ARTICLES
https://doi.org/10.1038/s41592-022-01490-7

nature methods

ScanNet: an interpretable geometric deep learning model for structure-based protein binding site prediction

Jérôme Tubiana[1]✉, Dina Schneidman-Duhovny[2]✉ and Haim J. Wolfson[1]✉

Predicting the functional sites of a protein from its structure, such as the binding sites of small molecules, other proteins or antibodies, sheds light on its function in vivo. Currently, two classes of methods prevail: machine learning models built on top of handcrafted features and comparative modeling. They are, respectively, limited by the expressivity of the handcrafted features and the availability of similar proteins. Here, we introduce ScanNet, an end-to-end, interpretable geometric deep learning model that learns features directly from 3D structures. ScanNet builds representations of atoms and amino acids based on the spatio-chemical arrangement of their neighbors. We train ScanNet for detecting protein–protein and protein–antibody binding sites, demonstrate its accuracy—including for unseen protein folds—and interpret the filters learned. Finally, we predict epitopes of the SARS-CoV-2 spike protein, validating known antigenic regions and predicting previously uncharacterized ones. Overall, ScanNet is a versatile, powerful and interpretable model suitable for functional site prediction tasks. A webserver for ScanNet is available from http://bioinfo3d.cs.tau.ac.il/ScanNet/.

Despite recent progresses in experimental[1] and AI-based[2,3] protein structure determination, there remains a gap between structure and function[4]. The most accurate functional site prediction method is comparative modeling[5–13]: given a query protein, similar proteins with known functional sites are searched for and their sites are mapped onto the query structure. Comparative modeling has several shortcomings. First and foremost, its coverage is limited, as the pool of experimentally characterized protein folds or structural motifs is small. Second, functional sites are variably preserved throughout evolution. On the one hand, the B cell epitopes (BCEs) of viral proteins frequently undergo antigenic drift, that is, the abolition of recognition by antibodies after only one or few mutations. On the other hand, some protein–protein interactions (PPIs) are mainly driven by few 'hotspot' residues: mutations and/or conformational changes of the other interface residues preserve the interaction. Put differently, the invariances in both sequence and conformation spaces of such function-determining structural motifs are in general motif-dependent and therefore unknown. This hampers our ability to both define and recognize such motifs using conventional comparative approaches.

An alternative to comparative modeling is feature-based machine learning[12–18]. For each amino acid of a query protein, various features of geometrical (for example, secondary structure, solvent accessibility, molecular surface curvature), physico-chemical (for example, hydrophobicity, polarity, electrostatic potential) and evolutionary (for example, conservation, position–weight matrices, coevolution) nature are calculated. Then, the target property is predicted using a machine learning model for tabular data such as random forest or gradient boosting. Reasoning on mathematically defined features offers three advantages: (1) ability to generalize to proteins with no similarity to any of the train set proteins, (2) high sequence sensitivity, that is, ability to output distinct predictions for highly similar protein sequences and (3) fast inference speed. Machine learning models are, however, limited by the expressiveness of the features used, as these cannot capture the spatio-chemical arrangements of atoms or amino acids characterizing function-bearing motifs. Examples of such function-bearing motifs include Zinc fingers that are signatures of DNA or RNA binding sites[19], or PPI hotspot 'O-rings'[20]: namely, exposed hydrophobic/aromatic amino acids surrounded by polar/charged ones. Despite over 50 years of experimental structural determination, new function-determining motifs are still being discovered[21].

End-to-end differentiable models, that is, deep learning, can potentially overcome the limitations of both approaches. Indeed, deep learning models can learn the data features and their invariances directly by backpropagation, and generalize well despite a large number of parameters. Adapting the deep learning approach to protein structures requires defining an appropriate representation for proteins. Proteins can indeed be represented in multiple, complementary ways, for example as sequences[22,23], residue graphs[24–27], atomic density maps[28–34], atomic point clouds[35] or molecular surfaces[36,37], each capturing different functionally relevant features. Voxelated atomic density maps can be readily processed using classical 3D convolutional neural networks, but the approach is computationally intensive and the predictions are not invariant on rotation of the input structure. Point clouds, graphs and surfaces can be analyzed via geometric deep learning[38,39], that is, end-to-end differentiable models tailored for data with no natural grid-like topology or shared global coordinate system. Graphs can be derived from 3D structures by taking residues as nodes and the distances and angles between them as edges and processed using graph neural networks (GNN) such as message passing neural networks[40] or graph attention networks[41]. By design, GNNs are invariant on Euclidean transformation and expressive, but can be challenging to regularize and interpret. In particular, it is unclear whether—and if yes, which—structural motifs are captured by GNNs. Here, we introduce ScanNet (spatio-chemical arrangement of neighbors neural network), a new geometric deep learning architecture tailored for protein structures.

