rate assesses the model's ability to design sequences that closely match the target structure, while the BLOSUM score measures the similarity between the designed sequences and the native sequences.

On both CAMEO and CASP datasets, CarbonDesign's sequence recovery rate and the BLOSUM score metrics outperform the other comparative methods (Fig. 2a,b). Remarkably, we have observed that utilizing a larger language model, ESM-3B, leads to a further improvement in sequence design accuracy (Fig. 2e).

We further evaluated CarbonDesign using a dataset of orphan proteins characterized by limited or no homologous sequences and a lack of structure templates. These proteins pose a significant challenge for existing structure prediction methods due to the scarcity of evolutionary information[16,17,20,30,31]. They also serve as a rigorous test set for protein sequence design, as they lack homologous information in existing sequence and structure databases. In our evaluation on the orphan proteins from CASP15, CarbonDesign still demonstrated robust performance, achieving a sequence recovery rate of 49.1% and outperforming all other representative methods (Supplementary Table 6).

Recent advancements in diffusion-based methods have enabled the design of long backbone structures, which pose a challenge for protein sequence design. We curated a dataset of long proteins (>800 amino acids) from CASP15 and CAMEO test sets to evaluate CarbonDesign's performance. Notably, CarbonDesign achieved a sequence recovery rate of 55.1%, surpassing the compared methods. As an illustrative example, we evaluated CarbonDesign on the multidrug-resistant protein T1158 (Bos taurus MRP4) with a length of 1,340 amino acids (Fig. 2f and Supplementary Table 5). CarbonDesign demonstrated a sequence recovery rate of 58.1% and a template modelling (TM) score of 0.97 when comparing the predicted structure via ESMFold with the native structure.

As a case study, we examine the protein dual-wield NTPase (dwNTPase) (Fig. 2g)[32], which exhibits a highly novel architecture discovered through data mining of predicted structures in the AlphaFold DataBase[33]. CarbonDesign successfully generates a sequence with a high sequence recovery rate of 70.2%. This case highlights the robustness of CarbonDesign with predicted backbone structures and its strong model generalization, enabling accurate designs for novel fold types.

Improving de novo protein design with CarbonDesign

Recent diffusion-based methods, such as RFdiffusion, have revolutionized de novo protein design by generating novel backbone structures across diverse fold types that have never been observed in nature. In light of these advancements, we evaluate the efficacy of CarbonDesign in enhancing protein de novo design by generating more accurate sequences for these backbone structures.

Since native sequences are unavailable for evaluating sequence recovery rate and BLOSUM similarity score, we employ the self-consistency TM (scTM) score as an alternative measure. Specifically, we first utilize ESMFold to predict the structures of the designed sequences corresponding to the backbone structures generated by RFdiffusion. We then use the TM score to measure the consistency

https://doi.org/10.1038/s42256-024-00838-2

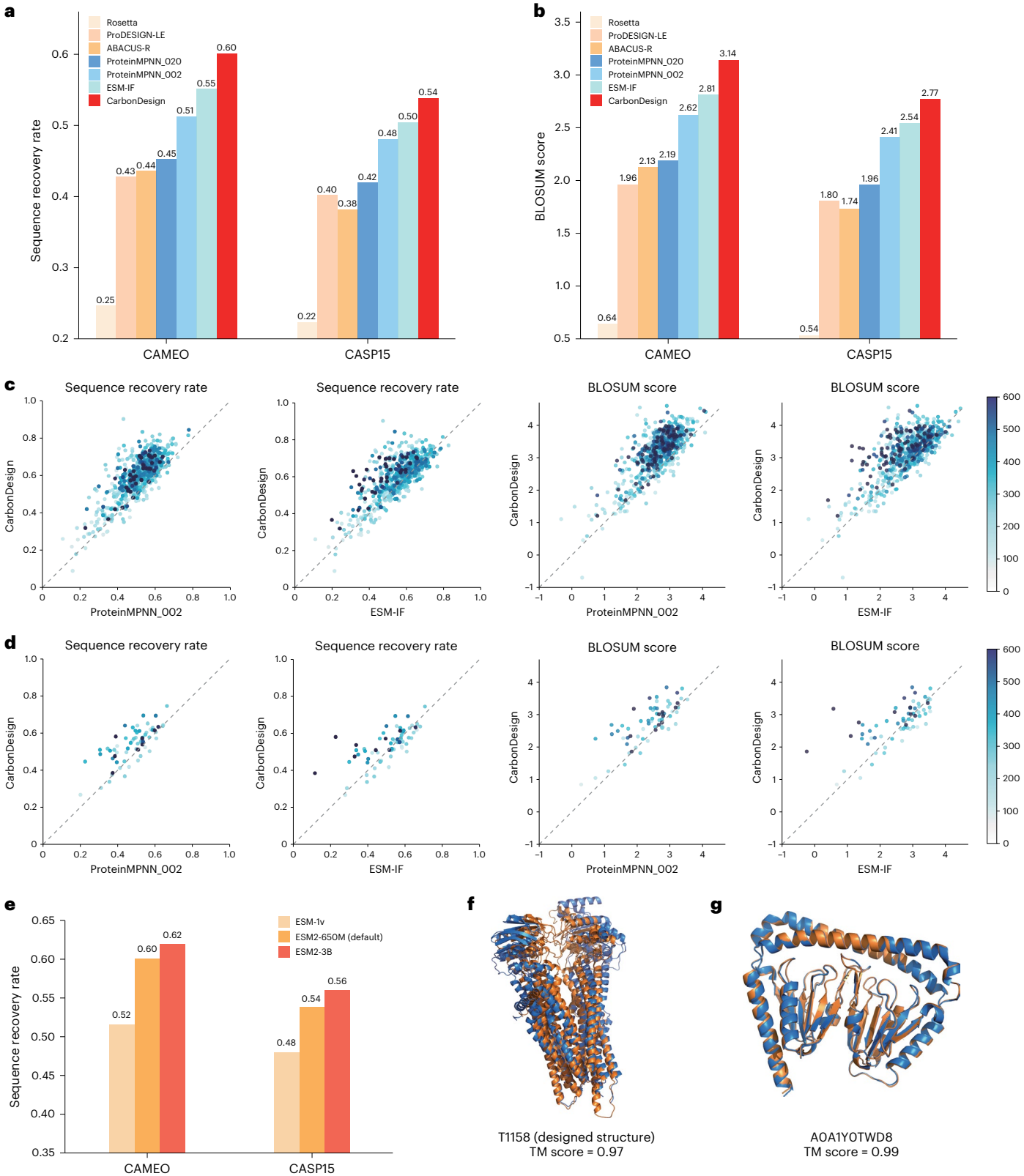

Fig. 2 | Evaluation of CarbonDesign with the CAMEO and CASP15 independent testing sets. a,**b**, Evaluation with sequence recovery rate (**a**) and BLOSUM score (**b**). **c**,**d**, Head-to-head comparisons with other representative methods on CAMEO (**c**) and CASP15 (**d**) testing sets, with colour intensity indicating sequence lengths. **e**, Evaluation of CarbonDesign with various protein language models based on sequence recovery rate. **f**, Illustrative case of a long protein T1158 (length, 1,340 amino acids) showing the native structure (blue) and the predicted structure of the designed sequence (orange). **g**, Illustrative case of the novel fold protein dwNTPase mined from AlphaFold DataBase, with the predicted structure of native sequence (blue) and designed sequence (orange).

