scientific reports

Check for updates

OPEN

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[1,3,4,5]

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 (pLM) ProtT5 as input. Using only single sequences without creating multiple sequence alignments (MSAs), *bindEmbed21DL* outperformed MSA-based predictions. Combination with homology-based inference increased performance to F1 = 48 ± 3% (95% CI) and MCC = 0.46 ± 0.04 when merging all three ligand classes into one. All results were confirmed by three independent data sets. Focusing on very reliably predicted residues could complement experimental evidence: For the 25% most strongly predicted binding residues, at least 73% were correctly predicted even when ignoring the problem of missing experimental annotations. 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 and predicted about 6% of all residues as binding to metal ions, nucleic acids, or small molecules.

Abbreviations

AI	Artificial intelligence (expanding ML through deep learning, i.e., using more free parameters)
BFD	Big Fantastic Database (large database of protein sequences)
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

Experimental data for protein binding remains limited. Knowing protein function is crucial to understand the molecular mechanisms of life[1]. For most proteins, function 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 binding sites resembles the diversity of the ligands, binding sites are highly specific and often determined by a few key residues[2]. Binding residues are

[1]Department of Informatics, Bioinformatics and Computational Biology, I12, TUM (Technical University of Munich), 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, Garching, 85748 Munich, Germany. [4]TUM School of Life Sciences Weihenstephan (TUM-WZW), Alte Akademie 8, Freising, Germany. [5]Department of Biochemistry and Molecular Biophysics, Columbia University, 701 West, 168th Street, New York, NY 10032, USA. ✉email: littmann@rostlab.org

experimentally determined most reliably through high-resolution structures of the protein in complex with the ligand marking residues close to this ligand as binding residues (e.g., $\leq$ 5Å)[3].

Prediction methods usually rely on MSAs. Despite immense progress in quantitative high-throughput proteomics, experimentally verified binding residues remain unknown for most proteins[4]. In fact, reliable data remain so sparse to even challenge Machine Learning (ML) models with fewer parameters than tools from Artificial Intelligence (AI)[5]. Thus, reliable prediction methods importantly bridge, e.g., studying the effect of sequence variation in human populations[6,7]. Homology-based inference (HBI) transfers e.g., binding residues from sequence-similar proteins with known annotations to uncharacterized proteins[5,8]. Although accurate, HBI is only applicable to the few proteins for which a sequence-similar protein with binding annotations exists. If unavailable, de novo prediction methods based on ML try to fill the gap. HBI (or template-based methods) usually outperforms sequence-based (or de novo) methods[9,10], but they rely on the existence of structurally similar proteins with experimentally verified binding annotations (structural template)[10–14]. Notably, *AlphaFold 2* that solved the protein structure prediction problem[15] might be the first AI-based prediction method consistently outperforming template-based solutions. AlphaFold 2 heavily relies on information from multiple sequence alignments (MSAs). Recent structure predictions without MSAs remain less accurate[16]. It remains unclear to which extent structure predictions could improve binding prediction beyond stepping up from binding residues to binding sites.

Some template-based methods also require substantial computing resources, e.g., COACH[10] is an ensemble classifier combining five individual approaches and has been considered the state-of-the-art (SOTA) for binding residue prediction for many years[17,18]. One protein prediction took about 10 h on their webserver, while local installations require 60 GB free disk space. Although neither aspect renders the method unusable, both limit ease of access to predictions and comparisons. 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[5,15,17,19,20]. Our previous method bindPredictML17[5] predicted binding residues for enzymes and DNA-binding proteins relying mainly on information from sequence variation[21,22] and co-evolving residues[23], both requiring the time-consuming computation of MSAs. Another method, ProNA2020[19], uses MSAs and various features from PredictProtein[24] to predict protein–protein, protein–DNA, and protein–RNA binding. In addition to the complexity of their input features, many methods specialize on specific ligands or sets thereof[5,14,18–20,25–27]. For instance, PredZinc[20] only predicts zinc ions and IonCom[18] provides predictions for 13 metals and for four radical ion ligands. Most existing sufficiently reliable sequence-based methods cannot be applied to generic proteome-wide binding predictions due to restrictions in computational resources or to limited sets of ligands.

Here, we propose a new method dubbed *bindEmbed21* consisting of two components (bindEmbed21DL and bindEmbed21HBI) that predict binding residues for three ligand classes. We input protein representations (fixed-length per-protein embeddings) from pre-trained protein Language Models (pLMs), in particular from ProtT5[28]. Using only those embeddings, bindEmbed21DL predicts residues binding to metal ions, nucleic acids (DNA and RNA), and/or regular small molecules. Combining the de novo prediction method with HBI (bindEmbed21HBI) further improved performance. Since embeddings can be easily extracted for any protein sequence, bindEmbed21 enables fast and easy predictions for all available protein sequences.

Results and discussion

Embedding-based predictions from bindEmbed21DL achieved F1 = 43%. 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 performance differed substantially between the three classes [Fig. 1, Supplementary Table S1 in Supporting Online Material (SOM)]: binding residues were predicted best (e.g., highest F1 or MCC, Eqs. 3, 4) for small molecules and worst for nucleic acids (Table 1 DevSet1014; Fig. 1A–C). Those differences might point to differences in the abundance of experimental data for each ligand class: Small molecules were the most prominent ligand class, while nucleic binding was the lowest (Supplementary Table S12). This might suggest that any class could be predicted better given more data. In fact, using a smaller training set (515 proteins) with equal numbers of proteins with small molecule as with nucleic acid binding (108 proteins) dropped performance immensely for the small molecule class (Supplementary Table S3) suggesting that better prediction of small molecule binding resulted largely from access to more experimental data. Alternatively, performance differences could be due to properties of small molecule binding being more clearly encoded in the embeddings than those of nucleic acid binding. This might render these easier to predict. However, we could neither support this speculation by explicit evidence, nor refute it and proof that only the increase in data caused better predictions. 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), indicating that many residues were correctly identified as binding residues despite confusing the ligand classes (Supplementary Table S5).

For all ligand classes, precision (Eq. 2) remained below recall (Eq. 1; Fig. 1) and the fraction of proteins for which not a single residue was predicted as binding (CovNoBind(l), Eq. 9), was low, especially for metal ions and small molecules (Supplementary Table S4). Therefore, 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). As the binary prediction (binding/not) outperformed by far the 3-class prediction (Table 1), cross-predictions (confusions between ligand classes, Supplementary Table S5) constituted one major limitation. The most common cause for prediction mistakes appeared to be over-prediction (Supplementary Table S4), but at least some of the alleged over-predictions might indicate missing observations