[1]Blavatnik School of Computer Science, Tel Aviv University, Tel Aviv, Israel. [2]School of Computer Science and Engineering, The Hebrew University of Jerusalem, Jerusalem, Israel. ✉e-mail: jertubiana@gmail.com; dina.schneidman@mail.huji.ac.il; wolfson@tau.ac.il

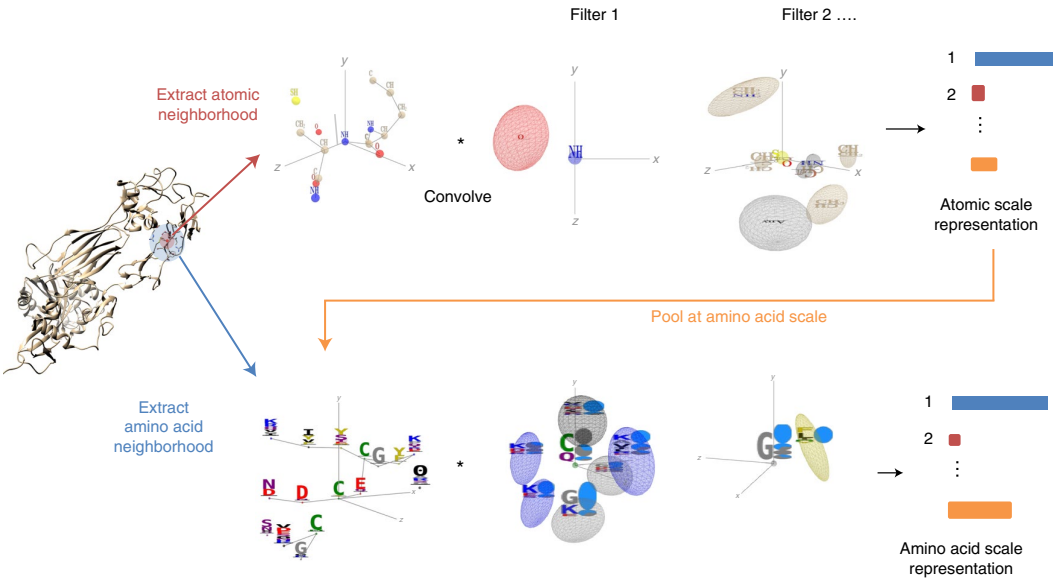


Fig. 1 | Overview of the ScanNet architecture. ScanNet inputs are the primary sequence, tertiary structure and, optionally, position–weight matrix computed from a MSA of evolutionarily related proteins. First, for each atom, neighboring atoms are extracted from the structure and positioned in a local coordinate frame (top left). The resulting point cloud is passed through a set of trainable, linear filters detecting specific spatio-chemical arrangements (top middle), yielding an atomic-scale representation (top right). After aggregation of the atomic representation at the amino acid level and concatenation with amino acid attributes, the process is reiterated with amino acids to obtain a representation of an amino acid (bottom). The latter is projected and locally averaged for residue-wise classification.

ScanNet builds representations of atoms and amino acids based on the spatio-chemical arrangement of their neighbors and exploits them to predict labels for each amino acid. By construction, ScanNet is end-to-end differentiable with minimal structure preprocessing, yielding fast training and inference. ScanNet predictions are local, invariant on Euclidean transformations and integrate information from multiple scales (atom, amino acid) and modalities (structure, multiple sequence alignment (MSA)) in a synergistic fashion. Its corresponding parametric function is expressive, meaning that it can efficiently approximate known handcrafted features. Through appropriate parameterization and regularization, the filters learned by ScanNet can be readily visualized and interpreted. We showcase the capabilities of ScanNet on two related tasks: prediction of protein–protein binding sites (PPBS) and BCE (that is, antibody binding sites). ScanNet outperforms baseline methods based on machine learning, structural homology and surface-based geometric deep learning. We further visualize and interpret the representations learned by the network. We find that they encompass known handcrafted features and find filters detecting simple, generic structural motifs, such as hydrogen bonds, as well as filters recognizing complex, task-specific motifs, such as O-rings and transmembrane helical domains. Applied to the SARS-CoV-2 spike protein, ScanNet predictions validate known antigenic regions and predict a previously uncharacterized one.

