Open access • Posted Content • DOI:10.1101/2021.09.03.458869

Protein embeddings and deep learning predict binding residues for various ligand classes — Source link

Maria Littmann, Michael Heinzinger, Christian Dallago, Konstantin Weissenow ...+2 more authors

Institutions: Technische Universität München, Columbia University

Published on: 05 Sep 2021 - bioRxiv (Cold Spring Harbor Laboratory)

Topics: Binding site

Protein embeddings and deep learning predict binding residues for various ligand classes

Maria Littmann[1,*], Michael Heinzinger[1,2], Christian Dallago[1,2], Konstantin Weissenow[1,2], & Burkhard Rost[1,3,4]

1 TUM (Technical University of Munich) Department of Informatics, Bioinformatics & Computational Biology - i12, Boltzmannstr. 3, 85748 Garching/Munich, Germany
2 TUM Graduate School, Center of Doctoral Studies in Informatics and its Applications (CeDoSIA), Boltzmannstr. 11, 85748 Garching, Germany
3 Institute for Advanced Study (TUM-IAS), Lichtenbergstr. 2a, 85748 Garching/Munich, Germany & TUM School of Life Sciences Weihenstephan (TUM-WZW), Alte Akademie 8, Freising, Germany
4 Department of Biochemistry and Molecular Biophysics, Columbia University, 701 West, 168th Street, New York, NY 10032, USA

* Corresponding author: littmann@rostlab.org, http://www.rostlab.org/
Tel: +49-289-17-814 (email rost: assistant@rostlab.org)

Abstract

One important aspect of protein function is the binding of proteins to ligands, including small molecules, metal ions, and macromolecules such as DNA or RNA. Despite decades of experimental progress many binding sites remain obscure. Here, we proposed *bindEmbed21*, a method predicting whether a protein residue binds to metal ions, nucleic acids, or small molecules. The Artificial Intelligence (AI)-based method exclusively uses embeddings from the Transformer-based protein Language Model ProtT5 as input. Using only single sequences without creating multiple sequence alignments (MSAs), *bindEmbed21DL* outperformed existing MSA-based methods. Combination with homology-based inference increased performance to F1=29±6%, F1=24±7%, and F1=41±% for metal ions, nucleic acids, and small molecules, respectively; it reached F1=45±2% when merging all three ligand classes into one. Focusing on very reliably predicted residues could complement experimental evidence: the 25% most strongly predicted binding residues, at least 73% were correctly predicted even when counting missing annotations as incorrect. The new method *bindEmbed21* is fast, simple, and broadly applicable - neither using structure nor MSAs. Thereby, it found binding residues in over 42% of all human proteins not otherwise implied in binding.

Key words: function prediction, binding residue prediction, machine learning, deep learning, language model, transfer learning, convolutional neural networks

Abbreviations used: **AI**, artificial intelligence (expanding ML through deep learning, i.e., using more free parameters); **CI**, confidence interval; **CNN**, Convolutional Neural Network; **HBI,** homology-based inference; **(p)LM**, (protein) language model; **MCC**, Matthews Correlation Coefficient; **ML,** machine learning; **MSA**, multiple sequence alignment; **PDB**, Protein Data Bank; **PIDE,** pairwise sequence identity; **SOTA**, state-of-the-art; **SVM,** support vector machine.

Introduction

Experimental data for protein binding remains limited. Knowing protein function is crucial to understand the molecular mechanisms of life[1]. For most proteins, function of proteins depends on binding to other molecules called *ligands*[2]; these include metal ions, inorganic molecules, small organic molecules, or large biomolecules such as DNA, RNA, and other proteins. Although the variation in characteristics of protein binding sites resembles the diversity of the biophysical properties of the ligands, binding sites are highly specific and often determined by a few key residues [2]. Binding residues are experimentally determined most reliably through high-resolution structures of the protein in complex with the respective ligand and identifying residues in close proximity to this ligand as binding residues (e.g., ≤ 5Å)[3].

Prediction methods usually rely on evolutionary information. Despite immense progress in quantitative high-throughput proteomics, experimentally verified binding residues remain unknown for most proteins[4]. In fact, reliable binding data remains so sparse to render even Machine Learning (ML) approaches optimizing fewer parameters than tools from Artificial Intelligence (AI) extremely challenging[5]. Thus, reliable prediction methods become an important bridge, e.g., to study the effect of sequence variation in human populations[6,7]. Homology-based inference allows the transfer of binding residues from sequence-similar proteins with known annotations to experimentally uncharacterized proteins[5,8]. If unavailable, *de novo* prediction methods based on ML try to fill the gap. Structure-based methods usually outperform sequence-based methods[9,10], but they also rely on the availability of experimental high-resolution structures and are computationally intensive[10-14]. For instance, COACH[10] is an ensemble classifier combining five individual approaches and has been considered the state-of-the-art (SOTA) method for binding residue prediction for many years[15,16]. However, the prediction for a single protein takes about 10 hours on their webserver and a local installation of the method requires 60GB free disk space to download the necessary databases of structural templates. On the other hand, sequence-based methods usually depend on sufficiently diverse and reliable experimental data and expert-crafted input features including evolutionary information to represent protein sequences[5,15,17,18]. Our previously published method bindPredictML17[5] allowed predictions of binding residues for enzymes and DNA-binding proteins while relying mainly on information from sequence variation[19,20] and co-evolving residues[21], both requiring the time-consuming computation of multiple sequence alignments (MSAs). Similarly, ProNA2020[17] uses evolutionary profiles and various features from PredictProtein[22] to predict protein-protein, protein-DNA, and protein-RNA binding again requiring the computation of MSAs. In addition to the complexity of their input features, many methods specialize on specific ligands or sets thereof, since the biophysical features optimal for prediction differ between ligands[5,14,16-18,23-27]. For instance, PredZinc[18] only predicts zinc ions and IonCom[16] provides predictions for 13 metals and four radical ion ligands. Most existing somehow reliable sequence-based methods cannot be applied to large sets of protein sequences due to time limitations for feature computation or due to restriction to a very limited set of ligands.

Here, we propose a new method dubbed *bindEmbed21* predicting binding residues for three main classes of ligands. To overcome the limitation of expert-crafted input features and the necessity to create MSAs, we represent protein sequences as embeddings, i.e., fixed-length vectors derived from pre-trained protein Language Models (pLMs), in particular tapping into the power of the pLM ProtT5[28]. Based on those embeddings, bindEmbed21 predicts whether or not a residue binds to metal ions, nucleic acids (DNA and RNA), and/or regular small molecules. Combining the *de novo* prediction method with homology-based inference further improved performance. Because embeddings can be easily extracted for any protein sequence, bindEmbed21 allows fast and easy predictions for all available protein sequences.

Results & Discussion

Embedding-based predictions from bindEmbed21DL successful. Inputting raw ProtT5[28] embeddings into a shallow two-layer CNN, our new method, *bindEmbed21DL*, predicted for each residue in a protein, whether or not it binds to a metal ion, a nucleic acid (DNA or RNA), or a small molecule. The prediction differed substantially between the three classes (Fig. 1, Table S1 in Supporting Online Material (SOM)): binding residues were predicted best for small molecule and worst for nucleic acids (Table 1, DevSet1014; Fig. 1A-C). Performance appeared highest when dropping the distinction between ligand classes, i.e., simplifying the task to the prediction of binding vs. non-binding (Table 1; Fig. 1D).

Table 1: F1 score (harmonic mean of precision and recall). *

Method	Dataset	F1-metal	F1-XNA	F1-small	F1-all
bindEmbed21DL	*DevSet1014*	24±2%	18±3%	26±2%	39±2%
bindEmbed21DL	*TestSet300*	22±4%	24±6%	33±3%	43±2%
bindEmbed21DL	*TestSetNew46*	26±14%	19±11%	29±9%	37±6%
bindEmbed21DL	*TestSet225*	n/a	n/a	n/a	47±2%
bindPredictML17	*TestSet225*	n/a	n/a	n/a	34±2%
bindEmbed21DL	$TestSet300_{XNA66}$	n/a	31±5%	n/a	n/a
ProNA2020	$TestSet300_{XNA66}$	n/a	33±7%	n/a	n/a
bindEmbed21DL	$TestSet300_{Zinc51}$	58±8%	n/a	n/a	n/a
PredZinc	$TestSet300_{Zinc51}$	58±10%	n/a	n/a	n/a

* **Measure**: F1 (Eqn. 3); **±** : 95% confidence intervals (1.96 standard errors); **Methods**: *bindEMbed21DL*: method introduced here, *bindPredictML17*[5]: MSA-based method predicting binding, *ProNA2020*[17]: method specialized on predicting binding to DNA, RNA, and other proteins; *PredZinc*[18]: method specialized on predicting zinc-binding; **Data**: *DevSet1014*: development set (validation/cross-training) set with 1,014 proteins, *TestSet300*: Test set used for development with 300 proteins, *TestSet225*: subset of test set shared with bindPredictML17, *TestSetNew46*: 46 sequence-unique proteins added since development of this work began – all sequence-unique with respect to each other and all other proteins used, $TestSet300_{XNA66}$: subset with DNA or RNA (dubbed XNA) binding proteins from our test set. $TestSet300_{Zinc51}$: subset with zinc-binding proteins from our test set.