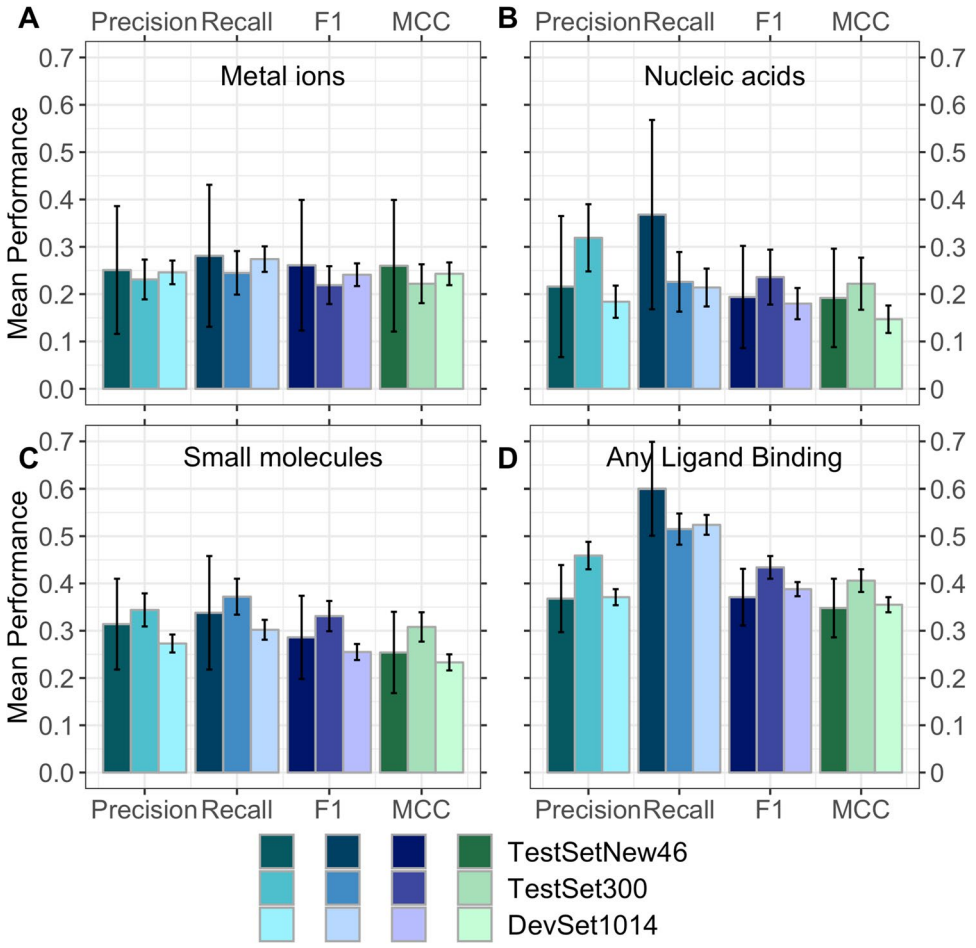

Figure 1. Performance of new method *bindEmbed21DL*. Performance captured by four per-residue measures: precision (Eq. 2), recall (Eq. 1), F1 score (Eq. 3), and MCC (Eq. 4). Data sets: *DevSet1014* (validation splits of cross-validation development, most light colors), *TestSet300* (fixed test set used 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 validation 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.

(analysis below). Remarkably, bindEmbed21DL performed similar to its binarized version solely trained on the distinction of binding vs. non-binding (Supplementary Table S6).

In a typical cross-validation split (training, validation, 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 except for binding to metal ions (Table 1, Fig. 1) although most differences were within the confidence intervals (CI; Fig. 1, Supplementary Table S1). The test set had more proteins binding to nucleic acids and small molecules than the development set, due to constraints imposed on the test set to facilitate comparisons with other methods. Those were the classes for which bindEmbed21DL reached higher values on the test than on the validation set (Fig. 1B,C). Thus, the higher numbers for the test set for nucleic acid and small molecule binding could indicate that binding residues are better defined and therefore easier to predict for enzymes than for other proteins in the development set.

To investigate, we created an independent test set from recent annotations (TestSetNew46, Methods: 46 unique from a total of 1592 new proteins). For these, *bindEmbed21DL* reached values that, within the 95% CI, agreed with both the original test and validation sets, possibly due to large CIs for the tiny new data set. 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 for the next 1592 proteins submitted for prediction ("Methods").

Embeddings clearly outperformed MSA-based predictions. One recent binding method, *bindPredictML17*[5] predicts binding residues based on MSAs. A subset of the test set (225 of the 300 proteins in

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%
Random	*TestSet300*	1 ± 1%	6 ± 2%	6 ± 1%	9 ± 1%
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
ZincBindPredict	$TestSet300_{Zinc51}$	17 ± 9%	n/a	n/a	n/a

Table 1. F1 score (harmonic mean of precision and recall). Measure: F1 (Eq. 3); ±: 95% confidence intervals (1.96 standard errors); Methods: *bindEMbed21DL*: method introduced here, *bindPredictML17*[5]: MSA-based method predicting binding, *ProNA2020*[19]: method specialized on predicting binding to DNA, RNA, and other proteins; *PredZinc*[20] and *ZincBindPredict*[29]: methods specialized on predicting zinc-binding; *Random:* random prediction by randomly shuffling the original output probabilities of bindEmbed21DL; Data: *DevSet1014*: development set (validation) set with 1014 proteins, *TestSet300*: Test set created during method 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.

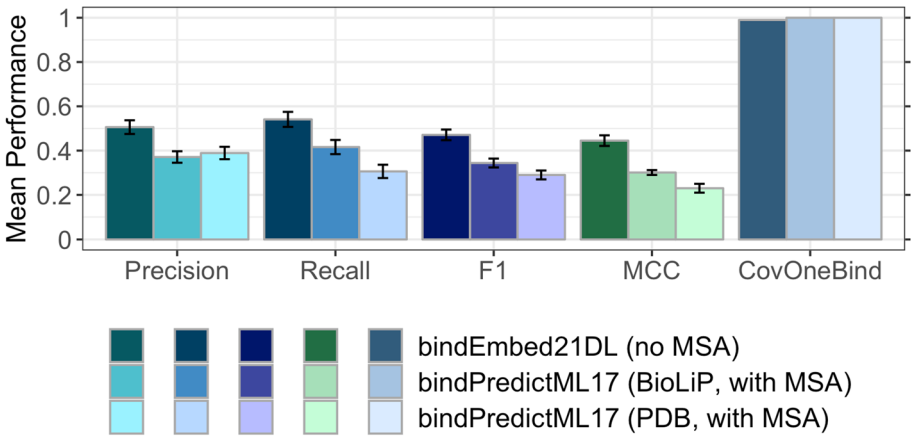

Figure 2. Embeddings outperformed MSA-based predictions. Comparison of performance between *bindPredictML17*[5] using multiple sequence alignments (MSAs) and the method introduced here, *bindEmbed21DL*, using only embeddings from ProtT5[28]. We also compare using binding annotations from BioLiP[9] or the PDB[31]. 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[31] for enzymes and PDIdb[57] for DNA-binding proteins, it reached higher performance (lighter shaded colors vs. lightest shaded colors) for BioLiP annotations. Error bars indicate 95% CIs.