https://doi.org/10.1038/s42256-024-00838-2

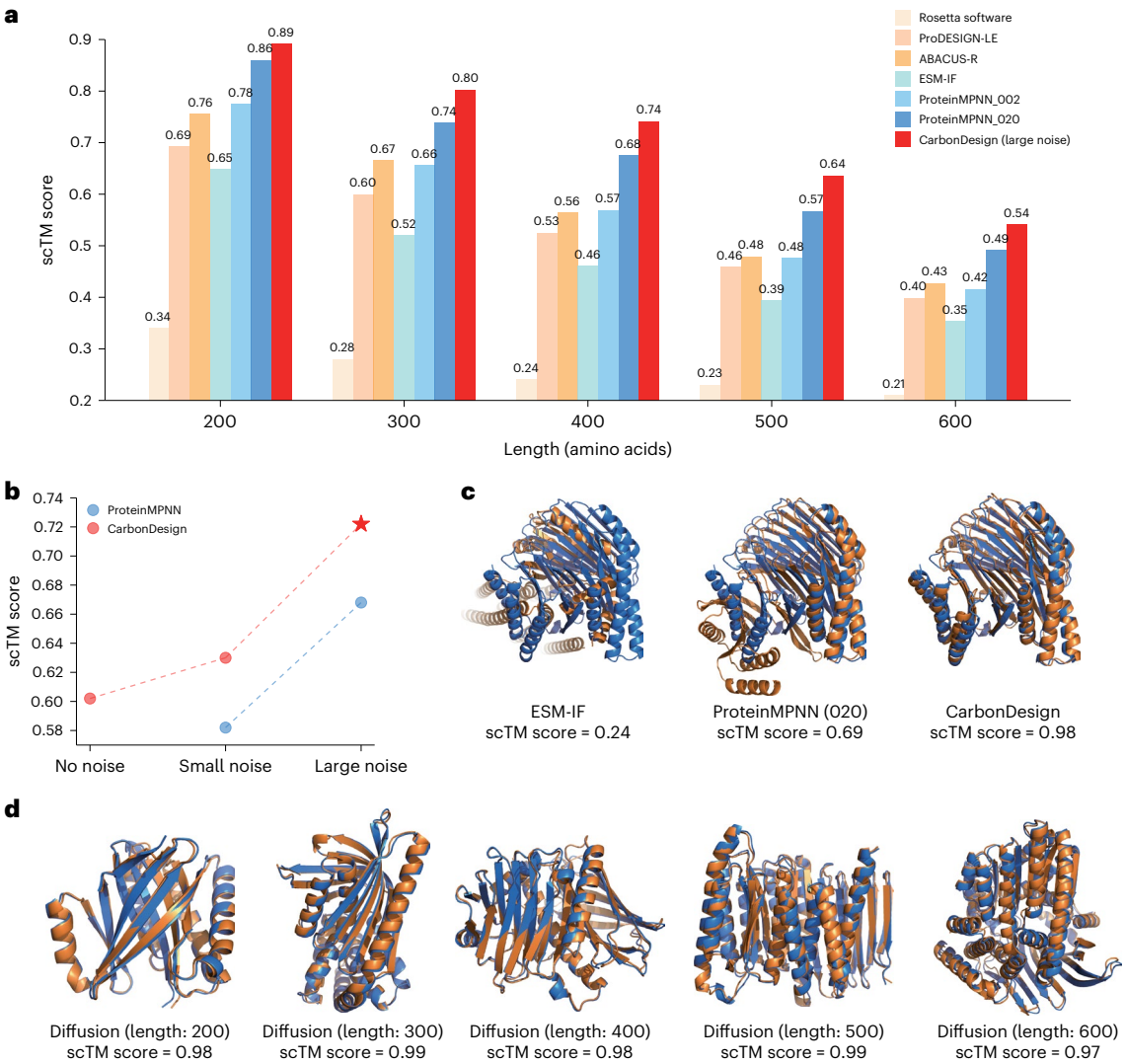

Fig. 3 | Evaluation of CarbonDesign on de novo backbone structures from RFdiffusion. a, Evaluation on backbone structures with varying lengths, measured by the scTM score. **b**, Impacts of training noise levels on the performance of CarbonDesign and ProteinMPNN. **c**, Illustrative case showing a de novo backbone structure (blue) and predicted structures (orange) of designed sequences from ESM-IF, ProteinMPNN and CarbonDesign, respectively. **d**, Additional illustrative cases of de novo backbone structures (blue) with varying lengths and predicted structures (orange) of designed sequences from CarbonDesign.

between predicted and original structures. We also note that while the scTM score is commonly used as a surrogate when native sequences and crystal structures are unavailable, its reliability is contingent upon the accuracy of protein structure prediction.

Following ProteinMPNN and ESM-IF, we introduced noise into the crystal structures during training. This approach accounts for the fact that in practical applications, de novo-generated structures or predicted structures may not exhibit the same level of precision as crystal structures commonly used in training. We generated 2,560 backbone structures of variable lengths (ranging from 200 to 600 amino acids) using RFdiffusion and evaluated the performance of CarbonDesign and ProteinMPNN with different noise levels.

Our results highlight two main findings. First, CarbonDesign consistently outperforms ProteinMPNN in terms of the scTM score at each noise level (Fig. 3b). Second, we observed that higher noise levels improve the performance of both CarbonDesign and ProteinMPNN, indicating the beneficial role of noise in generating sequences for de novo structures. More specifically, CarbonDesign demonstrates superior performance over the existing representative methods, including ProteinMPNN and ESM-IF, across all different lengths (Fig. 3a).

To assess the broad applicability of CarbonDesign in enhancing protein de novo design, we extend our evaluation to include FrameDiff, another recent diffusion-based method. CarbonDesign still outperforms all other comparison methods. (Supplementary Fig. 3), demonstrating the efficacy of CarbonDesign in enhancing the performance of FrameDiff.

Moreover, we present a successful example of a generated backbone structure consisting of 500 residues. CarbonDesign achieves a scTM score of 0.98, which is significantly higher than ESM-IF (scTM = 0.24) and ProteinMPNN (scTM = 0.69) (Fig. 3c). Furthermore, we demonstrate other successful examples of designed sequences of variable lengths (Fig. 3d).

To generate a group of diverse sequences for a given backbone structure in various downstream design tasks, we introduce the temperature parameter T during inference to control the diversity of the designed sequences (Methods). As T increases, CarbonDesign can sample a group of more diverse sequences (Supplementary Tables 10 and 11). Additionally, we observe that a more constrained structural context leads to a decrease in residue-level diversity of the designed sequences (Supplementary Fig. 6).

Predicting functional effects of variants via CarbonDesign

The accurate interpretation of the functional effects of variants is crucial in directed evolution-based protein engineering[34,35], as well as in the context of human genetic studies and clinical testing[36,37]. Pretrained language models have emerged as effective zero-shot predictors, alleviating the issue of limited labelled data and mitigating potential human biases in variant annotation[38]. We now show that CarbonDesign also supports zero-shot learning for functional effects prediction, indicating its ability to capture the inherent sequence–structure–function relations.

We first use AlphaFold to predict the protein structures for the testing sequences, which serve as inputs of CarbonDesign. Subsequently, to score the mutational effects of variants on a particular sequence, we calculate the ratio between the likelihoods of the mutated and wild-type sequences based on the CarbonDesign model (Methods).