Results

Spatio-chemical arrangement of neighbors network (ScanNet). ScanNet takes as input a protein structure file and, optionally, a position–weight matrix derived from a MSA and outputs a residue-wise label probability. Its four main stages, shown in Fig. 1 and detailed in the Methods (Extended Data Figs. 1 and 2), are: atomic neighborhood embedding, atom to amino acid pooling, amino acid neighborhood embedding and neighborhood attention.

ScanNet first builds, for each heavy atom, a local coordinate frame centered on its position and oriented according to its covalent bonds. Next, it identifies its closest neighboring atoms. The resulting neighborhood, formally a point cloud with coordinates and attributes (atom group type) is passed through a set of spatio-chemical linear filters to yield an atom-wise representation. Each filter outputs a matching score between its (trainable) spatio-chemical pattern and the neighborhood. The patterns, which are parameterized using Gaussian kernels and sparse bilinear products, are localized in both physical and attribute space. Localization facilitates interpretation and is biologically motivated since motif functionality is often born by a few key atomic groups/amino acids in a specific arrangement, whereas other neighbors are irrelevant and interchangeable. Trainable, localized spatio-chemical patterns generalize to proteins the well-known concept of pharmacophores for small molecules.

Toward calculation of amino acid-wise output, the atom-wise representation is pooled at the amino acid scale and concatenated with embedded amino acid-level information (either amino acid type or position–weight matrix). The constituting atoms of an amino acid have various types and may play different functional roles. In particular, some handcrafted features such as accessible surface area average information over all the atoms, whereas others such as secondary structure consider only subsets (the backbone atoms). Therefore, a trainable, multi-headed attention pooling operation capable of learning which atoms are relevant for each feature is used rather than a conventional symmetric pooling operation such as average or maximum.

The neighborhood embedding procedure is then repeated at the amino acid scale: a local coordinate frame is constructed for each amino acid from its C_α atom, sidechain orientation and local backbone orientation and its nearest neighbors are identified. The resulting neighborhood with learned attributes is passed through a set of trainable filters to yield an amino acid-wise representation.

Finally, spatially consistent output probabilities are obtained by projecting the amino acid representations to scalar values, smoothing them across a local neighborhood and converting to probabilities with a logistic function. The smoothing scheme integrates two

specifics of protein binding sites. First, PPIs are frequently driven by key hotspot residues that contribute most of the binding energy, whereas other nearby passenger residues have a small contribution to the binding energy[20,42]. Such passenger residues are harder to detect directly as they do not necessarily have the salient features of PPBSs[43]. Second, some amino acid pairs consistently have opposite binding site labels—in particular, consecutive amino acids along the sequence because their sidechains typically point in opposite directions. Altogether, this motivates the introduction of trainable, attention-based weighted averages, with algebraic weights.

ScanNet for prediction of PPBSs. The PPBS of a protein are defined as the residues directly involved in one or more native, high affinity PPIs. Not every surface residue is a PPBS, as (1) binding propensity competes with structural stability and (2) PPIs are highly partner- and conformation-specific. Knowledge of the PPBS of a protein provides insight about its in vivo behavior, particularly when its partners are unknown and can guide docking algorithms. Prediction of PPBS with conventional approaches is challenging as PPBS structural motifs are more diverse, less conserved and more extended than small molecule binding sites. Additionally, only incomplete and noisy labels can be derived from structural data, as (1) most PPIs of a given protein are not structurally characterized, and (2) a substantial fraction (roughly 15%, ref. [44]) of the structurally characterized protein–protein interfaces are not physiological but crystal-induced.

We constructed a nonredundant dataset of 20K representative protein chains with annotated binding sites derived from the Dockground database of protein complexes[45]. The PPBS dataset covers a wide range of complex sizes, types, organism taxonomies, protein lengths (Extended Data Fig. 3a–d) and contains around 5M amino acids, of which 22.7% are PPBS. To address the uneven sampling of the protein space, we introduced sample weights for each chain that are inversely proportional to the number of similar chains found in the dataset (Methods and Extended Data Fig. 3h). To investigate the relationship between homology and generalization error, we divided the validation/test sets into four splits based on the degree of homology with respect to their closest train set example (Fig. 2 and Extended Data Fig. 3g).