TestSet300) enabled an unbiased comparison: *bindEmbed21DL* significantly (beyond 95% CI) outperformed the old MSA-based method *bindPredictML17*, e.g., raising the harmonic mean over precision and recall by 13 percentage points (Fig. 2, Table 1, bindEmbed21DL vs. bindPredictML17 last column for TestSet225). However, *bindEmbed21DL* predicted binding for only 222 of the 225 test proteins (CovOneBind = 99%, Eq. 8), while its predecessor predicted for all 225. This could be attributed to bindEmbed21DL focusing more on precision than bindPredictML17: The gain in precision was larger than the gain in recall (Fig. 2). However, higher precision reduced recall, thereby missing binding in three of 225 proteins.

bindEmbed21DL and *bindPredictML17* differed in two major aspects: (1) the annotations used for training, and (2) the usage of embeddings vs. MSA-derived input features. Both factors contributed to the improvement of bindEmbed21DL over bindPredictML17. For instance, the F1 score improved by 18 percentage points; 13 of the 18 originated from using embeddings rather than MSA-based input (Supplementary Fig. S2, SOM 1.3),

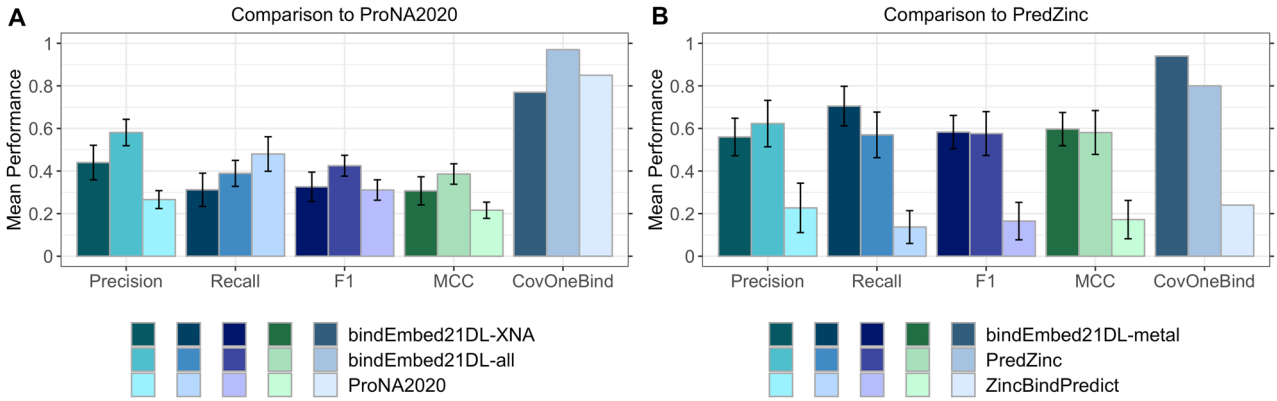

Figure 3. BindEmbed21DL competitive with specialists. (**A**) XNA binding. Data: 66 DNA- or RNA-binding (dubbed XNA) proteins from the test set *TestSet300*. ProNA2020[19] (lightest shaded bars) uses MSAs to predict DNA-, RNA-, and protein-binding, while the method introduced here uses embeddings only (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, Eq. 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 (Supplementary Table S5). (**B**) Zinc-binding. Data: 51 zinc-binding proteins from *TestSet300*. ZincBindPredict[29] (lightest shaded bars) and PredZinc[20] (darker shaded bars) predict zinc-binding; bindEmbed21DL-metal (darkest shaded bars) marked predictions for metal ions. bindEmbed21DL-metal achieved a similar performance as PredZinc, while providing predictions for more proteins (CovOneBind(bindEmbed21DL-metal) = 94% vs. CovOneBind(PredZinc) = 80%). ZincBindPredict was not competitive due to only providing predictions for 12 proteins (CovOneBind(ZincBindPredict) = 24%).

while five of 18 reflected the new annotations (Fig. 2, SOM 1.2). Thus, embeddings can significantly outperform methods explicitly using evolutionary information through MSAs.

bindEmbed21DL competitive to specialist methods. *bindEmbed21DL* predicted three ligand classes, while many state-of-the-art (SOTA) methods specialize on one ligand class or subsets thereof. For instance, *ProNA2020*[19] 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. The MSA-based method ProNA2020 shines through unifying a hierarchy of prediction tasks and outperformed all other sequence-based methods in predicting binding to DNA or RNA (dubbed XNA)[19]. 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, *ProNA2020* performed slightly worse in XNA-binding prediction than the embedding-based MSA-free *bindEmbed21DL* (Fig. 3A). However, when analyzing how many proteins had at least one residue predicted as XNA-binding (CovOneBind, Eq. 8), the situation reversed (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). This again pointed to the problem of cross-predictions (Supplementary Table S5).

PredZinc[20] and *ZincBindPredict*[29] specialize on predicting residues binding to zinc ions. 51 proteins in *TestSet300* were annotated as zinc-binding (dubbed TestSet300$_{Zinc51}$ in Table 1) and were used to compare *PredZinc* and *ZincBindPredict* to the generalist *bindEmbed21DL*. The newer method, *ZincBindPredict*, only predicted for 12 proteins (CovOneBind = 24%, Fig. 3B). Therefore, we also compared to the older method *PredZinc*. Despite having only been trained on metal-binding in general and not zinc-binding specifically, *bindEmbed21DL* matched the F1 score of *PredZinc* (Fig. 3B) with a lower precision but higher recall (Fig. 3B); it also reached a higher CovOneBind (Eq. 8) predicting for 94% instead of for 80% as *PredZinc*. Also, *bindEmbed21DL* clearly outperformed *ZincBindPredict* (Fig. 3B) through higher CovOneBind, i.e., *ZincBindPredict* is very accurate when applicable.

We evaluated specialized methods only on proteins binding to those ligand classes. In a more realistic application not knowing the ligand, specialized methods likely perform worse. Also, we may have overestimated the performance of other methods because we could not exclude their development sets. Nevertheless, *bindEmbed21DL* remained competitive on the turf of the specialists and generic enough to be applicable to three different ligand classes.

More reliable predictions better. For the binary prediction of binding vs. non-binding residues, *bindEmbed21DL* reached 37 ± 2% precision at 52 ± 2% recall (Fig. 1D) while making predictions for 1,000 of 1,014 proteins in the validation splits (*DevSet1014*; CovOneBind = 99%). These values resulted from the default threshold

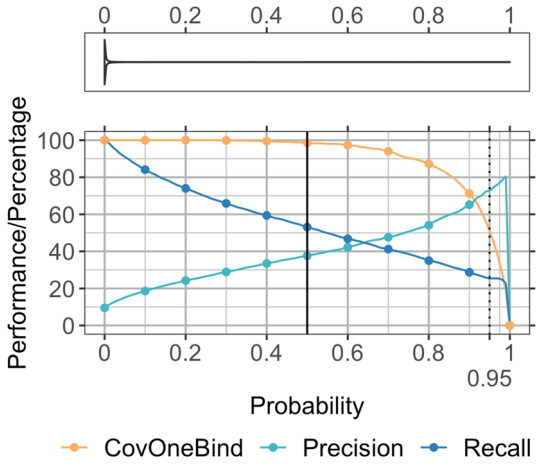

Figure 4. Residues predicted stronger more often correctly predicted. Data set: DevSet1014. Precision and recall are only shown for the proteins for which at least one residue was predicted as binding where the number of such proteins is indicated by CovOneBind. The x-axis gives the output probability 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 (marked as dashed line) 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 (Supplementary Table S7; SOM 1.5 for more details). The violin plot in the top panel reflects the actual distribution of probabilities: 50% of the residues were predicted with probability $\leq 10^{-3}$ and 75% with probability $\leq 6 \times 10^{-3}$. While we expected binding to be the more evolutionary conserved feature, non-binding residues were apparently easier to predict reliably.