We evaluate CarbonDesign on deep mutational scanning datasets with experimentally determined functional scores[39]. CarbonDesign achieves a Spearman correlation of 0.43, outperforming pure language model-based approaches including ESM-1v and ProGen2 (Fig. 4a). Furthermore, integrating the scores of CarbonDesign and the other two methods improves the performance, resulting in a Spearman correlation of 0.47. This highlights that CarbonDesign, as a structure-based method, can improve the interpretation of functional effects in combination with purely language model-based methods.

We also compared CarbonDesign with multiple sequence alignment (MSA)-based methods, such as EVE[37] and MSA-Transformer[40], as well as the ensemble methods (Supplementary Table 14). Notably, when combined with the MSA-based method EVE, CarbonDesign achieved a Spearman correlation of 0.50, surpassing the current leading ensemble method EVE+Tranception (MSA retrieval)[39]. This observation suggests that CarbonDesign, integrating structural information, can also enhance the performance of MSA-based methods in variant effects prediction.

We next assess CarbonDesign in predicting the pathogenicity of human genetic variants. Specifically, we focus on four well-known disease risk genes (*BRCA1* (ref. 41), *TP53* (ref. 42), *PTEN* (ref. 43) and *MSH2* (ref. 44)) that have a substantial number of high-quality clinical labels in ClinVar. CarbonDesign achieves good separation of benign and pathogenic variants for *TP53* and *PTEN*, with area under the receiver operating characteristic curve values exceeding 0.95 (Fig. 4b). Additionally, CarbonDesign outperforms pure language model-based approaches on average in this context (Supplementary Table 7).

Furthermore, we observed a correlation between the predicted amino acid distribution and the protein structures. We utilize the entropy of the predicted amino acid distribution as a metric of conservation, with lower entropy indicating higher conservation. As a proof of concept, we examine two proteins, Nav1.4–β_1 (ref. 45) (Fig. 4c) and indole-3-glycerol phosphate synthase[46] (Fig. 4d). In both cases, regions with lower entropy coincide with hydrophobic core regions associated with functional regions such as the sodium channel and phosphate binding sites.

Interpreting the CarbonDesign

We trained and evaluated several ablation models to evaluate the relative contributions of the key architecture to CarbonDesign accuracy.

CarbonDesign utilizes the side-chain head to generate side-chain structures of all possible amino acids at each position. We evaluated the prediction accuracy of side chains using the CAMEO and CASP15 datasets and investigated the contribution of side-chain heads for sequence design accuracy.

CarbonDesign achieves an average root mean squared distance (RMSD) of 0.805. Moreover, the side-chain prediction accuracy strongly correlates with the structural context constraints, measured by the number of Cβ atoms within an 8 Å radius around each residue. Higher side-chain prediction accuracy was observed for more constrained residues (Fig. 5a). For example, the side-chain head of CarbonDesign demonstrated higher prediction accuracy with an RMSD of 0.683 for the protein T1159 (PDB ID, 7PTZ (ref. 47)) (Fig. 5c).

There also exhibits a strong correlation between the side-chain prediction accuracy and sequence design accuracy, with a Pearson correlation of 0.73 (Fig. 5b). The more constrained structural context leads to improved prediction accuracy for both side-chain prediction and sequence design tasks, consistent with prior studies[8,10]. Additionally, training a modified model with the side-chain head removed demonstrates the beneficial effect of the side-chain head in enhancing the accuracy of designed sequences (Supplementary Table 1).

Network recycling allows the model to incorporate the protein language model in an end-to-end manner. We further assess the contribution of network recycling and the additional sequence embedding from the language models during the recycling stages. Increasing the number of recycling iterations results in an improved sequence recovery rate of designed sequences (Fig. 5d).

Additionally, network recycling and the protein language model enhance de novo protein design evaluated on the backbone structures from the diffusion generative model (Fig. 5e). We also investigate the ablation model of CarbonDesign without using pretrained language models for the task of predicting variant effects, and we observe that the language models can enhance the performance (Supplementary Tables 15 and 16).

We next explore the accuracy of sequence design at protein core and surface regions. CarbonDesign demonstrates notably higher accuracy at core regions compared to surface regions (Supplementary Fig. 4), in line with previous research[10].

The pair amino acid head in CarbonDesign directly guides the learning of pair representations in Inverseformer and the pair couplings term in the MRF-sequence module. We trained a modified model excluding the pair head to evaluate its contribution. Notably, the pair head significantly improves performance for both crystal structures (Fig. 5h) and de novo structures (Fig. 5i). Furthermore, we investigated the differences between the amino acid distribution in the designed and native sequences, measured as Kullback–Leibler (KL) divergence. The model with the pair head can generate sequences with a closer amino acid distribution to the native sequences (Fig. 5f and Supplementary Fig. 5). We also observed a slight improvement in predicting the functional effects of the variants with the deep mutational scanning (DMS) testing dataset (Fig. 5g). These findings underscore the efficacy of the pair head in CarbonDesign.

Discussion

We present CarbonDesign, an approach for protein sequence design that incorporates key concepts from recent successful methods in protein structure prediction. Specifically, CarbonDesign utilizes the inverseformer architecture, network recycling technique and multitask learning strategy to enhance sequence design. Our results demonstrate that CarbonDesign outperforms existing methods in generating candidate sequences for crystal structures, predicted structures and de novo structures derived from diffusion generative models, showing its utility in the de novo protein design scenario. Moreover, CarbonDesign supports zero-shot learning for predicting the functional effects of sequence variants, highlighting its ability to capture the intrinsic relationships between protein sequences and their functions.

We utilize diverse metrics, including the sequence recovery rate, BLOSUM score, scTM score and Rosetta energy, to assess the quality of the designed sequences. The choice of computational metrics varies depending on the nature of the testing sets. For crystal structures with known sequences such as the independent testing set of CASP15 and CAMEO, we use more exact metrics including the sequence recovery rate and BLOSUM similarity score[10]. For de novo backbone structures generated from RFDiffusion or other computational methods in practical applications, where the true sequences are unknown, the scTM score acts as a proxy measure, assessing the deviation between the provided backbone structures and the predicted structures of the