We evaluated three models on the PPBS dataset: (1) ScanNet, (2) a machine learning pipeline based on handcrafted features and (3) a structural homology pipeline (see Methods for technical details). For the handcrafted features baseline, we computed for each amino acid various geometric, chemical and evolutionary features, and used xgboost, a state-of-the-art tree-based classification algorithm[46]. For the structural homology pipeline, pairwise local structural alignments between the train set chains and the query chain were first constructed using MultiProt[47]. Then, alignments were weighted and aggregated to produce binding site probabilities for each amino acid. For all three models, the validation set was used for hyperparameters selection and early stopping, and performance is reported on the test set. Training and evaluation of a single model took 1–2 hours for ScanNet (excluding preprocessing time, roughly 10 ms per step using a single Nvidia V100 graphical processing unit (GPU)), a few minutes for the machine learning baseline (excluding feature calculation time, using Intel Xeon Phi processor with 28 cores) and 1 month for the structural homology baseline (Intel Xeon Phi processor with 28 cores). We also evaluated Masif-site[36], a surface-based geometric deep learning model. Since Masif-site was not trained on the same dataset, we only report its global test set performance.

We found that for the full test set, ScanNet achieved an area under the precision-recall curve (AUCPR) of 0.694 (Table 1), accuracy of 87.7% (Supplementary Table 1) and 73.5% precision at 50% recall (Extended Data Fig. 4e,f), the best performance by a substantial margin. The next best model was the structural homology baseline, whereas Masif-site and the handcrafted features model performed similarly. The model ranks differed when considering only subsets (Fig. 2a–d). The structural homology baseline performed best in the high homology setting, but its performance degraded rapidly with the degree of relatedness; when the test protein had no similar fold in the train set, it was the worst algorithm. Conversely, the performance of the handcrafted features baseline increased slowly with the degree of homology, meaning that it could not faithfully recognize previously seen folds. In contrast, ScanNet could both recognize previously seen folds and generalize to unseen ones. Visualizations of ScanNet predictions for representative examples (Fig. 2e,f and Supplementary Figs. 1–4) illustrate that predictions are spatially coherent and that in most cases, the binding sites are correctly identified. Overall, the network performed uniformly well across complex types and sizes, protein lengths and organisms (Extended Data Fig. 5). PPBS identification was slightly harder when no or few homologs were found in the MSA (Extended Data Fig. 5b) and slightly easier for enzymes (Extended Data Fig. 5d). We next identified and visualized train and test examples on which ScanNet performed poorly (Supplementary Fig. 5). We found bona fide false negative (undetected interacting patches) and false positives (predicted interacting patches), although for the latter we could not rule out involvement in another PPI for which no structural data was available. Another source of mistake was confusion between types of binding site: we found at least one instance where the incorrectly predicted PPBS were actually RNA binding sites. However, only a minority of RNA binding domain were confused as protein binding (Supplementary Fig. 6). Finally, confusion between crystal and native interfaces was a substantial source of apparent mistakes. We found several train set examples in which the network refused to learn the train label and instead predicted another binding interface with high confidence (Supplementary Fig. 7). The predicted binding sites matched well the interface found in another biological assembly file. We found a posteriori that the biological assembly files used in the train set were annotated as probably incorrect by QSbio[44]. Overall, this demonstrated the robustness of predictions with respect to noise in training labels.

We next performed ablation experiments to investigate the importance of the network components (Table 1 and Extended Data Fig. 4). ScanNet performance decreased but remained above the other methods when discarding the evolutionary information (by replacing the position–weight matrix by the one-hot encoded sequence) or all the atomic-scale information (by removing the first two modules). Removing the sparse regularization on the spatio-chemical patterns and the early stopping yielded an homology-like performance profile, with better performance in the high homology setting but poorer otherwise. Last, training the model on all chains without redundancy reduction nor using sample weights yielded worse performance, highlighting the importance of sample weights.

Finally, we investigated the impact of conformational changes on binding (that is, induced fit) on ScanNet predictions using the Dockground unbound X-ray and simulated datasets[45]. Overall, predictions based on bound and unbound structures were highly consistent, and accuracy decreased only mildly from bound to unbound (Methods, Extended Data Fig. 6 and Supplementary Table 4).