($p \geq 0.5$) optimized by the ML method. If only the 1000 proteins with a prediction were considered, both precision and recall rose by one percentage point (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, Supplementary Table S7). For instance, at a cutoff of 0.28, at least one binding residue was predicted for every protein (CovOneBind = 100%) at the expense of precision dropping by nine percentage points (Fig. 4, Supplementary Table S7). On the other hand, precision could be increased by applying higher cutoffs to predict binding. For instance, for a cutoff of 0.95, precision almost doubled (Fig. 4, Supplementary Table S7). Although recall and CovOneBind decreased for higher cutoffs, *bindEmbed21DL* still predicted binding for over half of the proteins and for one fourth of all binding residues at 0.95 (Fig. 4, Supplementary Table S7). Residues falsely predicted as binding at such a high cutoff could point to yet unknown candidates for binding residues. In fact, comparing the internal representations from the first CNN layer of falsely predicted binding residues with those of correct predictions provided some evidence that highly reliable, not yet observed predictions clustered with those of experimental annotations (Supplementary Fig. S5). This confirmed the hypothesis that highly reliable binding predictions might help to identify missing binding annotations.

Missing experimental annotations limit the top precision reachable (if we equate "not observed as binding" with "non-binding"). For probability > 0.95, precision reached 80% (Fig. 4). To some extent, this estimated the effect from missing annotations: At least for the 25% most reliably predicted binding residues, at most one fifth (100–80) could be attributed to missing annotations. This might not imply that, at $p > 0.5$, precision would maximally rise by 20 percentage points because the most reliably predicted binding residues might be more likely to coincide with easy to obtain experimental data. Clearly, the opposite holds: Regions with low information, such as intrinsically disordered regions, are more difficult to predict and to experimentally resolve[30].

Alternatively, predictions could be refined taking the number of predicted residues into consideration: A low number of binding predictions in a protein indicated that those predictions were incorrect (Supplementary Fig. S6). Removing such predictions led to an increase in CovNoBind(l) (Eq. 9) while decreasing CovOneBind (Eq. 8; Supplementary Fig. S6).

Since the probability score correlated with prediction reliability, we defined a single-digit integer reliability index (RI; Eq. 10) ranging from 0 (unreliable; probability = 0.5) to 9 (very reliable). This RI empowers users, depending on their interest, either to focus on the most precise/reliable predictions for binding (or non-binding), or to focus on the perspective most likely to identify any binding residue that might exist.

Reliable predictions could help refining experimental annotations. Using a cutoff of 0.95 to classify a residue as "binding", *bindEmbed21DL* reached 73% precision with at least one residue predicted as binding for 519 proteins (CovOneBind = 51%; Fig. 4, Supplementary Table S7). For 84 of the 519 proteins (16%), none of the residues predicted that reliably (probability ≥ 0.95) had been experimentally annotated as binding. We analyzed two of those 84 in more detail.

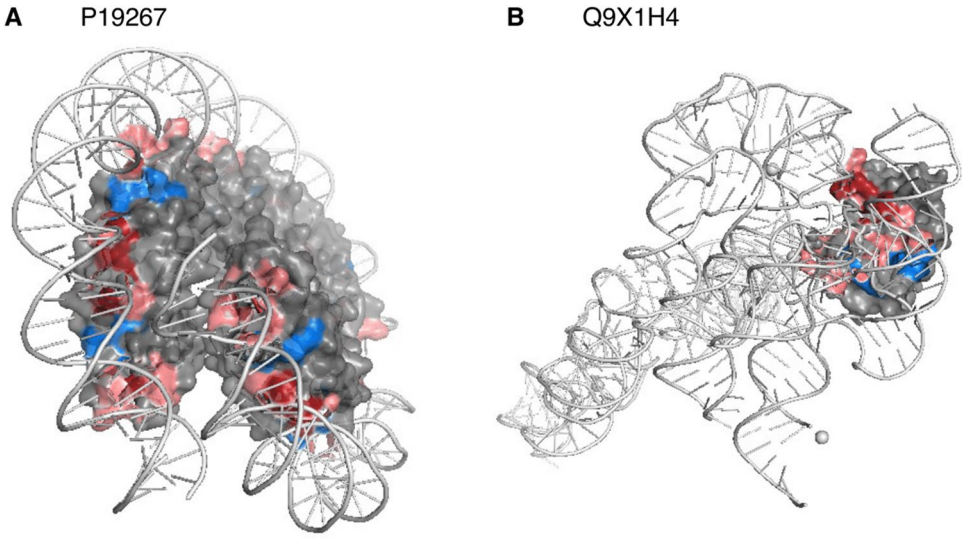

Figure 5. Annotations from low-resolution structures supported through reliable predictions. (**A**) Our development set (DevSet1014) contained the PDB structure 1A7W[31,32] for the DNA-binding protein HMf-2 (UniProt ID: P19267). No DNA/nucleic acid binding was annotated in that structure, but our new method, bindEmbed21DL, reliably predicted (probability ≥ 0.95) four residues to bind nucleic acids. Shown is the PDB structure 5T5K[31,33] for the same protein that has a resolution of 4.0 Å 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[31,34]. However, those four residues were considered as binding according to the two low-resolution structures 3Q1Q (3.8 Å)[31,35] (visualized) and 3Q1R (4.21 Å)[31,35]. 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.

The DNA-binding protein HMf-2 (UniProt ID: P19267) is annotated to bind metal with residues 34 and 38 by the experimental structure with PDB identifier (PDBid) 1A7W[31,32] resolved at 1.55 Å. None of those two were predicted as binding (at $p \geq 0.5$). Both the name and the available annotations suggested DNA-binding. If so, the observed metal-binding might point to allosteric binding. Four residues were predicted reliably (probability ≥ 0.95) to bind nucleic acids (Fig. 5A, dark red residues). For another PDB structure of this protein (PDBid 5T5K[31,33] at 4.0 Å resolution), BioLiP annotates DNA-binding for all four reliably predicted residues. Due to our threshold in minimal resolution, this structure had not been included in our data sets. Overall, BioLiP annotates 13 residues in 5T5K as DNA-binding, 10 of those were correctly predicted (77% recall; Fig. 5A, lighter red). With respect to the three remaining: although our sequence-based method clearly did not reach remotely 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): The PDB structure 6MAX[31,34] (1.42 Å) annotated this protein with seven residues binding to a small molecule; none of the those were predicted at $p \geq 0.95$. Indeed, the available functional annotations clearly suggest nucleic acid-binding; the small molecule bound in 6MAX seems to mainly inhibit RNA-binding[34]. Four residues were predicted to bind nucleic acids reliably ($p \geq 0.95$, Fig. 5B, dark red). The low-resolution structures 3Q1Q (3.8 Å)[31,35] and 3Q1R (4.21 Å)[31,35] confirmed nucleic acid-binding for this protein. All four most reliable predictions were experimentally confirmed by those structures, and of the 21 residues annotated as binding, 16 were correctly predicted by default ($p \geq 0.5$, 76% recall, Fig. 5B, lighter red).

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 because all those correct binding predictions had been used as "non-binding" during training. On the one hand, this facilitates the identification of previously unknown binding sites; 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. Overall, the two examples suggested that the seemingly low performance values of *bindEmbed21DL* were, at least partially, rooted in the missing experimental annotations (residues not observed to bind treated as non-binding). We had selected the two of 84 by a simple algorithm: Pick those with an abundance of reliable binding predictions for which alternative experimental information was

available. In doing this, we found that most seemingly incorrect binding predictions appeared correct. In fact, for those investigated in detail, precision was closer to 100% than to 80% (precision at $p \geq 0.95$, Fig. 4). Of the 84 proteins with seemingly incorrect, highly reliable binding 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 non-observed reliable ($p \geq 0.95$) metal predictions (1%) and one of the 80 proteins with non-observed reliable ($p \geq 0.95$) small molecule predictions were confirmed by low resolution structures. While those examples demonstrated qualitatively that our assessment clearly under-estimated performance, they did not suffice to adjust performance measures.