Performance for the individual ligand classes appeared limited by over-prediction (binding predictions not experimentally confirmed, yet) and cross-predictions (predicted to bind ligand C1, annotated for C2). Thus, predicting individual ligand classes was more challenging than the binary distinction of *residue binding/non-binding*. Nevertheless, bindEmbed21DL performed similar to a method trained solely on this binary task (Table S4; SOM section 1.1 for more details).

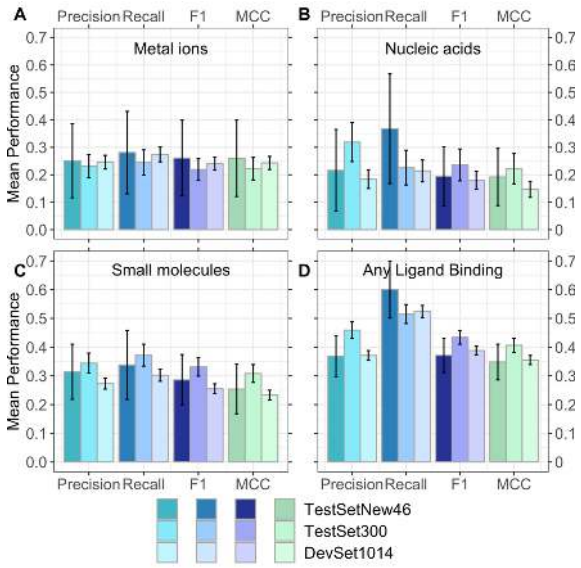

Fig. 1: Performance of new method *bindEmbed21DL*. Performance captured by four per-residue measures: precision (Eqn. 2), recall (Eqn. 1), F1 score (Eqn. 3), and MCC (Eqn. 4). Data sets: *DevSet1014* (validation/cross-training set of cross-validation development, most light colors), *TestSet300* (fixed test set used during development, darker colors), and *TestSetNew46* (additional test set compiled after development, most dark colors). Predictions of residues binding to **A.** metal ions, **B.** nucleic acids (DNA or RNA), **C.** small molecules, and **D.** any ligand class grouping all three classes into one (considering each residue predicted/observed to bind to one of the three ligand classes as binding, all others as non-binding). On the cross-training set *DevSet1014*, *bindEmbed21DL* predicted any binding residue with F1=39±2%. Surprisingly, the number was slightly higher for the test set *TestSet300* (F1=43±2%) while being similar on the additional test set *TestSetNew46* (F1=37±6%). Error bars indicate 95% CIs.

In a typical cross-validation split (training, validation/cross-training, test), performance values are higher for the validation than for the test set, because hyper-parameters are optimized on the former. We observed the inverse (Table 1, Fig. 1) although most differences were within the confidence intervals (Fig. 1, Table S1). We had frozen and set aside our test set, to simplify the comparison to an older method (bindPredictML17[5]) which was trained solely on enzymes and DNA-binding proteins. Thus, the higher numbers for the test set could indicate that binding residues are better defined and therefore easier to predict for enzymes.

To investigate, we created an independent test set from recent annotations (TestSetNew46, Methods: 46 unique from a total of 1,592 new proteins). For these, *bindEmbed21DL* reached values that, within the 95% confidence interval, agreed with both the original test and validation sets because two years did not accumulate enough experimental data to distinguish similar values with statistical significance. When merging all ligand classes, the new test set was large enough to establish with statistical significance (95% CI) that our performance estimates reflected what is to be expected from the next 1,592 proteins submitted for prediction (Methods).

To provide binding predictions for as many proteins as possible, we considered a protein to bind to a specific ligand class if at least one residue was predicted to bind to this class. However, binding usually involves more than one residue. Therefore, predictions could be further filtered by only considering residues as binding if at least *x* residues were predicted to bind to this ligand class. Applying this filter led to an increase in CovNoBind(l) (Eqn. 9) for larger *x* while decreasing CovOneBind (Eqn. 8; Fig. S1). While precision and recall were set to 0 for proteins annotated but not predicted to bind to a certain ligand class, those performance values still increased up to a certain threshold (Fig. S1; optimal threshold of 3, 10, and 8 residues for metal ions, nucleic acids,

and small molecules, respectively) because more proteins falsely predicted to bind to this ligand class were removed than proteins actually binding to a certain ligand. Therefore, the number of residues predicted to bind to a certain ligand class could help finding incorrect predictions (too few residues predicted: prediction less likely correct).

Embeddings clearly outperformed MSA-based predictions. Recently, we had developed *bindPredictML17*[5] predicting binding residues based on MSAs, namely information about co-evolving residues and sequence variant effect predictions. A subset of the test set (225 of the 300 proteins in TestSet300) enabled an unbiased comparison of both methods: *bindEmbed21DL*, statistically significantly (beyond 95% CI) outperformed the old MSA-based method *bindPredictML17* (Fig. 2A), e.g., raising the harmonic mean over precision and recall by 13 percentage points (Table 1, bindEmbed21DL vs. bindPredictML17 last column for TestSet225). However, *bindEmbed21DL* predicted binding for only 222 of the 225 test proteins (CovOneBind=99%, Eqn. 8), while its predecessor predicted for all 225.

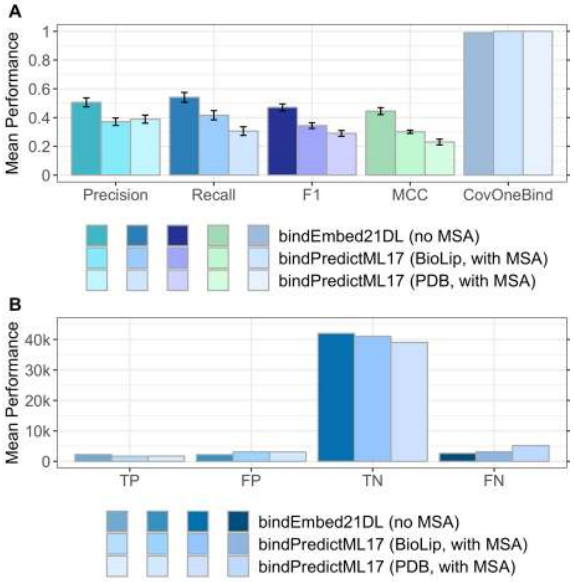

Fig. 2: Embeddings outperformed MSA-based predictions. This graph compares the performance between *bindPredictML17*[5] using multiple sequence alignments (MSAs) and the new method introduced here, *bindEmbed21DL*, using only embeddings from ProtT5 [28]. We also compare using binding annotations from BioLiP[9] or the PDB[29]. **Panel A:** bindEmbed21DL (embeddings-only) clearly outperformed bindPredictML17 (MSA+BioLiP) by 13 percentage points (F1=47±2% vs. F1=34±2%). We used annotations from BioLiP[9] to assess the performance for both methods. Although, bindPredictML17 had been trained on annotations from PDB[29] for enzymes and PDIdb[30] for DNA-binding proteins, it reached higher performance (lighter shaded colors vs. lightest shaded colors) for BioLiP annotations**. Panel B:** Investigating the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) revealed that bindPredictML17 predicted many more FN when measured by PDB annotations than by BioLiP annotations. Hence, bindPredictML17 captured the incorrect binding annotations from the PDB correctly predicting those as non-binding which worsened its performance when assessing on those annotations but actually better captured the true binding residues. Error bars indicate 95% CIs. More details on the comparison of bindPredictML17 using BioLiP or PDB annotations can be found in SOM, Section 1.2.

bindEmbed21DL competitive to specialist methods. *bindEmbed21DL* simultaneously predicted whether a residue is binding to metal ions, nucleic acids, or small molecules, while many state-of-the-art (SOTA) methods specialize on one ligand class. For instance, *ProNA2020*[17]

focuses on predicting protein-, DNA-, or RNA-binding, both on the per-protein (does protein bind DNA or not?) and the per-residue (which residue binds DNA?) level. ProNA2020 depends completely on MSAs. While ProNA2020 shines through unifying a hierarchy of prediction tasks, it also appeared to outperform all available other methods in predicting whether or not a residue binds DNA or RNA (dubbed XNA)[17]. We compared the specialist *ProNA2020* with the generalist *bindEmbed21DL* using 66 nucleic acid binding proteins in *TestSet300* (dubbed TestSet300$_{XNA66}$ in Table 1). For those 66 proteins, the MSA-based specialist *ProNA2020* performed slightly worse in XNA-binding prediction than the embedding-based MSA-free bindEmbed21DL (F1=31±5% vs F1=33±7%, Fig. 3A). However, when analyzing how many proteins had at least one residue predicted as XNA-binding (DNA or RNA), namely using the measure CovOneBind (Eqn. 8), the situation reversed: CovOneBind(ProNA2020)=85% vs. CovOneBind(bindEmbed21DL-XNA)=77% (Fig. 3A). When considering all residues predicted by bindEmbed21DL as binding (bind=nucleic acids + metal ions + small molecules), F1 rose almost ten percentage points to 43±5% and CovOneBind to 97% (Fig. 3A, bindEmbed21DL). This clearly indicated that performance of bindEmbed21DL for the individual ligand classes was limited due to cross-predictions (Table S3), i.e., residues predicted to bind to one ligand class and observed to bind to another ligand class.