https://doi.org/10.1038/s42256-024-00838-2

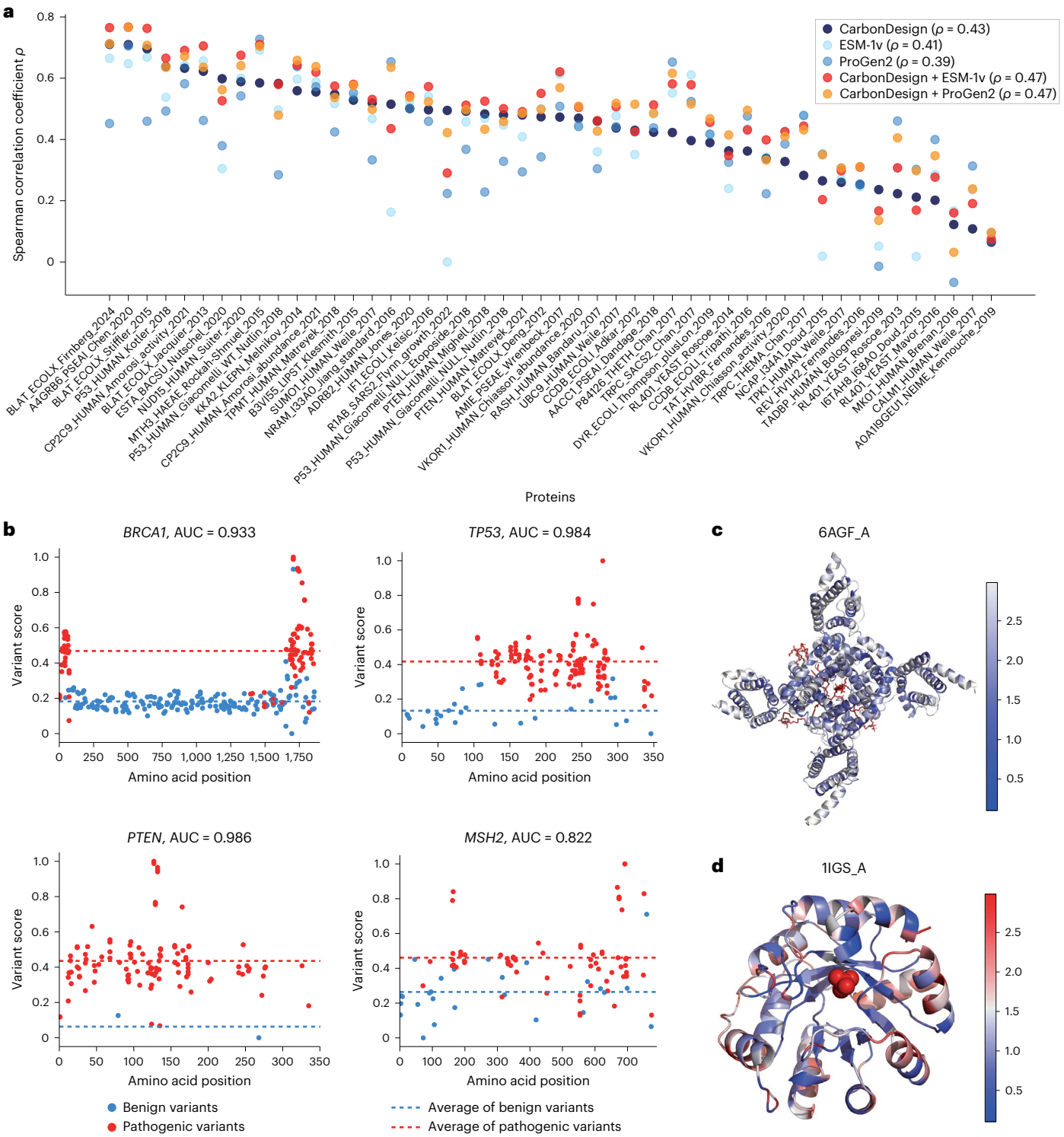

Fig. 4 | Evaluation of CarbonDesgin in interpreting functional effects of variants. a, Evaluation on variants from 49 deep mutational scanning essays. The *x* axis represents the names of the proteins in the essays and the *y* axis represents the Spearman correlation coefficient. **b**, Evaluation on clinical labelled variants in ClinVar for four well-known disease risk genes, *BRCA1*, *TP53*, *PTEN* and *MSH2*. The *x* axis represents the positions of variants on the proteins and the *y* axis represents the functional scores predicted by CarbonDesign. **c**, Entropy variation of protein Nav1.4–β_1, with each position colour-coded based on the level of entropy. Blue regions indicate areas of low entropy, white regions indicate areas of high entropy and red indicates other binding peptides. **d**, Entropy variation of protein indole-3-glycerol phosphate synthase, with each position colour-coded based on the level of entropy. Blue indicates areas of low entropy, red indicates areas of high entropy and red ions represent phosphate ions. AUC, the area under the receiver operating characteristic curve.

designed sequences[8,15,48]. We note that this proxy is limited by the capacity of the protein structure prediction methods. While recent deep learning-based methods such as CarbonDesign, ProteinMPNN[10] and ABCUS-R (ref. 13) have significantly improved protein sequence design, capable of generating more exact sequences in a high-throughput manner, classical energy-based methods such as Rosetta software have their distinct advantages. For example, Rosetta software exhibits a remarkable performance in terms of Rosetta energy[49], outperforming