Final method *bindEmbed21* combined HBI and ML. 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 validation set yielded very good results for low E-value thresholds, but at those thresholds, hits were only found for few proteins (Supplementary Fig. S7). For instance, for E-values $\leq 10^{-50}$, HBI achieved F1 = 56 ± 4% (Supplementary Fig. S7, leftmost dark red bar), but only 198 of the 1014 proteins found a hit, i.e., another protein with experimental annotations. When only using HBI to predict for all proteins, a random decision would have to be made for proteins without a hit. Thereby, performance dropped substantially (Supplementary Fig. S7, leftmost light red bar). Hence, HBI outperformed our ML method *bindEmbed21DL* only for a small subset of proteins. We combined the best of both (*bindEmbed21DL* and HBI) applying a simple protocol: Predict binding residues through HBI if an experimentally annotated sequence-similar protein is available, otherwise use ML. This combination was best (highest recall) at an E-value threshold of 10^{-3} (Supplementary Fig. S7A, blue bar). The optimum was not sharp; instead, numbers remained almost constant over at least six orders of magnitude in E-value. While this implied stability (other choices would have given similar results), one reason for the lack of a sharp optimum was the small data set combined with the fact that only about 5% of all residues bind. Therefore, increasing the E-value tenfold brought in much fewer binding residues than proteins (Supplementary Fig. S8).

Combining ML and HBI improved performance on *TestSet300* by five percentage points for F1 (F1 = 48 ± 3%; Fig. 6D, Supplementary Table S8). HBI also improved performance for each ligand class (Fig. 6A–C, Supplementary Table S8) except for the precision in predicting nucleic acid binding (Fig. 6B, Supplementary Table S8). ML performance was somehow limited by overprediction, especially for metal ions and small molecules (low CovNoBind; Supplementary Tables S4, S5), i.e., many proteins were predicted to bind to those ligand classes without matching annotations. Combining bindEmbed21DL with HBI slightly reduced overprediction (higher CovNoBind, lower CovOneBind, Supplementary Table S9) for all three ligand classes. Since the effect was largest for nucleic acids, this could explain the drop in precision of the final combined method *bindEmbed21* compared to the ML-only component *bindEmbed21DL*, because precision was set to zero for proteins annotated but not predicted to bind to a ligand class.

Prediction for complete human proteome discovered unknown candidate binding residues. Of the 20,386 sequences with 11,362,967 residues currently constituting the human proteome in SwissProt[36], only 3121 (15%) had any structure with binding annotations in BioLiP (Fig. 7, Supplementary Table S10). Using our protocol for HBI (transfer binding annotations of local alignment if E-value $\leq 10^{-3}$) transferred binding residues for another 7199 proteins pushing the annotations from *BioLiP + HBI* to 51% (Fig. 7, Supplementary Table S10; 53% for E-value cutoff of 1), i.e., for about half of all human proteins, no ligand is known. As most proteins likely bind some ligand to function correctly, many ligands remain obscure. In fact, this calculation substantially under-estimated the extent of missing annotations by considering a single binding annotation as "protein covered" although 80% of the proteins have several domains[37,38], i.e., there are not only missing annotations in the 49% of the proteins without annotation but also in other domains (or even other regions) of the proteins covered by *BioLiP + HBI*. 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 (8510), no binding information had been available without our prediction (Fig. 7, Supplementary Table S10) and 21% of those 8510 (1751) were predicted reliably (probability ≥ 0.95 corresponding to > 73% precision, Supplementary Table S7). In addition, for 21% of the proteins with experimental or HBI-inferred annotations, bindEmbed21DL provided highly reliable binding predictions previously unknown.

Comparing the probability distributions of residues predicted to bind between proteins with and without annotations, we observed a clear difference between those (Supplementary Fig. S9). Neither abundance in disordered regions nor abundance in membrane proteins nor different length distributions explained any aspect of the difference (Supplementary Fig. S10). On the other hand, the large overlap between the distributions (Supplementary Fig. S9) suggested that, while some of the newly predicted binding residues potentially stem from prediction mistakes, especially highly reliably predicted residues could point towards new binding residues.

One important result from the human proteome prediction was the relative contribution of the three ligand classes: Of all human residues, 1.2%, 2.0%, 3.1% were predicted to bind metal, nucleic acids, and small molecules, respectively (Supplementary Tables S10, S11). Thus, about 20% of the binding residues were predicted to bind metal, 30% nucleic acid, and about 50% small molecules. Overall we assume that the mistakes made in all binding predictions were unbiased, i.e., the 20:30:50 (metal:nucleic:small) are likely good estimates for what a complete experimental coverage of all human proteins would reveal. This finding suggested that our *TestSet300* provided a much more representative mixture of these classes than *TestSetNew46* and a slightly more representative mixture than *DevSet1014* (Supplementary Table S11).

As seen for the example of the human proteome, binding annotations are far from complete and cannot be inferred using HBI for most proteins leading to two major observations: (1) fast and generally applicable de novo