PredZinc[18] is another specialist trained to predict residues binding to zinc ions. While it is not the most recent method available, it provides a webserver which is still maintained and generates results quickly. With newer metal-binding prediction methods, we experienced problems either those were unavailable or took too long to predict for multiple proteins. Therefore, we chose PredZinc as a specialist predictor for metal binding. 51 proteins in *TestSet300* were annotated to bind to zinc ions (dubbed TestSet300$_{Zinc51}$ in Table 1), and we used those to compare *PredZinc* to the generalist *bindEmbed21DL*. While not being trained to predict zinc-binding, bindEmbed21DL achieved the same performance in terms of F1 score as PredZinc (F1=58±8% vs. F1=58±10%, Fig. 3B) with a lower precision, but higher recall than PredZinc (Fig. 3B). bindEmbed21DL also achieved a higher CovOneBind (Eqn. 8) than PredZinc making a prediction for 94% of the proteins compared to 80% for PredZinc. Different to the observation for nucleic acid binding, performance dropped when considering all residues predicted by bindEmbed21DL as binding (F1=34±5%, Fig. 3B). While there were some cross-predictions as seen by the gain in recall (Fig. 3B), only a few residues are usually involved in metal binding. Therefore, combining all binding prediction introduced many false positives (predicted to bind, not observed), while only removing few false negatives (observed to bind, not predicted).

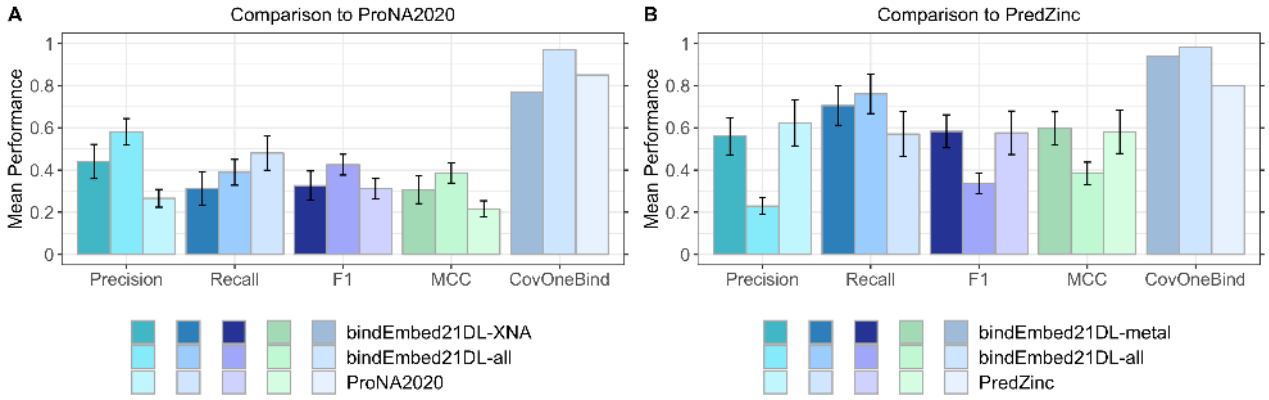

Fig. 3: bindEmbed21DL competitive with specialists. Panel A: XNA binding. Data: 66 DNA- or RNA-binding (dubbed XNA) proteins from the test set *TestSet300*. ProNA2020[17] (lightest shaded bars) uses MSAs to predict DNA-, RNA-, and protein-binding, while the method introduced here uses embeddings only (no MSA); bindEmbed21DL-XNA (darkest shaded bars) marked predictions of either DNA or RNA (XNA); bindEmbed21DL-all (lighter shaded bars) marked using

all binding predictions and assessing only XNA-binding. While the difference in F1 scores between the three methods was within the error bars (95% CIs), bindEmbed21DL (-XNA and -all) achieved a statistically significant higher performance than ProNA2020 while ProNA2020 achieved a higher recall. Also, the fraction of proteins with at least one XNA prediction (CovOneBind, Eqn. 8) was higher for ProNA2020 than for bindEmbed21DL-XNA. However, when considering any residue predicted as binding (*bindEmbed21DL-all*: nucleic acid, or metal ion, or small molecule), our new method apparently reached the highest values due to confusions between XNA and other ligands (Table S3). **Panel B: Zinc-binding.** Data: 51 zinc-binding proteins from *TestSet300*. PredZinc[18] (lightest shaded bars) predicts zinc-binding; bindEmbed21DL-metal (darkest shaded bars) marked predictions for metal ions, bindEmbed21DL-all (lighter shaded bars) marked using all binding predictions and assessing only metal binding. bindEmbed21DL-metal achieved a similar performance as PredZinc, while providing predictions for more proteins (CovOneBind(bindEmbed21DL-metal)=94% vs. CovOneBind(PredZinc)=80%).

More reliable predictions better. For the prediction of binding vs non-binding residues, bindEmbed21DL achieved precision=37±2% and recall=52±2% (Fig. 1D, lighter colored bars) while making predictions for 1,000 of 1,014 proteins in the cross-training set (*DevSet1014*) (CovOneBind=99%). These values resulted from the default threshold optimized by the ML method considering all predictions with probability≥0.5 as binding, all others as non-binding. If only the 1,000 proteins with a prediction were considered, precision and recall rose by one percentage point to 38% and 53%, respectively (Fig. 4). We analyzed the trade-off between precision, recall, and CovOneBind in dependence of the output probability: Precision decreased for lower cutoffs but recall and CovOneBind increased allowing more binding predictions for more proteins (Fig. 4, Table S5). For instance, at a cutoff of 0.28, at least one binding prediction was generated for every protein (CovOneBind=100%) corresponding to a drop in precision by nine percentage points (Fig. 4, Table S5). On the other hand, precision could be increased by applying higher cutoffs to define a residue as binding. For instance, for a cutoff of 0.95, precision almost doubled (Fig. 4, Table S5). While recall and CovOneBind in general decreased for higher cutoffs, bindEmbed21DL still made predictions for more than half of the proteins and for one fourth of all binding residues at this very high cutoff of 0.95 (Fig. 4, Table S5).

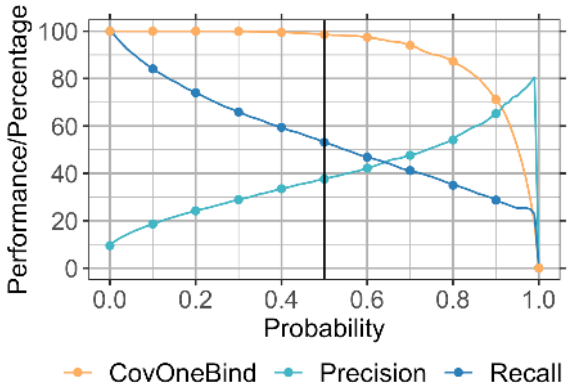

Fig. 4: Residues predicted stronger more often correctly predicted. Data set: DevSet1014. Precision and recall are only shown for the proteins for which at least one residue was predicted as binding where the number of such proteins is indicated by CovOneBind. The x-axis gives the output probability of bindEmbed21DL for a prediction corresponding to the prediction strength. The y-axis gives the average performance or percentage of proteins with a prediction at the respective probability cutoff. All curves give the cumulative values, e.g., the precision of all residues predicted with probability ≥ 0.95 was 73% corresponding to a recall of 25%; and at that value, at least one binding residue was predicted in 51% of the proteins. While higher probabilities correspond to more reliable binding predictions, lower probabilities correspond to highly reliable non-binding predictions (Table S5).