https://doi.org/10.1038/s42256-024-00838-2

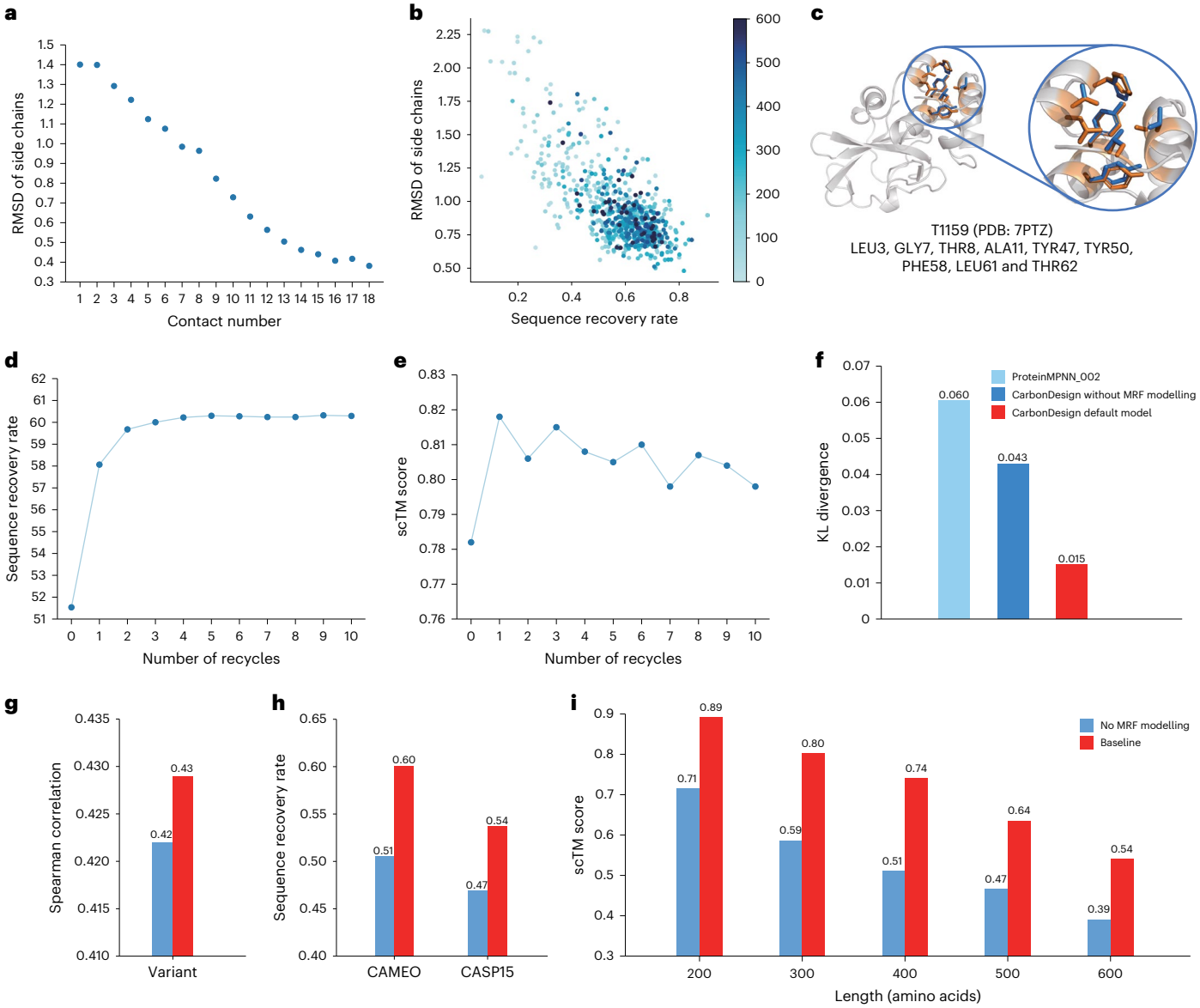

Fig. 5 | Evaluation of ablation models of CarbonDesign. a, Correlation between RMSD error of side-chain structure prediction and the number of Cβ atoms within an 8 Å radius around each residue. **b**, Correlation between sequence design accuracy and side-chain structure prediction accuracy on CAMEO and CASP15 datasets. The *x* axis represents the sequence recovery rate and the *y* axis represents the RMSD between predicted and native side-chain structures. **c**, Illustrative case of protein T1159 with predicted side-chain structures. Positions of LEU3, GLY7, THR8, ALA11, TYR47, TYR50, PHE58, LEU61 and THR62 are shown, with predicted structures in orange and native structures in blue. **d**, Evaluation of CarbonDesign with varying recycling times, measured by sequence recovery rate on CAMEO and CASP15 testing sets. **e**, Evaluation of CarbonDesign with varying recycling times, measured by scTM score on the backbone structures from RFdiffusion. **f**, KL divergence of the amino acid distribution between designed sequences and the sequences from CAMEO and CASP15 datasets. **g**–**i**, Evaluation of the effects of pair head in MRF modelling on performance in deep mutational scanning testing set (**g**), CAMEO and CASP15 testing sets (**h**), and de novo backbone structures from RFdiffusion (**i**). Blue represents the default CarbonDesign model and red represents the model with the pair head in the MRF model excluded.

all other deep learning-based methods. They do not rely on extensive training data, thereby avoiding biases introduced by training data.

CarbonDesign can leverage evolutionary constraints from large-scale pretrained protein language models. Several previous studies have also demonstrated the utility of language models in various computational protein design scenarios. For example, ProGen2 employs a generative pretrained transformer model to generate sequences with control tags specifying protein properties[50]. Ref. 51 utilizes general protein language models to efficiently evolve human antibodies, leading to a substantial improvement in antibody binding affinity. Our CarbonDesign adopts the network recycling technique to seamlessly integrate language models into structure-based protein design in an end-to-end manner.

Our work is limited in focusing solely on the in silico evaluation of the designed sequences. While in silico metrics provide empirical evidence of whether the designed sequences can fold correctly and exhibit the desired function and are commonly used in the existing methods[10,11,15], wet-lab experimental validation is crucial for a comprehensive evaluation of CarbonDesign. It could offer valuable insights and opportunities for improvement and remains our main future work.

Methods

Evaluation datasets

CAMEO testing set. We compiled a test set of 728 proteins from the recent CAMEO campaign (between February 2022 and February 2023).

After excluding short proteins with fewer than 80 amino acids, the final test set consisted of 642 proteins.

CASP15 testing set. We included all available proteins from CASP15 that were not cancelled and excluded proteins with lengths less than 80 amino acids, resulting in a test set of 65 proteins (Supplementary Table 2).

Testing set of long proteins. To benchmark the performance of long proteins, we collected proteins with more than 800 amino acids from the CAMEO and CASP15 testing sets. We then used MMseqs to filter overlaps between the two sets with a sequence identity of 40% and selected the representative protein from each cluster. The final test set comprises 13 proteins, with an average length of 1,239 amino acids (Supplementary Table 3).

Testing set of orphan proteins. We curated a testing set of nine orphan proteins from the CASP15 set. Following the criteria of orphan proteins in previous work[20], we first perform the standard AlphFold MSA search process against UniProt Reference Clusters (UniRef)[52], MGnify[53] and BFD[54] databases using HHBlits[55] and Jackhmmer[56]. Our selection was ultimately narrowed down to proteins that have fewer than 100 homologous sequences and failed to produce a template with a TM score surpassing 0.5 (Supplementary Table 4).

De novo backbone structures. We employ RFdiffusion to generate de novo backbone structures with variable lengths (200, 300, 400, 500 and 600 amino acids), producing 512 structures for each length. We also utilize FrameDiff to generate another set of de novo backbone structures.

Deep mutational scanning dataset. To evaluate CarbonDesign's efficacy in predicting the functional effects of variants, we compiled the experimentally validated variants from DMS essays. For the proteins lacking solved crystal structures or with incomplete structures, we use AlphaFold to predict their structures as inputs for CarbonDesign. Due to the limited prediction accuracy of AlaphaFold and other prediction methods for long protein sequences, and the substantial computational resources required, we restrict our analysis to proteins with fewer than 600 amino acids from the ProteinGym DMS dataset. The final testing dataset consists of 179,023 variants on 49 genes.

Genetic variants on disease genes. To access CarbonDesign's performance in prioritizing human disease-related variants, we collected the clinically labelled variants from the ClinVar database for four well-studied disease risk genes: *TP53*, *PTEN*, *BRCA1* and *MSH2*. Each variant in this dataset is annotated as either pathogenic or benign. This data includes 118 pathogenic (positives) and 175 benign (negatives) variants for *BRCA1*, 111 positives and two negatives for *PTEN*, 130 positives and 33 negatives for *TP53*, and 69 positives and 31 negatives for *MSH2*, respectively.

Training dataset

We trained CarbonDesign on protein chains in the Protein Data Bank (PDB) released before 1 January 2020, determined by X-ray crystallography or cryogenic electron microscopy. We only include the structures with a resolution better than 5.0 Å and with more than 50 amino acids. Sequences were clustered at 40% sequence identity cutoff using MMseqs2, resulting in 30,828 clusters.

Input features

CarbonDesgin incorporates inter-residue distances as edge features and local orientations of four consecutive Cα atoms as node features.

Edge features. Following ProteinMPNN[10], we calculate the distances between N, Cα, C and O atoms, and virtual Cβ atoms for each residue pair. We then divide the distances from 0 Å to 15 Å into 20 bins. The bin indices are then one-hot encoded and mapped through a feed-forward layer to initialize the pair representations. We note that we mask all the edges whose distance exceeds 15 Å. Additionally, following AlphaFold, we incorporate relative positional encoding for edge features.

Node features. For each residue at position i, we employ the Gram–Schmidt process to calculate the local frame defined by the C_α^{i-2}, C_α^{i-1} and C_α^{i} atoms. In this frame, C_α^{i} serves as the origin, the direction of C_α^{i-1} as the x axis, and C_α^{i-2} determines the x–y plane. Specifically, its basis [**a**, **b**, **c**] is obtained as follows:

$$\begin{aligned}\mathbf{a} &= \frac{C_\alpha^{i-1}-C_\alpha^{i}}{\left\|C_\alpha^{i-1}-C_\alpha^{i}\right\|}\\ \mathbf{b} &= \frac{\mathbf{a}\times\left(C_\alpha^{i-2}-C_\alpha^{i}\right)}{\left\|\mathbf{a}\times\left(C_\alpha^{i-2}-C_\alpha^{i}\right)\right\|}\\ \mathbf{c} &= \mathbf{a}\times\mathbf{b}\end{aligned} \tag{1}$$

Subsequently, the orientation of C_α^{i+1} is represented using its local coordinate with respect to this frame (Supplementary Fig. 2). Similarly, we calculate the local orientation of C_α^{i-1} with respect to the C_α^{i}, C_α^{i+1} and C_α^{i+2} atoms.