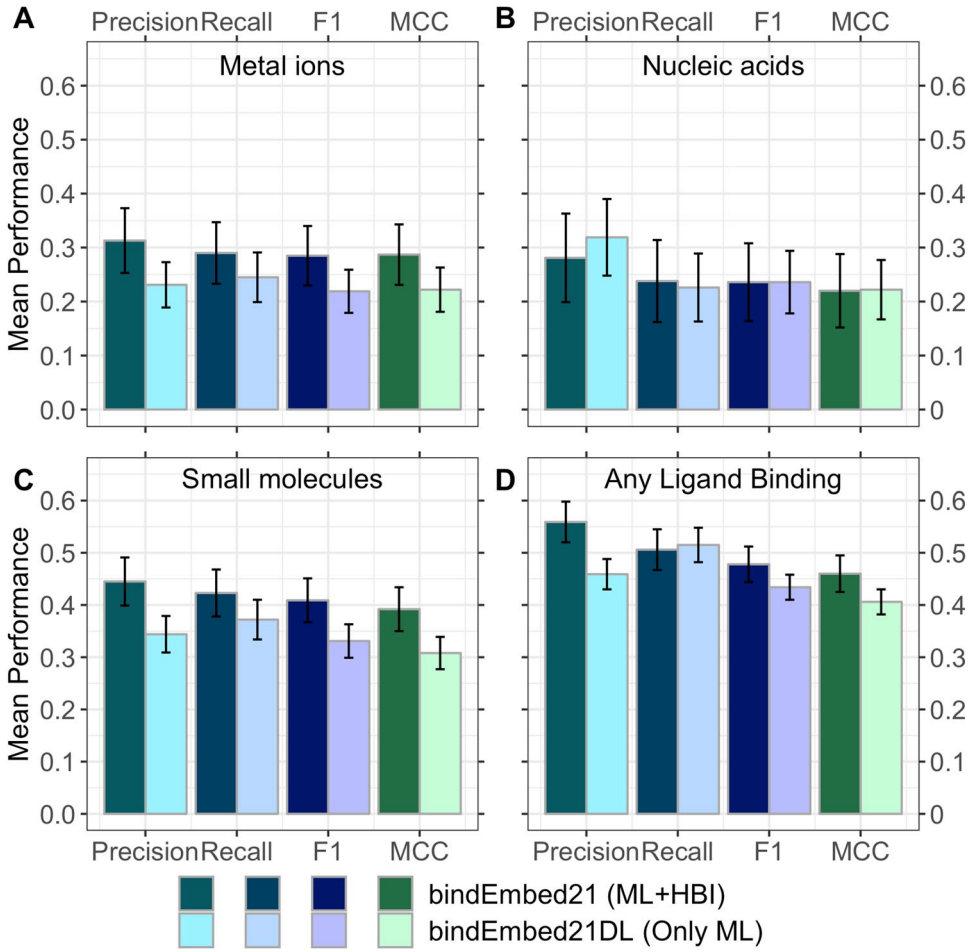

Figure 6. Best performance by combining ML and HBI. We combined homology-based inference (HBI) and Machine Learning (ML) by transferring annotations between homologs (E-value < 10^{-3}) if available and running de novo ML predictions using bindEmbed21DL, otherwise. This combination improved performance 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 methods such as *bindEmbed21DL* are an important tool for the identification of new potential binding residues and ligands that could guide future experiments, and (2) our performance estimates are most likely too conservative due to missing annotations. In fact, while 48,700 residues were annotated as binding in structures with a resolution ≤ 2.5 Å, 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%, Supplementary Table S7), 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 min using one single Xeon machine with 400 GB RAM, 20 cores and a Quadro RTX 8000 GPU with 48 GB vRAM (40 min for the generation of the embeddings, 40 min for the predictions), i.e., generating binding residue predictions for one protein sequence took around 0.2 s allowing fast predictions for large sets of proteins.

Availability. All data, 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[39]. In addition, *bindEmbed21* and its components *bindEmbed21DL* and *bindEmbed21HBI* are publicly available through bio_embeddings. Users can apply the combined method or run its components independently. Therefore, binding residue predictions can be generated fully without the need of any alignment method.

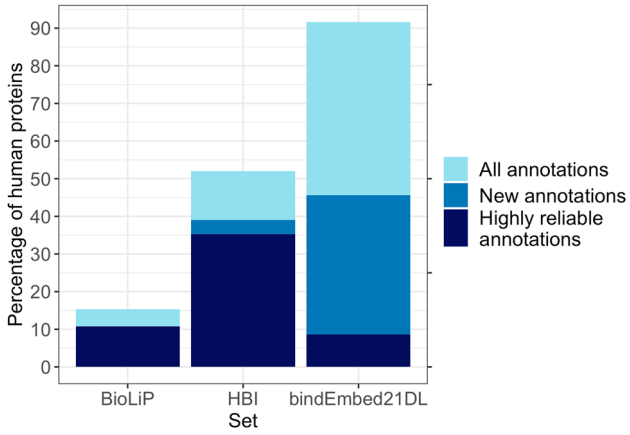

Figure 7. Binding predictions for complete human proteome. BioLiP: experimental annotations, HBI: homology-based inference, bindEmbed21DL: predictions. Data: human proteome from Swiss-Prot[36] with 20,386 proteins. For 3121 human proteins (15%), any binding annotation was experimentally known; 2211 of those were reliable (resolution $\leq$ 2.5 Å; darkblue bar for BioLiP). For 10,526 proteins (52%), binding annotations could be inferred using HBI at EVAL $\leq$ 1 (light blue). Of those, 7973 proteins were not previously annotated (blue bar); for 7199 proteins without binding annotations, binding annotations could be inferred at EVAL $\leq 10^{-3}$ following the protocol of bindEmbed21HBI (dark blue). Therefore, BioLiP + HBI allowed annotating some binding in 52% of the human proteome. bindEmbed21DL predicted binding residues for 18,663 proteins (92%) (light blue); no annotations were previously known (neither through BioLiP nor HBI) for 8510 (blue). Highly reliable predictions (probability $\geq$ 0.95) were possible for 1751 proteins without previous binding annotations (dark blue).

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 input from 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 1100 non-redundant proteins from any organism. Given the data sparsity, it is likely that many binding residues remain unknown even in the subset of 1100 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, Eq. 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 MSA-based method ProNA2020[19] specialized to predict DNA- and RNA-binding and the zinc-binding prediction methods PredZinc[20] and ZincBindPredict[29] (Fig. 3). Prediction strength correlated with performance (Fig. 4): For the one fourth of all binding residues predicted with a probability $\geq$ 0.95, 73% corresponded to experimentally known binding annotations available today (Supplementary Table S7). Detailed analysis of very reliable predictions not matching known experimental annotations revealed that *bind*Embed21DL correctly predicted binding residues missing in the structures 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 *bindEmbed21DL* can help in identifying new potential binding sites (Fig. 7, Supplementary Table S10). Overall, about 6% of all residues in the human proteome were predicted to bind any of the three ligand classes covered (metals 1.2%, nucleotides 2.0%, small molecules 3.1%). The proteome analysis also suggested our performance estimates as too conservative: For the two carefully investigated case studies, many reliably predicted ligands not annotated tended to be correct. 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 and methods

Data sets. Protein sequences with binding annotations were extracted from BioLiP[9]. BioLiP provides binding annotations for residues based on structural information from the Protein Data Bank (PDB)[31], i.e., proteins for which several PDB structures with different identifiers exist may have multiple binding annotations. We extracted and combined (union) all binding information from BioLiP for all chains of PDB structures match-

ing a given sequence, which have been determined through X-ray crystallography[40] with a resolution of ≤ 2.5 Å (≤ 0.25 nm). 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. 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[36]. This set was clustered to remove redundancy using UniqueProt[41] with an HVAL < 0 (corresponding to no pair of proteins in the data set having over 20% pairwise sequence identity over 250 aligned residues[42,43]). We provide more details on the data in Supplementary Table S12 and on the redundancy reduction in Supplementary Sect. 2.1 of the Supporting Online Material (SOM). The final set of 1314 proteins was split into a development set with 1014 proteins (called *DevSet1014* with 13,999 binding residues, 156,684 non-binding residues; Supplementary Table S12) used for optimizing model weights and hyperparameters (after another random split into training and validation), and test set with 300 proteins (*TestSet300* with 5869 binding residues, 56,820 non-binding residues; Supplementary Table S12). To allow maximum overlap to the development set of bindPredictML17[5], we first extracted the 225 proteins from the 1314 proteins which were also part of the data set of bindPredictML17. Since bindPredictML17 was only trained on enzymes and DNA-binding proteins, this set was highly biased towards nucleic acid and small molecule binding (59 proteins binding to nucleic acids, 176 to small molecules, and 95 to metal ions). The additional 75 proteins were added to slightly adjust for this imbalance. However, a full adjustment was not possible without decreasing the size of the training set too much.

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 1592 proteins. However, upon redundancy reduction with HVAL < 0, this set melted down to 46 proteins with 575 binding and 5652 non-binding residues (*TestSetNew46*; Supplementary Table S12). These numbers imply two interesting findings: Firstly, about 17 experiments with binding data have been published every week over the last 91 weeks. Secondly, less than one experiment per week provided completely new insights into binding of residues not previously characterized in similar proteins (3% of all experiments). 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 solely on unlabeled protein sequences from BFD (Big Fantastic Database; 2.5 billion sequences including meta-genomic sequences)[44] and UniRef50[36]. ProtT5 has been built in analogy to the NLP (Natural Language Processing) T5[45] ultimately learning some of the constraints of protein sequence. 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 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 ProtT5 (Supplementary Fig. S13, Step 1) without fine-tuning it on the task of binding residue prediction (i.e., the gradient of the binding prediction was not backpropagated to ProtT5).