Considering different ligand classes, we observed similar results for precision and CovOneBind, i.e., precision increased while CovOneBind dropped for higher cutoffs and vice versa for lower cutoffs (Fig. S3). However, the trend was different for recall: While recall decreased as expected for higher cutoffs for small molecules (Fig. S3C), it first decreased and then increased for metal ions (Fig. S3A), and first increased and then decreased for nucleic acids (Fig. S3B). For proteins not binding to a certain ligand class *x* for which any residue was predicted to bind to *x,* precision and recall were set to 0. Increasing the cutoff to define a residue as binding decreased the number of residues incorrectly predicted to bind to *x.* Therefore, for more proteins not bound to *x,* there were also no residues predicted to bind to *x,* and those proteins were then ignored for the performance assessment (i.e., recall and precision are not set to 0). Therefore, recall could increase for higher cutoffs because CovNoBind increased (Fig. 3).

Since the probability cutoff correlated with the reliability of the predictions, we transformed the probability into a single-digit integer reliability index (RI) (Eqn. 10) ranging from 0 (unreliable; probability=0.5) to 9 (very reliable). This RI allowed the user to easily focus on the most reliable predictions either for binding or non-binding residues.

Reliable predictions could help refining experimental annotations. Using a cutoff of 0.95 to classify a residue as "binding", bindEmbed21DL achieved a precision of 73% with at least one residue predicted as binding for 519 proteins (CovOneBind=51%; Fig. 4, Table S5). Despite this high precision, for 84 of the 519 proteins (16%), none of the reliably predicted residues predicted that reliably had been experimentally annotated as binding. We analyzed two of those 84 in more detail.

For instance, the DNA-binding protein HMf-2 (UniProt ID: P19267) is annotated to bind to a metal ion at positions 34 and 38 based on the PDB structure 1A7W[29,31] with a resolution of 1.55Å. However, none of those positions was predicted as binding, either at a cutoff of 0.5 or 0.95. In addition, the name and the available functional annotations suggested this protein to bind DNA. If correct, the observed metal-binding might point to allosteric binding. Four residues were also predicted reliably (probability≥0.95) to bind nucleic acids (Fig. 5A, dark red residues). For another PDB structure of this protein (PDB identifier 5T5K[29,32] at 4.0Å resolution), BioLiP annotates DNA-binding, including for all four reliably predicted residues. Due to our threshold in resolution, this protein had not been included in our data sets. Overall, BioLiP annotates 13 residues in 5T5K as binding, 10 of those were correctly predicted as nucleic acid-binding (Fig. 5A, lighter red residues) corresponding to a recall of 77%. With respect to the three remaining: although our sequence-based method clearly did not aspire to reach anywhere near the power of X-ray crystallography, at least some of the parts of the proteins seemingly bridged over by the major grove (Fig. 5A: dark blue) might, indeed not bind DNA.

We observed similar results for the ribonuclease P protein component (UniProt ID: Q9X1H4): Using the PDB structure 6MAX[29,33] with a resolution of 1.42Å, this protein is annotated to have seven residues binding to a small molecule while bindEmbed21DL did not predict any of those with a high probability above 0.95. In fact, the available functional annotations clearly suggest this protein to be binding to nucleic acids and the small molecule bound according to the PDB structure 6MAX seems to mainly serve as inhibitor for RNA-binding[33]. Four residues were also predicted to bind to nucleic acids above a probability of 0.95 (Fig. 5B, dark red residues). The low-resolution structures 3Q1Q (3.8Å)[29,34] and 3Q1R (4.21Å)[29,34] also provided annotations for binding to nucleic acids for this protein. The four most reliable predictions were also annotated as binding based on those two structures, and of the 21 residues annotated as binding, 16 were also predicted to be binding with a probability ≥ 0.5 (Fig. 5B, lighter red residues; recall=76%).

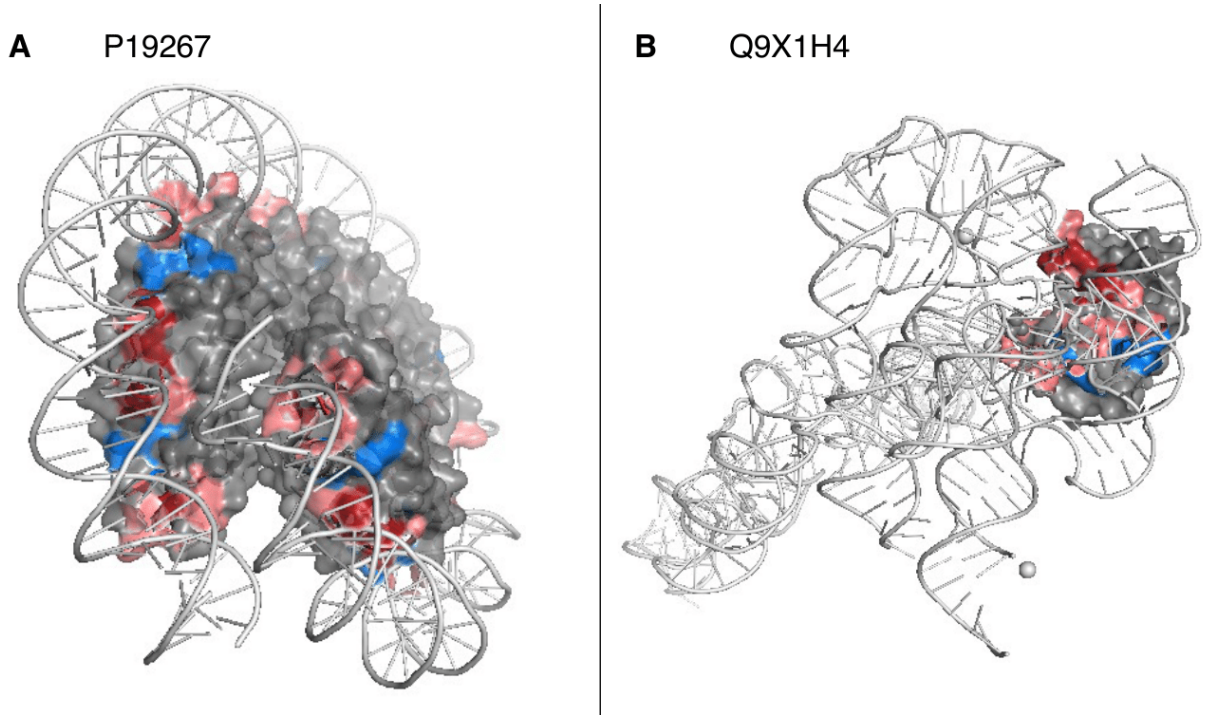

Fig. 5: Annotations from low-resolution structures supported through reliable predictions. A: Our development set (DevSet1014) contained the PDB structure 1A7W[29,31] for the DNA-binding protein HMf-2 (UniProt ID: P19267). No DNA/nucleic acid binding was annotated in that structure, but our new method, bindEmbed21DL, reliably predicted (probability ≥0.95) four residues to bind nucleic acids. Shown is the PDB structure 5T5K[29,32] for the same protein that has a resolution of 4.0Å and annotations of DNA-binding, including the four most reliable predictions (dark red). Overall, 10 of 13 (77%) residues annotated as DNA-binding in 5T5K were also predicted by bindEmbed21DL (shown in lighter red; blue residues indicate experimental annotations which were not predicted). **B:** For the ribonuclease P protein component (UniProt ID: Q9X1H4), four residues were predicted with a probability ≥0.95 (indicated in dark red), none of these matched the annotations in the PDB structure 6MAX[29,33]. However, those four residues were considered as binding according to the two low-resolution structures 3Q1Q (3.8Å)[29,34] (visualized) and 3Q1R (4.21Å)[29,34]. In total, those structures marked 21 binding residues; 15 of those 21 (71%) were correctly predicted (light red; blue residues observed to bind but not predicted). These two examples highlighted how combining low-resolution experimental data and very reliable predictions from bindEmbed21DL could refine those annotations and/or help designing new investigations.

These two of 84 examples pitched bindEmbed21DL as a candidate tool to help in experimentally characterizing new binding residues completely different from the annotations it was trained on. On the one hand, this facilitates the identification of previously unknown binding sites, and on the other hand, it might also help to verify and refine known, but potentially unreliable binding annotations, especially if multiple structures annotating different binding sites are available. In the two examples shown here, both proteins had already been annotated as binding to nucleic acids in less well-resolved structures, while the binding annotations from high-resolution structures rather pointed to binding of co-factors or inhibitors. Combining the low-resolution annotations with the very reliable predictions from bindEmbed21DL clearly suggested four positions (Fig. 5, dark red residues) to be involved in nucleic acid binding. Those strongly predicted binding residues could be further complemented by surrounding residues with weaker predictions (Fig. 5, lighter red residues). The 3-5 residues with experimental annotations that were not predicted (Fig. 5, blue residues) might even point to potential annotation mistakes originating from the limited experimental resolution. Overall, the examples suggested that the seemingly low performance of bindEmbed21DL clearly partially rooted in the incomplete experimental annotations used to assess performance (not yet observed to bind treated as non-binding, which

proved incorrect for most residues of the two proteins assessed). In fact, of the 84 proteins with incorrect, highly reliable predictions, 32 were predicted to bind nucleic acids. For 6 of those 32 proteins (19%), low resolution structures with binding annotations at least partially matching the predictions were available. On the other hand, only one of the 75 proteins with incorrect metal predictions (1%) and one of the 80 proteins with incorrect predictions to small molecules could be explained by annotations from low resolution structures. This clearly suggested that the highly reliable predictions from bindEmbed21DL did not only correspond to binding annotations from low-resolution structures but could in fact point towards still unknown binding sites.