Inverseformer architecture

We utilize a series of Inverseformer blocks to learn representations from the input backbone structures (Algorithm 1). Each block has a single representation $\mathbf{s}_i$ of nodes and a pair representation $\mathbf{z}_{ij}$ of edges as its input and output and processes them through several layers.

We leverage row and column aggregation layers to update the single representations from the pair representations (equation (2)). We note that the aggregation layers are specifically tailored to incorporate edge information directly. The original row and column attention layers in the AlaphaFold Evformer architecture are unsuitable for our purpose, as they primarily focus on aggregating information on nodes, with only a bias on edges.

$$\begin{aligned}\mathbf{s}_i &\leftarrow \mathbf{s}_i + \left(\sum_{i=0}^{L}\mathrm{Transition}(\mathbf{z}_{ij})\right)^{T}\\ \mathbf{s}_i &\leftarrow \mathbf{s}_i + \left(\sum_{j=0}^{L}\mathrm{Transition}(\mathbf{z}_{ij})\right)\end{aligned} \tag{2}$$

We adopt a similar approach as AlphaFold for updating pair representations. We use an 'Outer product mean' block to integrate the single representations, followed by triangular update blocks. Furthermore, we introduce residual connections and dropout layers to prevent overfitting.

The final Inverseformer block produces a highly processed single representation $\mathbf{s}_i$ for individual residues and a pair representation $\mathbf{z}_{ij}$ for residue–residue pairs, which contain the necessary information for the MRF-sequence module to decode the sequences. These representations are crucial for accurately predicting the protein sequences.

Algorithm 1: Inverseformer

function INVERSEFORMER($\mathbf{s}_i$, $\mathbf{z}_{ij}$)
  $\mathbf{s}_i \leftarrow \mathbf{s}_i$ + Dropout(RowAgreggation($\mathbf{z}_{ij}$)) ▷ Node update
  $\mathbf{s}_i \leftarrow \mathbf{s}_i$ + Dropout(ColumnAgreggation($\mathbf{z}_{ij}$))
  $\mathbf{z}_{ij} \leftarrow \mathbf{z}_{ij}$ + OuterProductMean($\mathbf{s}_i$) ▷ Communication
  $\mathbf{z}_{ij} \leftarrow \mathbf{z}_{ij}$ + Dropout(TriangularMultiplicativeOutgoing($\mathbf{z}_{ij}$))
  ▷ Edge update
  $\mathbf{z}_{ij} \leftarrow \mathbf{z}_{ij}$ + Dropout(TriangularMultiplicativeIncoming($\mathbf{z}_{ij}$))
  $\mathbf{z}_{ij} \leftarrow \mathbf{z}_{ij}$ + Dropout(TriangleAttentionStartingNode($\mathbf{z}_{ij}$))
  $\mathbf{z}_{ij} \leftarrow \mathbf{z}_{ij}$ + Dropout(TriangleAttentionEndingNode($\mathbf{z}_{ij}$))
  $\mathbf{z}_{ij} \leftarrow \mathbf{z}_{ij}$ + Dropout(PairTransition($\mathbf{z}_{ij}$))
  return $\mathbf{s}_i$, $\mathbf{z}_{ij}$
end function

MRF-sequence module

We employ an MRF(Markov random field)-sequence module to decode the sequence from the learned representations. We denote a protein sequence of length L as $\mathbf{x}$ and the type of the i-th amino acid as x_i. And we use the random variable $\mathbf{X}$ to denote the predicted amino acid sequence.

MRFs have proven effective in modelling the distributions of sequences within a protein family[26,27]. In CarbonDesign, we adopt an amortized MRF model to describe the distribution of the designed sequences (equation (3)), which is conditioned on the learned single representations $\mathbf{s}$ and pair representations $\mathbf{z}$:

$$\mathrm{P}(\mathbf{X}=\mathbf{x}\mid\mathbf{s},\mathbf{z})=\frac{1}{Z}\exp\left[\sum_{i=1}^{L}h_i(x_i\mid\mathbf{s}_i)+\sum_{i=1}^{L}\sum_{j=i+1}^{L}e_{ij}(x_i,x_j\mid\mathbf{z}_{ij})\right] \tag{3}$$

Here, h_i and e_{ij} are the conversation bias term and pairwise coupling term, respectively, in the vanilla MRF model, and Z is the partition function. For CarbonDesign, we employ a feed-forward layer to project the learned single representation $\mathbf{s}_i$ and pair representation $\mathbf{z}_{ij}$ to h_i and e_{ij}, respectively. The training and inference of the MRF model are interconnected with other modules in CarbonDesign and will be elaborated on in the subsequent sections.

Training losses

The network is trained end-to-end, with gradients coming from the losses for reconstructing native sequences and predicting side-chain atomic coordinates. The total per-example loss can be defined as follows:

$$\mathcal{L}=\mathcal{L}_{\text{single}}+\mathcal{L}_{\text{pair}}+0.2\,\mathcal{L}_{\text{sidechain}} \tag{4}$$

To restore native sequences, we utilize single cross-entropy loss $\mathcal{L}_{\text{single}}$ and pairwise cross-entropy loss $\mathcal{L}_{\text{pair}}$ as direct supervision for the conservation bias term $h_i(x_i\mid\mathbf{s}_i)$ and the pairwise coupling term $e_{ij}(x_i,x_j\mid\mathbf{z}_{ij})$, respectively. To calculate $\mathcal{L}_{\text{single}}$, we linearly project the single representations $\mathbf{s}_i$ to obtain logits and then compute the cross-entropy loss using the native sequence as labels. For $\mathcal{L}_{\text{pair}}$, we use a pairwise pseudo-likelihood (equation (5)) to approximate the full likelihood of the sequence under the MRF model, following our previous work on residue–residue contacts prediction[25]. For each pair of amino acids in the sequence, its pseudo-likelihood conditioned on other amino acids is given by:

$$\begin{aligned}\mathcal{L}_{\text{pseudo}}(x_i,x_j)&=\log\mathrm{P}\left(X_i=x_i,X_j=x_j\mid X_{\neg\{i,j\}}=x_{\neg\{i,j\}};\mathbf{s},z\right)\\&=\log\frac{1}{Z_{ij}}\exp\Big\{h_i\left(x_i\mid\mathbf{s}_i\right)+h_j\left(x_j\mid\mathbf{s}_j\right)+e_{ij}\left(x_i,x_j\mid\mathbf{z}_{ij}\right)\\&\quad+\sum_{k\notin\{i,j\}}\left[e_{ik}\left(x_i,x_k\mid\mathbf{z}_{ik}\right)+e_{jk}\left(x_j,x_k\mid\mathbf{z}_{jk}\right)\right]\Big\}\end{aligned} \tag{5}$$

Here, Z_{ij} is the local partial function. This pseudo-likelihood produces the predicted distribution of amino acid pairs, and $\mathcal{L}_{\text{pair}}$ is computed with pairwise amino acid identities as the labels. We note that $\mathcal{L}_{\text{pair}}$ can directly supervise e_{ij} in the MRF-sequence module and pair representation $\mathbf{z}_{ij}$ in the inverseformer. Additionally, we added a 0.01 factor of L1 and L2 regularization terms into $\mathcal{L}_{\text{pair}}$.