AI/Deep learning architecture. For *bindEmbed21DL*, we realized the supervised learning through a relatively shallow (few free parameters) two-layer Convolutional Neural Network (CNN; Supplementary Fig. S13, Step 2). The CNN was implemented in PyTorch[46] and trained with the following settings: Adamax optimizer, learning rate = 0.01, early stopping, and a batch size of 406 (resulting in two batches). ProtT5 embeddings (from the last layer of ProtT5, 1024-dimensional vector 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 × 1024 to an output of L × 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 × 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)[47] and a dropout layer[48], with a dropout rate of 70%. The two-layer CNN proved to be the best-performing architecture among a variety of architectures including CNNs with more layers, feedforward neural networks, and combinations of both. Feature channels, learning rate, kernel size, and dropout rate were optimized using an exhaustive grid search.

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 (Eq. 3) and MCC (Eq. 4). Higher weights in the loss function increased recall (Eq. 1), lower weights increased precision (Eq. 2). The final weights were 8.9, 7.7, and 4.4 for metal ions, nucleic acids, and small molecules, respectively.

Homology-based inference. Homology-based inference (HBI) generally 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 determined empirically. Most successful in silico predictions of function are predominantly based on HBI[4,8,49–54].

In our case, we aligned query proteins without binding annotations with MMseqs2[55], creating evolutionary profiles from the resulting multiple sequence alignments (MSAs) for each protein (family) (two MMseqs2

iterations, at E-value $\leq 10^{-3}$) against a 80% non-redundant database combining UniProt[36] and PDB[31] adapting a standard protocol based on PSI-BLAST[56] which was implemented for other methods before[19,24,50]. The resulting profiles were then aligned at E-value $\leq 10^{-3}$ against a set of proteins with experimentally known binding annotations (see SOM 2.3 for explicit MMseqs2 commands). To save resources, we clustered the set of proteins with known annotations at 95% pairwise sequence identity (PIDE; 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 PIDE to the query was chosen. If this hit contained any binding annotations in the aligned region, 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 incorrectly predicted as non-binding. Based on this classification for each residue, we evaluated performance using recall (or sensitivity, Eq. 1), precision (Eq. 2), F1 score (Eq. 3), and Matthews Correlation Coefficient (MCC, Eq. 4).

$$Recall = \frac{TP}{TP + FN}, \tag{1}$$

$$Precision = \frac{TP}{TP + FP}, \tag{2}$$

$$F1 = 2 \bullet \frac{Recall \cdot Precision}{Recall + Precision}, \tag{3}$$

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

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

$$Negative\ Recall = \frac{TN}{TN + FP}, \tag{5}$$

$$Negative\ Precision = \frac{TN}{TN + FF}, \tag{6}$$

$$Negative\ F1 = 2 \cdot \frac{Negative\ Recall \cdot Negative\ Precision}{Negative\ Recall + Negative\ Precision}. \tag{7}$$

The measure *CovOneBind* (Eq. 8) indicated the fraction of proteins for which at least one residue was predicted as binding. Accordingly, the inverse of this, *CovNoBind* (Eq. 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 ligand classes, i.e., the fraction of proteins for which ligand *l* was neither observed nor predicted (Eq. 9).

$$CovOneBind = \frac{Number\ of\ proteins\ with\ one\ binding\ residue\ predicted}{Number\ of\ proteins\ with\ binding\ annotations}, \tag{8}$$

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

CovOneBind is an interesting measure to consider for experimentalists who submit only one sequence to a server and want to gauge how likely absence of prediction for that protein implies absence of binding. It does not give a clear indication of the performance of the method for a specific protein but attempts to capture how broadly applicable a method is. If a method only predicts binding residues for a small subset of proteins with high precision, it could still be considered inferior to a method predicting binding residues less precisely but for more proteins because those predictions can still provide valuable information.

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. While the performance values are not following a normal distribution, the sample size was sufficiently large to assume that a normal distribution can be applied in this case. For security we also tested bootstrapped CIs yielding the same results (SOM 2.4).

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) (Eq. 10).

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

Comparison to other methods. We compared our new method to the following four. We could not compare with other methods for different reasons (Supplementary Table S14).

bindPredictML17[5] predicts binding residues from enzymes (trained on the PDB) and DNA-binding residues from PDIdb[57]. The method first builds MSAs and uses those to compute evolutionary couplings[23] and effect predictions[21,22]. Those two main features are then used as input to the machine learning method.

ProNA2020[19] 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. The second per-residue level predicts which residue binds to any (or all) of the three ligand classes. ProNA2020 combines HBI 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[24] as input.

PredZinc[20] predicts binding to zinc ions using a combination of HBI 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.

ZincBindPredict[29] is based on different Random Forest models to predict one particular zinc-binding site family. The models were trained on feature vectors encoding inter-residue distance, hydrophobicity, and number of charges around a residue.

Data availability

The source code, data sets, and the trained model are publicly available as a GitHub repository (https://github.com/Rostlab/bindPredict). ProtT5 embeddings can be generated using the bio_embeddings pipeline. Predictions using bindEmbed21 can be easily generated using its integration into bio_embeddings.

Received: 6 September 2021; Accepted: 2 December 2021
Published online: 13 December 2021

References

1. Whisstock, J. C. & Lesk, A. M. Prediction of protein function from protein sequence and structure. *Q. Rev. Biophys.* **36**, 307–340. https://doi.org/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. https://doi.org/10.1002/prot.23174 (2011).
4. Radivojac, P. *et al.* A large-scale evaluation of computational protein function prediction. *Nat. Methods* **10**, 221–227. https://doi.org/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. https://doi.org/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 Bioinform.* **21**, 452. https://doi.org/10.1186/s12859-020-03759-0 (2020).
7. Mahlich, Y. *et al.* Common sequence variants affect molecular function more than rare variants?. *Sci. Rep.* **7**, 1608. https://doi.org/10.1038/s41598-017-01054-2 (2017).
8. Hamp, T. *et al.* Homology-based inference sets the bar high for protein function prediction. *BMC Bioinform.* **14**(Suppl 3), S7. https://doi.org/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. https://doi.org/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. https://doi.org/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. https://doi.org/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. https://doi.org/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. https://doi.org/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. https://doi.org/10.1093/bioinformatics/btaa110 (2020).
15. Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* https://doi.org/10.1038/s41586-021-03819-2 (2021).
16. Weißenow, K., Heinzinger, M. & Rost, B. Protein language model embeddings for fast, accurate, alignment-free protein structure prediction. *bioRxiv.* https://doi.org/10.1101/2021.07.31.454572 (2021).
17. Cui, Y., Dong, Q., Hong, D. & Wang, X. Predicting protein-ligand binding residues with deep convolutional neural networks. *BMC Bioinform.* **20**, 93. https://doi.org/10.1186/s12859-019-2672-1 (2019).
18. 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. https://doi.org/10.1093/bioinformatics/btw396 (2016).
19. 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. https://doi.org/10.1016/j.jmb.2020.02.026 (2020).
20. Shu, N., Zhou, T. & Hovmoller, S. Prediction of zinc-binding sites in proteins from sequence. *Bioinformatics* **24**, 775–782. https://doi.org/10.1093/bioinformatics/btm618 (2008).