Final method bindEmbed21 combines HBI and ML to top performance. Homology-based inference (HBI) assumes that two sequence-similar proteins are evolutionary related, and therefore, also share a common function. Using HBI to predict binding residues for three different ligand classes for our training set yielded very good results for low E-value thresholds, but at those thresholds, hits were only found for very few proteins (Fig. S4). For instance, for E-values $\leq 10^{-50}$, HBI achieved F1=56±4% (Fig. S4, leftmost dark red bar), but at that restrictive E-value, only 198 of the 1,014 proteins found a hit, i.e., another protein with experimental annotations. When only using HBI to make a prediction for all proteins, a random decision would have to be made if no homolog with experimentally known binding annotations were available at the given threshold. Penciling in such a random decision dropped performance immensely (F1=21±2% for E-value$\leq 10^{-50}$; Fig. S4, leftmost light red bar). To harness the strong performance of HBI while allowing better than random predictions for proteins without close homologs, we combined bindEmbed21DL with HBI applying a simple protocol: Predict binding residues through HBI if a sequence-similar protein with annotations is available; otherwise use ML. This combination achieved optimal performance at an E-value threshold of 10^{-3} leading to F1=45±2% (Fig. S4A, blue bar at E-value = 10^{-3}) and precision=46±2% (Fig. S4B, blue bar at E-value = 10^{-3}). While F1 and precision were also higher than the performance for only using the ML method bindEmbed21DL for higher E-value cutoffs, recall dropped below the level of bindEmbed21DL (Fig. S4C). Therefore, we considered 10^{-3} the optimal threshold.

Combining ML and HBI improved performance on the test set TestSet300 by five percentage points for F1 (F1=48±3%; Fig. 6D). HBI also improved performance for each ligand class (F1=29±6%, 24±7%, and 41±4% for binding to metal ion, nucleic acid, or small molecule, respectively; Fig. 6A-C). Performance improved for all ligand classes and for all performance measurements except for the precision in predicting nucleic acid binding (Fig. 6B). The performance of bindEmbed21DL was limited by the low CovNoBind (Eqn. 9), especially for metal ions and small molecules (Tables S3 & S7), i.e., many proteins were predicted to bind to those ligand classes while not annotated to bind. Combining the ML method with HBI increased CovNoBind for all three ligand classes, while CovOneBind (Eqn. 8) dropped slightly (Table S6). Since the drop in CovOneBind was largest for nucleic acids, this could also explain the drop in performance of bindEmbed21 compared to only the ML method, because precision is set to zero for proteins annotated but not predicted to bind to a ligand class.

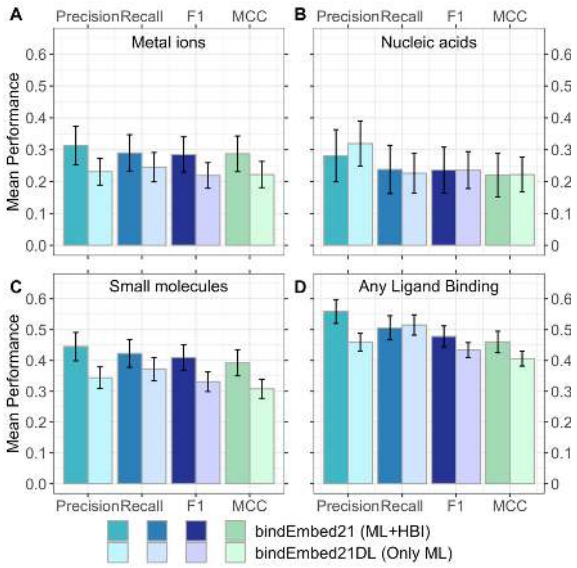

Fig. 6: Best performance by combining ML and HBI. We combined homology-based inference (HBI) and Machine Learning (ML) by transferring annotations between homologs (E-value<10^{-3}) if available and running *de novo* ML predictions using bindEmbed21DL, otherwise. This combination improved performance for the prediction of whether a residue binds to a certain ligand class for **A.** metal ions, **B.** nucleic acids, **C.** small molecules, and **D.** the combined, unspecific prediction of binding any of those three ligand classes vs. non-binding any of the three. The final version of bindEmbed21 achieved F1=29±6%, F1=24±7%, and F1=41±% for metal ions, nucleic acids, and small molecules, respectively. Lighter colored bars indicate the performance for the ML method, darker colors indicate the performance for the combination of ML and HBI.

Prediction for complete human proteome discovered unknown binding residues. Of the 20,386 sequences (corresponding to 11,362,967 residues) currently deposited as the human proteome to Swiss-Prot[35], only 3,121 (15%) had any structure with binding annotations available in BioLiP (Table 2, Table S7). Using our protocol for HBI (transfer binding annotations of local alignment if E-value $\leq 10^{-3}$) allowed inference of binding residues for another 7,199 proteins pushing the annotations of experiment + HBI to 51% (Table 2, Table S7). This number rose to 54% if we applied a less strict E-value cutoff of 1. Although most proteins likely bind ligands to function correctly, many of those remain obscure (on top the above statistics completely under-estimated the lack of knowledge by considering a single binding annotation as "protein covered" although 80% of the proteins have several domains[36,37]). Due to speed, applicability to three main ligand classes, and performance, bindEmbed21DL bridged this sequence-annotation gap predicting binding for 92% of the human proteins, for 42% of all human proteins (8,510), no binding information had been available without our prediction (Table 2, Table S7) and 21% of those 8,510 (1,751) were predicted reliably (probability≥0.95 corresponding to >73% precision, Table S5). In addition, for 21% of the proteins with experimental or HBI-inferred annotations, bindEmbed21DL provided highly reliable binding predictions previously unknown.

As seen for the example of the human proteome, binding annotations are far from complete leading to two major observations: (1) fast and generally applicable prediction methods such as bindEmbed21DL are an important tool for the identification of new binding residues and ligands that could guide future experiments, and (2) our performance estimates are most likely too conservative because the assumption that all residues not annotated as binding are non-binding was possibly wrong. In fact, while 48,700 residues were annotated as binding in structures with a resolution ≤2.5Å, an additional 21,057 residues were predicted as binding with a probability≥0.95.

Assuming that 15,372 of those are correct (precision at 0.95 is 73%, Table S5), our current set of annotations is likely missing 24% of binding residues.

Given its speed, bindEmbed21DL could also be easily applied to other complete proteomes. Predictions for all human proteins were completed within 80 minutes using one single Xeon machine with 400GB RAM, 20 cores and a Quadro RTX 8000 GPU with 48GB vRAM (40 minutes for the generation of the embeddings, 40 minutes for the predictions), i.e., generating binding residue predictions for one protein sequence took around 0.2 seconds allowing fast predictions for large sets of proteins.

Table 2: Binding predictions for complete human proteome. *

Method	Nprot	Pprot	Cumulative
BioLiP/PDB	3,121	15%	15%
(bindEmbed21)HBI	9,694	48%	51%
HBI_error prone	10,526	52%	52%
bindEmbed21DL reliable	5,962	29%	60%
bindEmbed21DL all	18,663	92%	93%

* **Method**: BioLiP/PDB: experimental annotations, (bindEmbed21)HBI: homology-based inference at EVAL≤10-3 integrated into bindEmbed21, HBI error-prone: HBI at EVAL≤1, bindEmbed21DL-reliable: probability ≥0.95 with expected precision >73%, *bindEmbed21DL-all*: prediction at probability≥0.5 (default threshold); **Data**: human proteome from Swiss-Prot[35] with 20,386 proteins; **Nprot**: number of proteins; **Pprot**: percentage of proteins; **Cumulative**: cumulative percentage, assuming the hierarchy: experimental, HBI, DL.

Availability. The data set including predictions for the human proteome, the source code, and the trained model are available via GitHub (https://github.com/Rostlab/bindPredict). Embeddings can be generated using the bio_embeddings pipeline[38]. In addition, *bindEmbed21DL* is publicly available as a standalone method as part of bio_embeddings.