The side-chain loss consists of three components:

$$\mathcal{L}_{\text{sidechain}}=\mathcal{L}_{\text{mse}}+\mathcal{L}_{\text{torsion}}+0.01\,\mathcal{L}_{\text{anglenorm}} \tag{6}$$

$\mathcal{L}_{\text{mse}}$ is the mean squared error (MSE) for predicted side-chain atomic coordinates. Additionally, following AlphaFold, we incorporate the loss terms $\mathcal{L}_{\text{torsion}}$ and $\mathcal{L}_{\text{anglenorm}}$ to evaluate the error of side-chain torsion angles[16].

Additional training details

For training, we utilize the Adam[57] optimizer with a β_1 value of 0.9 and a β_2 value of 0.99, where β_1 and β_2 represent coefficients used for computing running averages of the gradient and its square, respectively. The base learning rate is set to 3×10^{-4} with a warm-up period of 1,000 steps, starting from 1×10^{-5}, and the training proceeds for an additional 20,000 steps. We randomly crop very long proteins during training with a crop size of 400. The network architecture and training pipeline are implemented in PyTorch[58], and training is performed on 16 NVIDIA A40 Graphics Processing Units.

We trained several ablation models to assess the contributions of different mechanisms utilized in CarbonDesign. Following ProteinMPNN[10] and ESM-IF[15], we add noises to structures during training to deal with noises in de novo and predicted backbone structures in practical applications. In the default CarbonDesin model, we added a 0.2 Å noise to half of the training samples (referred to as small noise). To further investigate the effects of noise levels on the performance with de novo backbone structures, we trained two additional models: one without any noise (referred to as no noise), and another with a 0.2 Å noise applied to all training samples (referred to as large noise). For more details on other ablation studies, please refer to Supplementary Table 9.

Score for predicting functional effects of variants

In CarbonDesign, each variant is scored using the log odds ratio between the mutated and wild-type sequences. The variant score is defined as:

$$\text{variant score}=\frac{\mathrm{P}(\mathbf{X}=\boldsymbol{x}^{\text{mt}}\mid\mathbf{s},\mathbf{z})}{\mathrm{P}(\mathbf{X}=\boldsymbol{x}^{\text{wt}}\mid\mathbf{s},\mathbf{z})} \tag{7}$$

Here, $\mathrm{P}(\mathbf{X}=\mathbf{x}^{\text{mt}}\mid\mathbf{s},\mathbf{z})$ and $\mathrm{P}(\mathbf{X}=\mathbf{x}^{\text{wt}}\mid\mathbf{s},\mathbf{z})$ represents likelihood of the mutated (mt) and wild-type (wt) sequence, respectively, under the amortized MRF model (equation (3)).

Reporting summary

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

Data availability

The training data were obtained from the PDB website (http://www.rcsb.org/). The testing sets were acquired from CASP15 (https://predictioncenter.org/casp15/) and CAMEO (https://www.cameo3d.org). Other datasets supporting the findings of this study are available in the paper and the Supplementary Information. Source data are provided with this paper.

Code availability

The CarbonDesign software is available on both GitHub (https://github.com/zhanghaicang/carbonmatrix_public) and Code Ocean (https://codeocean.com/capsule/5915382/tree)[59].

References

1. Cao, L. et al. De novo design of picomolar SARS-CoV-2 miniprotein inhibitors. *Science* **370**, 426–431 (2020).
2. Bryan, C. M. et al. Computational design of a synthetic PD-1 agonist. *Proc. Natl Acad. Sci. USA* **118**, 2102164118 (2021).
3. Yeh, A. H.-W. et al. De novo design of luciferases using deep learning. *Nature* **614**, 774–780 (2023).
4. Dou, J. et al. De novo design of a fluorescence-activating beta-barrel. *Nature* **561**, 485–491 (2018).
5. Vorobieva, A. A. et al. De novo design of transmembrane beta barrels. *Science* **371**, 8182 (2021).
6. Kuhlman, B. et al. Design of a novel globular protein fold with atomic-level accuracy. *Science* **302**, 1364–1368 (2003).