21. Hopf, T. A. *et al.* Mutation effects predicted from sequence co-variation. *Nat. Biotechnol.* **35**, 128–135. https://doi.org/10.1038/nbt.3769 (2017).
22. Hecht, M., Bromberg, Y. & Rost, B. Better prediction of functional effects for sequence variants. *BMC Genom.* **16**(Suppl 8), S1. https://doi.org/10.1186/1471-2164-16-S8-S1 (2015).
23. Marks, D. S., Hopf, T. A. & Sander, C. Protein structure prediction from sequence variation. *Nat. Biotechnol.* **30**, 1072–1080. https://doi.org/10.1038/nbt.2419 (2012).
24. Bernhofer, M. *et al.* PredictProtein—Predicting protein structure and function for 29 years. *Nucleic Acids Res.* https://doi.org/10.1093/nar/gkab354 (2021).
25. Ofran, Y., Mysore, V. & Rost, B. Prediction of DNA-binding residues from sequence. *Bioinformatics* **23**, i347-353 (2007).
26. Ofran, Y. & Rost, B. Predicted protein–protein interaction sites from local sequence information. *FEBS Lett.* **544**, 236–239 (2003).
27. Peng, Z. & Kurgan, L. High-throughput prediction of RNA, DNA and protein binding regions mediated by intrinsic disorder. *Nucleic Acids Res.* **43**, e121. https://doi.org/10.1093/nar/gkv585 (2015).
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.* https://doi.org/10.1109/TPAMI.2021.3095381 (2021).
29. Ireland, S. M. & Martin, A. C. R. Zincbindpredict-prediction of zinc binding sites in proteins. *Molecules* https://doi.org/10.3390/molecules26040966 (2021).
30. Dunker, A. K. *et al.* What's in a name? Why these proteins are intrinsically disordered. *Intrinsically Disord. Proteins* **1**, e24157 (2013).
31. 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 Res.* **47**, D464–D474. https://doi.org/10.1093/nar/gky1004 (2019).
32. 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. https://doi.org/10.1006/jmbi.2000.4104 (2000).
33. Mattiroli, F. *et al.* Structure of histone-based chromatin in Archaea. *Science* **357**, 609–612. https://doi.org/10.1126/science.aaj1849 (2017).
34. 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. https://doi.org/10.1093/nar/gkz285 (2019).
35. Reiter, N. J. *et al.* Structure of a bacterial ribonuclease P holoenzyme in complex with tRNA. *Nature* **468**, 784–789. https://doi.org/10.1038/nature09516 (2010).
36. The UniProt Consortium. UniProt: The universal protein knowledgebase in 2021. *Nucleic Acids Res.* **49**, D480–D489. https://doi.org/10.1093/nar/gkaa1100 (2021).
37. Liu, J. & Rost, B. Domains, motifs, and clusters in the protein universe. *Curr. Opin. Chem. Biol.* **7**, 5–11 (2003).
38. Liu, J. & Rost, B. CHOP proteins into structural domain-like fragments. *Proteins Struct. Funct. Bioinform.* **55**, 678–688 (2004).
39. Dallago, C. *et al.* Learned embeddings from deep learning to visualize and predict protein sets. *Curr. Protoc.* **1**, e113. https://doi.org/10.1002/cpz1.113 (2021).
40. Smyth, M. S. & Martin, J. H. X ray crystallography. *Mol. Pathol.* **53**, 8–14. https://doi.org/10.1136/mp.53.1.8 (2000).
41. Mika, S. & Rost, B. UniqueProt: Creating representative protein sequence sets. *Nucleic Acids Res.* **31**, 3789–3791. https://doi.org/10.1093/nar/gkg620 (2003).
42. Sander, C. & Schneider, R. Database of homology-derived structures and the structural meaning of sequence alignment. *Proteins Struct. Funct. Genet.* **9**, 56–68 (1991).
43. Rost, B. Twilight zone of protein sequence alignments. *Protein Eng.* **12**, 85–94 (1999).
44. Steinegger, M., Mirdita, M. & Söding, J. Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nat. Methods* **16**, 603–606. https://doi.org/10.1038/s41592-019-0437-4 (2019).
45. Raffel, C. *et al.* Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **21**, 1–67 (2020).
46. Paszke, A. *et al.* PyTorch: An imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **32**, 8026–8037 (2019).
47. Clevert, D.-A., Unterthiner, T. & Hochreiter, S. Fast and accurate deep network learning by exponential linear units (ELUs). arXiv preprint arXiv::1511.07289 (2015).
48. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).
49. Friedberg, I. & Radivojac, P. Community-wide evaluation of computational function prediction. *Methods Mol. Biol.* **1446**, 133–146. https://doi.org/10.1007/978-1-4939-3743-1_10 (2017).
50. Goldberg, T. *et al.* LocTree3 prediction of localization. *Nucleic Acids Res.* **42**, W350-355. https://doi.org/10.1093/nar/gku396 (2014).
51. Jiang, Y. *et al.* An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biol.* **17**, 184. https://doi.org/10.1186/s13059-016-1037-6 (2016).
52. Ofran, Y., Punta, M., Schneider, R. & Rost, B. Beyond annotation transfer by homology: Novel protein-function prediction methods to assist drug discovery. *Drug Discov. Today* **10**, 1475–1482 (2005).
53. 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. https://doi.org/10.1186/s13059-019-1835-8 (2019).
54. Littmann, M., Heinzinger, M., Dallago, C., Olenyi, T. & Rost, B. Embeddings from deep learning transfer GO annotations beyond homology. *Sci. Rep.* **11**, 1160. https://doi.org/10.1038/s41598-020-80786-0 (2021).
55. Steinegger, M. & Söding, J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.* **35**, 1026–1028. https://doi.org/10.1038/nbt.3988 (2017).
56. Altschul, S. F. *et al.* Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* **25**, 3389–3402. https://doi.org/10.1093/nar/25.17.3389 (1997).
57. Norambuena, T. & Melo, F. The Protein–DNA interface database. *BMC Bioinform.* **11**, 262. https://doi.org/10.1186/1471-2105-11-262 (2010).
58. Kuang, R. *et al.* Profile-based string kernels for remote homology detection and motif extraction. *J. Bioinform. Comput. Biol.* **3**, 527–550. https://doi.org/10.1142/s021972000500120x (2005).
59. Hamp, T., Goldberg, T. & Rost, B. Accelerating the original profile kernel. *PLoS One* **8**, e68459. https://doi.org/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. https://doi.org/10.1371/journal.pone.0141287 (2015).

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).

Author contributions

M.L. prepared the data set, implemented and evaluated the final method bindEmbed21 and its two components bindEmbed21DL and bindEmbed21HBI, and performed the major part of manuscript writing and figure generation. M.H. provided ProtT5 embeddings. M.H. and K.W. helped with the original setup of the Deep Learning architecture forming the basis of bindEmbed21DL. C.D. facilitated the integration of bindEmbed21 into bio_embeddings. B.R. supervised and guided the work over the entire time and proofread the manuscript. All authors read and approved the final manuscript.

Funding

Open Access funding enabled and organized by Projekt DEAL.

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/s41598-021-03431-4.

Correspondence and requests for materials should be addressed to M.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.