Conclusion

We proposed a new method, *bindEmbed21*, predicting whether a residue in a protein sequence binds to a metal ion, a nucleic acid (DNA or RNA), or a small molecule. The method combines homology-based inference (HBI: *bindEmbed21HBI*) with Artificial Intelligence (AI), in particular using deep learning (DL: *bindEmbed21DL*). *bindEmbed21DL* neither relied on knowledge of protein structure nor on expert-crafted features, nor on evolutionary information derived from multiple sequence alignments (MSAs). Instead, we inputted *embeddings* from the pre-trained protein Language Model (pLM) ProtT5[28] into a two-layer CNN. The major problem with experimental data is the lack thereof: high-resolution data was available for fewer than 1,100 non-redundant proteins from any organism. Given the data sparsity, it is likely that many binding residues remain unknown even in the subset of 1,100 proteins with experimental data. Nevertheless, our evaluation equated “not observed” with “not binding”, treating predictions of non-observed binding as false positives. Although apparently blatantly underestimating precision, this crude simplification was needed to avoid over-prediction: methods only considering “what fraction of the experimental annotations is predicted?” (Recall, Eqn. 1) tend to optimize recall. The simplest non-sense path toward that end of “always predict binding” was carefully steered clear off by bindEmbed21DL which outperformed its MSA-based predecessor, bindPredictML17[5], by 13 percentage points (Fig. 2A) and appeared competitive with the DNA- and RNA-prediction expert MSA-based method ProNA2020[17] and the zinc-binding prediction method PredZinc[18] (Fig. 3). Prediction strength correlated with performance (Fig. 4), e.g., of the one third of all binding residues predicted with a probability ≥0.84, 59% corresponded to experimentally known binding annotations available today (Table S5). Detailed analysis of very reliable predictions not matching known experimental annotations revealed that bindEmbed21DL correctly predicted binding residues which were not annotated in the high-resolution structure used for development (Fig. 5). The analysis of predictions for the entire human proteome underlined that most binding annotations remain unknown today (51% with binding annotations through experiments or homology) and that bindEmbed21 can help in identifying new potential binding sites (Table 2, Table S7). The proteome analysis also suggested our performance estimates to be much too conservative: for all carefully investigated case studies when bindEmbed21DL reliably predicted ligands that had not been observed, we found evidence that bindEmbed21DL was right and that some experimental evidence had been overlooked, missing, or dubious. We combined the best from both worlds, namely AI/ML and HBI, to simplify predictions for users and to optimally decide when to use which (Fig. 6). The new method, *bindEmbed21*, is freely available, blazingly simple and fast, and apparently outperformed our estimates.

Materials & Methods

Data sets. Protein sequences with annotations of binding residues were extracted from BioLiP[9]. BioLiP provides binding annotations for residues based on structural information from the Protein Data Bank (PDB)[29], i.e., proteins for which several PDB structures with different identifiers exist may have multiple binding annotations. To obtain binding annotations, we extracted and combined (union) all binding information from BioLiP for all chains of PDB structures matching a given sequence, which have been determined through X-ray crystallography[39] with a resolution of ≤2.5Å (≤0.25nm). All residues not annotated as binding were considered non-binding.

BioLiP distinguishes four different ligand classes: metal ions, nucleic acids (i.e., DNA and RNA), small ligands, and peptides (protein-protein interactions). Here, we focused on the first three, i.e., on predicting the binding of metal ions, nucleic acids, or small ligands (excluding peptides). At point of accession (26-11-2019), BioLiP annotated 104,733 structures with high enough resolution and binding annotations which could be mapped to 14,894 sequences in UniProt[35]. This set was redundancy reduced using UniqueProt[40] with an HVAL<0 (corresponding to no pair of proteins in the data set having over 20% pairwise sequence identity over 250 aligned residues[41,42]; more details about the data set in Table S8 and about the redundancy reduction in Section 2.1 of the Supporting Online Material (**SOM**)). The final set of 1,314 proteins was split into a development set with 1,014 proteins (called *DevSet1014* with 13,999 binding residues, 156,684 non-binding residues; Table S8) used for optimizing model parameters and hyperparameters (after another split into training and validation/cross-training), and test set with 300 proteins (named *TestSet300* with 5,869 binding residues, 56,820 non-binding residues; Table S8) which was frozen because it had been used by other methods that we compared performance to.

In addition, we created a new and independent test set by extracting all sequences with binding annotations which were added to BioLiP after our first data set had been built (deposited between 26 November 2019 and 03 August 2021). This yielded a promising 1,592 proteins. However, upon redundancy reduction with HVAL<0 (HVAL(P,Q)<0 for all pairs of proteins P and Q within new set and between the new and the original sets) melted down to 46 proteins with 575 binding and 5,652 non-binding residues (named *TestSetNew46*; Table S8). These numbers imply two interesting findings: Firstly, about 17 experiments with binding data have been published every week over the last 91 weeks. Secondly, only one experiment provides completely new insights into binding of residues not previously characterized (3% of all). These observations underscored the importance of complementing experimental with *in silico* predictions.

Protein representation and transfer learning. We used ProtT5-XL-UniRef50[28] (in the following *ProtT5*) to create fixed-length vector representations for each residue in a protein sequence. The protein Language Model (pLM) ProtT5 was trained on BFD[43] with 2.1 billion protein sequences and fine-tuned on UniRef50[35] with 45 million protein sequences.

ProtT5 is built in analogy to the NLP (Natural Language Processing) T5[44], a Transformer-based model[45] that stacks multiple attention layers[46] to perform an all-against-all comparison between all input tokens (for ProtT5: all residues within one protein sequence) to compute a weighted sum for each residue against all other residues in the protein sequence. This mechanism is used to reconstruct corrupted input tokens (for ProtT5: single residues) from the non-corrupted sequence context (for ProtT5: the non-corrupted part of the protein sequence). After this so-called pre-training step, features learned by the pLM can be transferred to any (prediction) task requiring numerical protein representations by extracting vector representations

for single residues from the hidden states of the pLM (transfer learning). As ProtT5 was only trained on reconstructing corrupted input tokens from unlabeled protein sequences, there is no risk of information leakage or overfitting to a certain label during pre-training. To predict whether a residue is binding a ligand or not, we extracted 1024-dimensional vectors for each residue from the last hidden layer of the ProtT5 model (Fig. S6, Step 1) without fine-tuning it (no gradient was backpropagated to ProtT5).

AI/Deep Learning architecture. For *bindEmbed21DL*, we realized the 2nd level supervised learning through a relatively shallow (few free parameters) two-layer Convolutional Neural Network (CNN; Fig. S6, Step 2). The CNN was implemented in PyTorch[47] and trained with the following settings: Adamax optimizer, learning rate: 0.01, early stopping, and a batch size of 406 (resulting in two batches). The ProtT5 embeddings which consisted of the last layer of ProtT5 corresponding to a vector of 1024 dimensions per residue were used as the only input. The first CNN layer consisted of 128 feature channels with a kernel (sliding window) size of k=5 mapping the input of size L x 1024 to an output of L x 128. The second layer created the final predictions by applying a CNN with k=5 and three feature channels resulting in an output of size L x 3, one channel per ligand class. A residue was considered as non-binding if all output probabilities were < 0.5. The two CNN layers were connected through an exponential linear unit (ELU)[48] and a dropout layer[49], with a dropout rate of 70%.

To adjust for the substantial class imbalance between binding (8% of residues) and non-binding (92%), we weighted the cross-entropy loss function. Individual weights were assigned for each ligand class and were optimized to maximize performance in terms of F1 score (Eqn. 3) and MCC (Eqn. 4). Higher weights in the loss function increased recall (Eqn. 1), lower weights increased precision (Eqn. 2). The final weights were 8.9, 7.7, and 4.4 for binding metal ions, nucleic acids, and small molecules, respectively.

Homology-based inference. Homology-based inference (or homology-based annotation transfer; HBI) proceeds as follows: Given a query protein Q of unknown binding and a protein E for which some binding residues are experimentally known, align Q and E; if the two have significant sequence similarity (SIM(Q,E)>T), transfer annotations from E to Q. The threshold T and the optimal way to measure the sequence similarity (SIM) are typically determined empirically. Most successful *in silico* predictions of function are predominantly based on homology-based inference[4,8,50-55]. We aligned all proteins with MMseqs2[56], creating evolutionary profiles for each protein (family) (two MMseqs2 iterations, at E-value $\leq 10^{-3}$) against a 80% non-redundant database combining UniProt[35] and PDB[29] adapting a standard protocol based on PSI-BLAST[57] which was implemented for other methods before[17,22,51]. The resulting profiles were then aligned at E-value $\leq 10^{-3}$ against a set of proteins with experimentally known binding annotations. To save resources, we redundancy reduced this set at 95% (PIDE(x,y)<95% for all protein pairs x, y). For performance estimates, self-hits were excluded. From all hits, the local alignment with the lowest E-value and highest pairwise sequence identity (PIDE) to the query was chosen. If this hit contained any binding annotations in the aligned region, binding annotations were transferred between aligned positions and all non-aligned positions in the query were considered as non-binding. If no binding annotations were located in the aligned region, the hit was discarded and no inference of binding annotations through homology was performed. Combining bindEmbed21HBI with the ML method bindEmbed21DL led to our final method, *bindEmbed21*.