7. Watson, J. L. et al. De novo design of protein structure and function with RFdiffusion. *Nature* https://doi.org/10.1038/s41586-023-06415-8 (2023).
8. Yim, J. et al. SE(3) diffusion model with application to protein backbone generation. In *Proc. of the 40th International Conference on Machine Learning* (eds Krause, A. et al.) 40001–40039 (PMLR, 2023).
9. Ingraham, J. et al. Illuminating protein space with a programmable generative model. *Nature* **623**, 1070–1078 (2023).
10. Dauparas, J. et al. Robust deep learning-based protein sequence design using ProteinMPNN. *Science* **378**, 49–56 (2022).
11. Hsu, C. et al. Learning inverse folding from millions of predicted structures. In *Proc. of the 39th International Conference on Machine Learning* (eds Chaudhuri, K. et al.) 8946–8970 (PMLR, 2022).
12. Anand, N. et al. Protein sequence design with a learned potential. *Nat. Commun.* **13**, 746 (2022).
13. Liu, Y. et al. Rotamer-free protein sequence design based on deep learning and self-consistency. *Nat. Comput. Sci.* **2**, 451–462 (2022).
14. Huang, B. et al. Accurate and efficient protein sequence design through learning concise local environment of residues. *Bioinformatics* **39**, 122 (2023).
15. Ingraham, J. et al. Generative models for graph-based protein design. In *Proc. of Advances in Neural Information Processing Systems* (eds Wallach, H. et al) 15820–15831 (NeurIPS, 2019).
16. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
17. Baek, M. et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **373**, 871–876 (2021).
18. Carreira, J. et al. Human pose estimation with iterative error feedback. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition* (eds Bajcsy, R. et al.) 4733–4742 (IEEE, 2016).
19. Tu, Z. & Bai, X. Auto-context and its application to high-level vision tasks and 3D brain image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**, 1744–1757 (2010).
20. Lin, Z. et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* **379**, 1123–1130 (2023).
21. Robin, X. et al. Continuous Automated Model EvaluatiOn (CAMEO)—perspectives on the future of fully automated evaluation of structure prediction methods. *Proteins* **89**, 1977–1986 (2021).
22. *CASP15. Critical Assessment of Techniques for Protien Structure Prediction, 15th Round. Abstract Book* (Protein Structure Prediction Center, 2022); https://predictioncenter.org/casp15/doc/CASP15_Abstracts.pdf
23. Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference* (Morgan Kaufmann, 1988).
24. Wainwright, M. J. & Jordan, M. I. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.* **1**, 1–305 (2008).
25. Zhang, H. et al. Predicting protein inter-residue contacts using composite likelihood maximization and deep learning. *BMC Bioinform.* **20**, 537 (2019).
26. Ekeberg, M., Lövkvist, C., Lan, Y., Weigt, M. & Aurell, E. Improved contact prediction in proteins: using pseudolikelihoods to infer Potts models. *Phys. Rev. E* **87**, 012707 (2013).
27. Morcos, F. et al. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc. Natl Acad. Sci. USA* **108**, 1293–1301 (2011).
28. Alford, R. F. et al. The Rosetta all-atom energy function for macromolecular modeling and design. *J. Chem. Theory Comput.* **13**, 3031–3048 (2017).
29. Henikoff, S. & Henikoff, J. G. Amino acid substitution matrices from protein blocks. *Proc. Natl Acad. Sci. USA* **89**, 10915–10919 (1992).
30. Wang, W., Peng, Z. & Yang, J. Single-sequence protein structure prediction using supervised transformer protein language models. *Nat. Comput. Sci.* **2**, 804–814 (2022).
31. Chowdhury, R. et al. Single-sequence protein structure prediction using a language model and deep learning. *Nat. Biotechnol.* **40**, 1617–1623 (2022).
32. Sakuma, K., Koike, R. & Ota, M. Dual-wield NTPases: a novel protein family mined from AlphaFold DB. *Protein Science.* **33**, e4934 (2024).
33. Varadi, M. et al. AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Res.* **50**, 439–444 (2022).
34. Yang, K. K., Wu, Z. & Arnold, F. H. Machine-learning-guided directed evolution for protein engineering. *Nat. Methods* **16**, 687–694 (2019).
35. Shin, J.-E. et al. Protein design and variant prediction using autoregressive generative models. *Nat. Commun.* **12**, 2403 (2021).
36. Lek, M. et al. Analysis of protein-coding genetic variation in 60,706 humans. *Nature* **536**, 285–291 (2016).
37. Frazer, J. et al. Disease variant prediction with deep generative models of evolutionary data. *Nature* **599**, 91–95 (2021).
38. Meier, J. et al. Language models enable zero-shot prediction of the effects of mutations on protein function. In *Proc. of Advances in Neural Information Processing Systems* (eds Ranzato, M. et al.) 29287–29303 (NeurIPS, 2021).
39. Notin, P. et al. Tranception: protein fitness prediction with autoregressive transformers and inference-time retrieval. In *Proc. of the 39th International Conference on Machine Learning* (eds Chaudhuri, K. et al.) 16990–17017 (PMLR, 2022).
40. Rao, R. M. et al. MSA transformer. In *Proc. of the 38th International Conference on Machine Learning* (eds Meila, M and Zhang, T.) 8844–8856 (PMLR, 2021).
41. Findlay, G. M. et al. Accurate classification of BRCA1 variants with saturation genome editing. *Nature* **562**, 217–222 (2018).
42. Kotler, E. et al. A systematic p53 mutation library links differential functional impact to cancer mutation pattern and evolutionary conservation. *Mol. Cell* **71**, 178–1908 (2018).
43. Mighell, T. L., Evans-Dutson, S. & O'Roak, B. J. A saturation mutagenesis approach to understanding PTEN lipid phosphatase activity and genotype-phenotype relationships. *Am. J. Hum. Genet.* **102**, 943–955 (2018).
44. Jia, X. et al. Massively parallel functional testing of MSH2 missense variants conferring Lynch syndrome risk. *Am. J. Hum. Genet.* **108**, 163–175 (2021).
45. Pan, X. et al. Structure of the human voltage-gated sodium channel Nav1.4 in complex with beta1. *Science* **362**, 2486 (2018).
46. Hennig, M., Darimont, B., Sterner, R., Kirschner, K. & Jansonius, J. N. 2.0 Å structure of indole-3-glycerol phosphate synthase from the hyperthermophile Sulfolobus solfataricus: possible determinants of protein stability. *Structure* **3**, 1295–1306 (1995).
47. Banerjee, S. et al. Protonation state of an important histidine from high resolution structures of lytic polysaccharide monooxygenases. *Biomolecules* https://doi.org/10.3390/biom12020194 (2022).
48. Watson, J. L. et al. De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).
49. Leman, J. K. et al. Macromolecular modeling and design in rosetta: recent methods and frameworks. *Nat. Methods* **17**, 665–680 (2020).
50. Madani, A. et al. Large language models generate functional protein sequences across diverse families. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-022-01618-2 (2023).
51. Hie, B. L. et al. Efficient evolution of human antibodies from general protein language models. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-023-01763-2 (2023).

52. Suzek, B. E. et al. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* **31**, 926–932 (2015).
53. Mitchell, A. L. et al. MGnify: the microbiome analysis resource in 2020. *Nucleic Acids Res.* **48**, 570–578 (2020).
54. Mirdita, M. et al. Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res.* **45**, 170–176 (2017).
55. Remmert, M., Biegert, A., Hauser, A. & Söding, J. HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods* **9**, 173–175 (2012).
56. Johnson, L. S., Eddy, S. R. & Portugaly, E. Hidden Markov model speed heuristic and iterative HMM search procedure. *BMC Bioinform.* **11**, 431 (2010).
57. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In *Proc. of the International Conference on Learning Representations* (eds Bengio, Y. et al.) 210–219, (ICLR 2015).
58. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. In *Proc. of Advances in Neural Information Processing Systems* (eds Wallach, H. et al.) 8024–8035 (NeurIPS, 2019).
59. Ren, M., Yu, C., Bu, D. & Zhang, H. Accurate and robust protein sequence design with Carbondesign. *Code Ocean* https://doi.org/10.24433/CO.5915382.v2 (2024).

Acknowledgements

We acknowledge the financial support from the National Natural Science Foundation of China (grant no. 32370657) and the Project of Youth Innovation Promotion Association CAS to H.Z. We also acknowledge the financial support from the Development Program of China (grant no. 2020YFA0907000) and the National Natural Science Foundation of China (grant nos. 32271297 and 62072435). We thank Beijing Paratera Co., Ltd and the ICT Computing-X Center, Chinese Academy of Sciences, for providing computational resources.

Author contributions

H.Z. conceived the ideas and implemented the CarbonDesign model and algorithms. H.Z. and M.R. designed the experiments, and M.R. conducted the main experiments and analysis. M.R. wrote the manuscript. H.Z., D.B. and C.Y. revised the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-024-00838-2.

Correspondence and requests for materials should be addressed to Dongbo Bu or Haicang Zhang.

Peer review information *Nature Machine Intelligence* thanks Haiyan Liu and Dong Xu for their contribution to the peer review of this work.

nature portfolio

Corresponding author(s): Haicang Zhang

Last updated by author(s): Apr 1, 2024

Reporting Summary

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

Statistics

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

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

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

Software and code

Policy information about availability of computer code

Data collection	We use RFdiffusion and FrameDiff to generate de novo protein structures. We use MMseqs2 to cluster the training data.
Data analysis	PyMOL (version 2.5.0) was used to display the 3D structures, TM-score (version 1.0) was used to evaluate the predicted structures.

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

Data

Policy information about availability of data

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

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

Our training set can be download from PDB website (http://www.rcsb.org/). Our testing sets are from CASP15 (https://predictioncenter.org/casp15/) and CAMEO (https://www.cameo3d.org).

Human research participants

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

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

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

Field-specific reporting

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

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

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

Life sciences study design

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

Sample size	No statistical methods were used to predetermine sample sizes. We did not choose any specific sample sizes.
Data exclusions	None (no data were excluded from the analyses.)
Replication	We have run our code three times, and each time it repeated successfully.
Randomization	N/A (all analyses are automated, so all data is generated through calculations with default settings.)
Blinding	N/A (all programs and analyses are preconfigured, so there was no user intervention that could have introduced bias.)

Reporting for specific materials, systems and methods

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

Materials & experimental systems

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

Methods

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