Performance evaluation. To assess whether a prediction was correct or not, we used the following standard annotations: True positives (TP) were residues correctly predicted as binding, false positives (FP) were incorrectly predicted as binding, true negatives (TN) were correctly predicted as non-binding, and false negatives (FN) were not predicted as binding while being annotated as binding. Based on this classification for each residue, we evaluated performance using standard performance measurements, namely recall (or sensitivity, Eqn. 1), precision (Eqn. 2), F1 score (Eqn. 3), and Matthews Correlation Coefficient (MCC, Eqn. 4).

$$Recall = \frac{TP}{TP+FN} \quad \text{(Eqn. 1)}$$

$$Precision = \frac{TP}{TP+FP} \quad \text{(Eqn. 2)}$$

$$F1 = 2 \cdot \frac{Recall \cdot Precision}{Recall+Precision} \quad \text{(Eqn. 3)}$$

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad \text{(Eqn. 4)}$$

Negative recall (Eqn. 5), negative precision (Eqn. 6), and negative F1 score (Eqn. 7) focusing on the negative class, i.e., non-binding residues, could be defined analogously:

$$Negative\ Recall = \frac{TN}{TN+FP} \quad \text{(Eqn. 5)}$$

$$Negative\ Precision = \frac{TN}{TN+FF} \quad \text{(Eqn. 6)}$$

$$Negative\ F1 = 2 \cdot \frac{Negative\ Recall \cdot Negative\ Precision}{Negative\ Recall+Negative\ Precision} \quad \text{(Eqn. 7)}$$

The measure *CovOneBind* (Eqn. 8) indicated the fraction of proteins for which at least one residue was predicted as binding. Accordingly, the inverse of this, the *CovNoBind* (Eqn. 9), indicated the fraction of proteins for which predictions as well as experiments detected no binding. Since our data set only consisted of proteins with a binding site, *CovNoBind* had to be computed for different classes of ligands, i.e., the fraction of proteins for which ligand *l* was neither observed nor predicted (Eqn. 9).

$$CovOneBind = \frac{Number\ of\ proteins\ with\ one\ binding\ residue\ predicted}{Number\ of\ proteins\ with\ binding\ annotations} \quad \text{(Eqn. 8)}$$

$$CovNoBind(l) = \frac{Number\ of\ proteins\ without\ binding\ predictions\ for\ ligand\ l}{Number\ \ of\ proteins\ without\ binding\ annotations\ for\ ligand\ l} \quad \text{(Eqn. 9)}$$

When predicting whether a residue binds a specific ligand class or not, a false positive prediction for a certain ligand class could result from three cases: a residue (i) not binding anything, (ii) binding another ligand, or (iii) not known to bind, yet. To capture (ii), we calculated the number of cross-predictions to any other ligand class (confusion table), i.e., how many residues were predicted to bind ligand class *l* while experimentally observed to bind to ligand class *m*.

Each performance measure was calculated for each protein individually. Then the mean was calculated over the resulting distribution and symmetric 95% confidence intervals (CI) assuming a normal distribution of the performance values were calculated as error estimates.

Reliability Index. We transformed the probability *p* into a single-digit integer reliability index (RI) ranging from 0 (unreliable; probability=0.5) to 9 (very reliable; probability=1.0 for binding and probability=0.0 for non-binding) (Eqn. 10).

$$\mathrm{RI(p)} = \begin{cases} (0.5 - p) \cdot \frac{9}{0.5} & if\ p < 0.5 \\ (p - 0.5) \cdot \frac{9}{0.5} & if\ p \geq 0.5 \end{cases} \qquad \text{(Eqn. 10)}$$

Comparison to other methods. *bindPredictML17*[5] predicts binding residues from enzymes (trained on the PDB) and DNA-binding residues from PDIdb[30]. Queried with protein sequences, the method first builds multiple sequence alignments, and uses those to compute evolutionary couplings[21] and effect predictions[19,20]. Those two main features, in turn, are used as input to the machine learning method.

ProNA2020[17] predicts binding to DNA, RNA, and other proteins using a two-step procedure: The first per-protein level predicts whether a protein binds DNA, RNA, or another protein. For proteins that bind to other proteins, DNA, or RNA, the second per-residue level predicts which residue binds to any (or all) of the three ligand classes. ProNA2020 combines homology-based inference and machine learning using motif-based profile-kernel[58,59] and word-based approaches (ProtVec)[60] for the per-protein prediction and uses standard neural networks with different expert-crafted features taken from PredictProtein[22] as input.

PredZinc[18] predicts binding to zinc ions using a combination of homology-based inference and a Support Vector Machine (SVM). The SVM was trained on feature vectors representing the conservativity and physicochemical properties of single amino acids and pairs of amino acids.

Acknowledgements

Thanks to Tim Karl and Inga Weise (both TUM) for invaluable help with technical and administrative aspects of this work. Last, but not least, thanks to all those who maintain public databases in particular Steven Burley (PDB, Rutgers), Ioannis Xenarios (Swiss-Prot, SIB, Geneva) and Yang Zhang (BioLiP, University of Michigan) and their crews, and to all experimentalists who enabled this analysis by making their data publicly available. This work was supported by the Bavarian Ministry of Education through funding to the TUM and by a grant from the Alexander von Humboldt foundation through the German Ministry for Research and Education (BMBF: Bundesministerium für Bildung und Forschung), by two grants from BMBF (031L0168 and program “Software Campus 2.0 (TUM) 2.0” 01IS17049) as well as by a grant from Deutsche Forschungsgemeinschaft (DFG-GZ: RO1320/4-1).

REFERENCES

1 Whisstock, J. C. & Lesk, A. M. Prediction of protein function from protein sequence and structure. *Q Rev Biophys* **36**, 307-340, doi:10.1017/s0033583503003901 (2003).

2 Alberts, B. *et al. Molecular Biology of the Cell*. (Garland Science, Taylor and Francis Group, 2018).

3 Schmidt, T., Haas, J., Gallo Cassarino, T. & Schwede, T. Assessment of ligand-binding residue predictions in CASP9. *Proteins* **79 Suppl 10**, 126-136, doi:10.1002/prot.23174 (2011).

4 Radivojac, P. *et al.* A large-scale evaluation of computational protein function prediction. *Nat Methods* **10**, 221-227, doi:10.1038/nmeth.2340 (2013).

5 Schelling, M., Hopf, T. A. & Rost, B. Evolutionary couplings and sequence variation effect predict protein binding sites. *Proteins* **86**, 1064-1074, doi:10.1002/prot.25585 (2018).

6 Qiu, J., Nechaev, D. & Rost, B. Protein-protein and protein-nucleic acid binding residues important for common and rare sequence variants in human. *BMC Bioinformatics* **21**, 452, doi:10.1186/s12859-020-03759-0 (2020).

7 Mahlich, Y. *et al.* Common sequence variants affect molecular function more than rare variants? *Science Reports* **7**, 1608, doi:10.1038/s41598-017-01054-2 (2017).

8 Hamp, T. *et al.* Homology-based inference sets the bar high for protein function prediction. *BMC Bioinformatics* **14 Suppl 3**, S7, doi:10.1186/1471-2105-14-S3-S7 (2013).

9 Yang, J., Roy, A. & Zhang, Y. BioLiP: a semi-manually curated database for biologically relevant ligand-protein interactions. *Nucleic Acids Res* **41**, D1096-1103, doi:10.1093/nar/gks966 (2013).

10 Yang, J., Roy, A. & Zhang, Y. Protein-ligand binding site recognition using complementary binding-specific substructure comparison and sequence profile alignment. *Bioinformatics* **29**, 2588-2595, doi:10.1093/bioinformatics/btt447 (2013).

11 Zhang, C., Freddolino, P. L. & Zhang, Y. COFACTOR: improved protein function prediction by combining structure, sequence and protein-protein interaction information. *Nucleic Acids Res* **45**, W291-W299, doi:10.1093/nar/gkx366 (2017).

12 Brylinski, M. & Skolnick, J. A threading-based method (FINDSITE) for ligand-binding site prediction and functional annotation. *Proc Natl Acad Sci U S A* **105**, 129-134, doi:10.1073/pnas.0707684105 (2008).

13 Capra, J. A., Laskowski, R. A., Thornton, J. M., Singh, M. & Funkhouser, T. A. Predicting protein ligand binding sites by combining evolutionary sequence conservation and 3D structure. *PLoS Comput Biol* **5**, e1000585, doi:10.1371/journal.pcbi.1000585 (2009).

14 Xia, C. Q., Pan, X. & Shen, H. B. Protein-ligand binding residue prediction enhancement through hybrid deep heterogeneous learning of sequence and structure data. *Bioinformatics* **36**, 3018-3027, doi:10.1093/bioinformatics/btaa110 (2020).

15 Cui, Y., Dong, Q., Hong, D. & Wang, X. Predicting protein-ligand binding residues with deep convolutional neural networks. *BMC Bioinformatics* **20**, 93, doi:10.1186/s12859-019-2672-1 (2019).

16 Hu, X., Dong, Q., Yang, J. & Zhang, Y. Recognizing metal and acid radical ion-binding sites by integrating ab initio modeling with template-based transferals. *Bioinformatics* **32**, 3260-3269, doi:10.1093/bioinformatics/btw396 (2016).

17 Qiu, J. *et al.* ProNA2020 predicts protein-DNA, protein-RNA, and protein-protein binding proteins and residues from sequence. *J Mol Biol* **432**, 2428-2443, doi:10.1016/j.jmb.2020.02.026 (2020).

18 Shu, N., Zhou, T. & Hovmoller, S. Prediction of zinc-binding sites in proteins from sequence. *Bioinformatics* **24**, 775-782, doi:10.1093/bioinformatics/btm618 (2008).

19 Hopf, T. A. *et al.* Mutation effects predicted from sequence co-variation. *Nat Biotechnol* **35**, 128-135, doi:10.1038/nbt.3769 (2017).

20 Hecht, M., Bromberg, Y. & Rost, B. Better prediction of functional effects for sequence variants. *BMC Genomics* **16 Suppl 8**, S1, doi:10.1186/1471-2164-16-S8-S1 (2015).

21 Marks, D. S., Hopf, T. A. & Sander, C. Protein structure prediction from sequence variation. *Nat Biotechnol* **30**, 1072-1080, doi:10.1038/nbt.2419 (2012).

22 Bernhofer, M. *et al.* PredictProtein - Predicting Protein Structure and Function for 29 Years. *Nucleic Acids Res*, doi:10.1093/nar/gkab354 (2021).

23 Nair, R., Carter, P. & Rost, B. NLSdb: database of nuclear localization signals. *Nucleic Acids Research* **31**, 397-399 (2003).

24 Ofran, Y., Mysore, V. & Rost, B. Prediction of DNA-binding residues from sequence. *Bioinformatics* **23**, i347-353 (2007).

25 Ofran, Y. & Rost, B. Predicted protein-protein interaction sites from local sequence information. *FEBS Letters* **544**, 236-239 (2003).

26 Peng, Z. & Kurgan, L. High-throughput prediction of RNA, DNA and protein binding regions mediated by intrinsic disorder. *Nucleic Acids Res* **43**, e121, doi:10.1093/nar/gkv585 (2015).

27 Schlessinger, A., Ofran, Y., Yachdav, G. & Rost, B. Epitome: Database of structure-inferred antigenic epitopes. *Nucleic Acids Research* **34**, D777-780 (2006).

28 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* **PP**, doi:10.1109/TPAMI.2021.3095381 (2021).

29 Burley, S. K. *et al.* RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. *Nucleic Acids Research* **47**, D464-D474, doi:10.1093/nar/gky1004 (2019).

30 Norambuena, T. & Melo, F. The Protein-DNA Interface database. *BMC Bioinformatics* **11**, 262, doi:10.1186/1471-2105-11-262 (2010).

31 Decanniere, K., Babu, A. M., Sandman, K., Reeve, J. N. & Heinemann, U. Crystal structures of recombinant histones HMfA and HMfB from the hyperthermophilic archaeon Methanothermus fervidus. *J Mol Biol* **303**, 35-47, doi:10.1006/jmbi.2000.4104 (2000).

32 Mattiroli, F. *et al.* Structure of histone-based chromatin in Archaea. *Science* **357**, 609-612, doi:10.1126/science.aaj1849 (2017).

33 Madrigal-Carrillo, E. A., Diaz-Tufinio, C. A., Santamaria-Suarez, H. A., Arciniega, M. & Torres-Larios, A. A screening platform to monitor RNA processing and protein-RNA interactions in ribonuclease P uncovers a small molecule inhibitor. *Nucleic Acids Res* **47**, 6425-6438, doi:10.1093/nar/gkz285 (2019).

34 Reiter, N. J. *et al.* Structure of a bacterial ribonuclease P holoenzyme in complex with tRNA. *Nature* **468**, 784-789, doi:10.1038/nature09516 (2010).

35 The UniProt Consortium. UniProt: the universal protein knowledgebase in 2021. *Nucleic Acids Res* **49**, D480-D489, doi:10.1093/nar/gkaa1100 (2021).

36 Liu, J. & Rost, B. Domains, motifs, and clusters in the protein universe. *Current Opinion in Chemical Biology* **7**, 5-11 (2003).

37 Liu, J. & Rost, B. CHOP proteins into structural domain-like fragments. *Proteins: Structure, Function, and Bioinformatics* **55**, 678-688 (2004).

38 Dallago, C. *et al.* Learned embeddings from deep learning to visualize and predict protein sets. *Curr Protoc* **1**, e113, doi:10.1002/cpz1.113 (2021).

39 Smyth, M. S. & Martin, J. H. x ray crystallography. *Mol Pathol* **53**, 8-14, doi:10.1136/mp.53.1.8 (2000).

40 Mika, S. & Rost, B. UniqueProt: Creating representative protein sequence sets. *Nucleic Acids Res* **31**, 3789-3791, doi:10.1093/nar/gkg620 (2003).

41 Sander, C. & Schneider, R. Database of homology-derived structures and the structural meaning of sequence alignment. *Proteins: Structure, Function, and Genetics* **9**, 56-68 (1991).

42 Rost, B. Twilight zone of protein sequence alignments. *Protein Engineering* **12**, 85-94 (1999).

43 Steinegger, M., Mirdita, M. & Söding, J. Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nat Methods* **16**, 603-606, doi:10.1038/s41592-019-0437-4 (2019).

44 Raffel, C. *et al.* Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* **21**, 1-67 (2020).

45 Vaswani, A. *et al.* Attention is All you Need in *Neural Information Processing Systems Conference.* (eds I Guyon *et al.*) 5998-6008 (Curran Associates, Inc.).

46 Bahdanau, D., Cho, K. H. & Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate in *arXiv.*

47 Paszke, A. *et al.* PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems* **32** (2019).

48 Clevert, D.-A., Unterthiner, T. & Hochreiter, S. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *arXiv preprint arXiv:1511.07289* (2015).

49 Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *The Journal of Machine Learning Research* **15**, 1929-1958 (2014).

50 Friedberg, I. & Radivojac, P. Community-Wide Evaluation of Computational Function Prediction. *Methods Mol Biol* **1446**, 133-146, doi:10.1007/978-1-4939-3743-1_10 (2017).

51 Goldberg, T. *et al.* LocTree3 prediction of localization. *Nucleic Acids Res* **42**, W350-355, doi:10.1093/nar/gku396 (2014).

52 Jiang, Y. *et al.* An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biol* **17**, 184, doi:10.1186/s13059-016-1037-6 (2016).

53 Ofran, Y., Punta, M., Schneider, R. & Rost, B. Beyond annotation transfer by homology: novel protein-function prediction methods to assist drug discovery. *Drug Discovery Today* **10**, 1475-1482 (2005).

54 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* **20**, 244, doi:10.1186/s13059-019-1835-8 (2019).

55 Littmann, M., Heinzinger, M., Dallago, C., Olenyi, T. & Rost, B. Embeddings from deep learning transfer GO annotations beyond homology. *Sci Rep* **11**, 1160, doi:10.1038/s41598-020-80786-0 (2021).

56 Steinegger, M. & Söding, J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat Biotechnol* **35**, 1026-1028, doi:10.1038/nbt.3988 (2017).

57 Altschul, S. F. *et al.* Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* **25**, 3389-3402, doi:10.1093/nar/25.17.3389 (1997).

58 Kuang, R. *et al.* Profile-based string kernels for remote homology detection and motif extraction. *J Bioinform Comput Biol* **3**, 527-550, doi:10.1142/s021972000500120x (2005).

59 Hamp, T., Goldberg, T. & Rost, B. Accelerating the Original Profile Kernel. *PLoS One* **8**, e68459, doi:10.1371/journal.pone.0068459 (2013).

60 Asgari, E. & Mofrad, M. R. Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. *PLoS One* **10**, e0141287, doi:10.1371/journal.pone.0141287 (2015).