Visualization and interpretation of the representations. What did ScanNet learn? Does the network reason solely by comparison with training instances or does it learn the underlying chemical principles of binding? How will it behave in out-of-sample settings such as disordered regions? To better understand the learned representations, we visualized the spatio-chemical patterns and low-dimensional projections of the representations at the atomic (Fig. 3) and amino acid (Fig. 4) levels. Recall that each pattern is composed by a set of Gaussian kernels characterized by their location in the local coordinate system and specificity in attribute space. At the atomic scale,

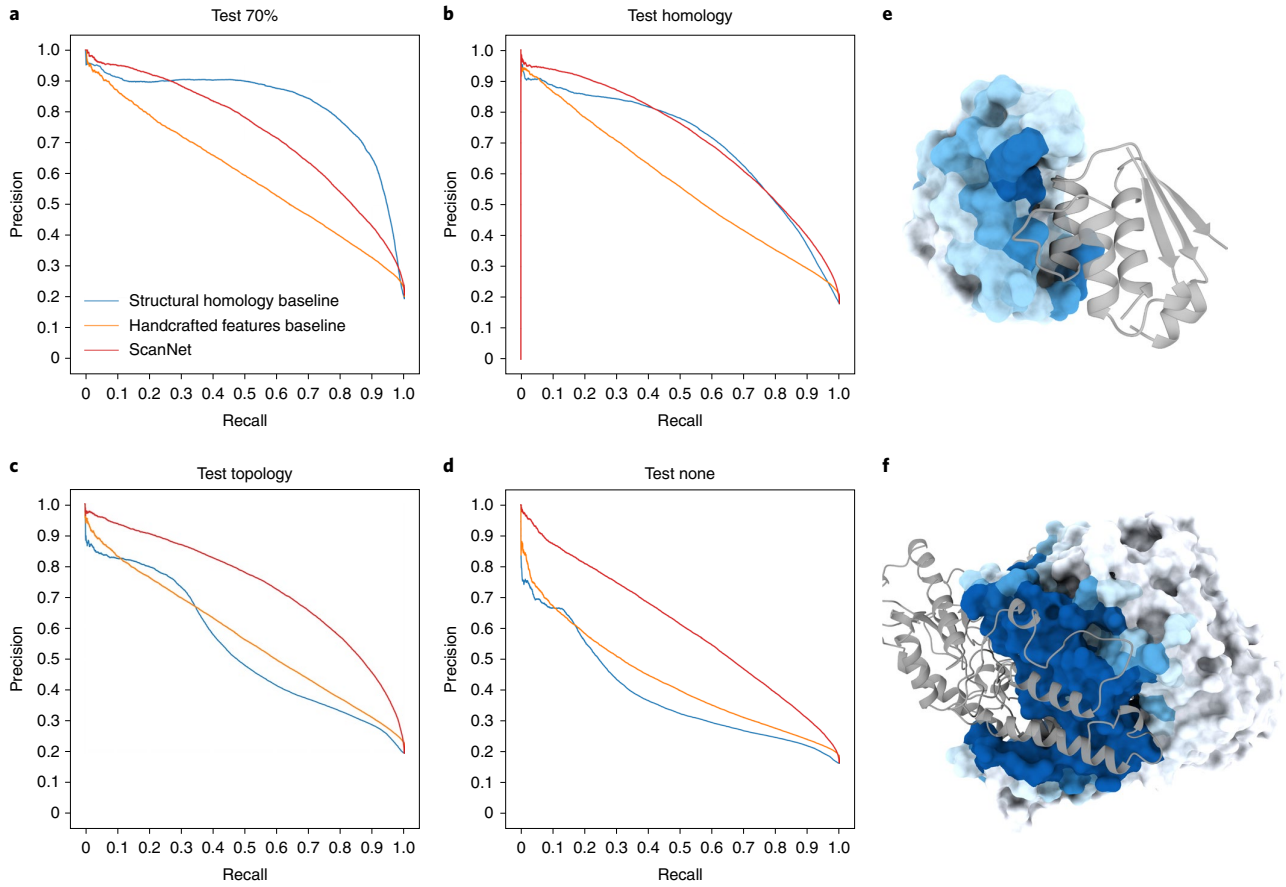


Fig. 2 | Prediction of PPBS with ScanNet. a–**d**, Precision-recall curves of PPBS prediction for ScanNet, structural homology and handcrafted features baseline methods (main text). Train and test sets constructed from the redundant Dockground template database[45]. Proteins of the test set are subdivided into four nonoverlapping groups. **a**, Test 70%, at least 70% sequence identity with at least one train set example. **b**, Test homology, at most 70% sequence identity with any train set example, at least one train set example belonging to same protein superfamily (H level of CATH classification[56]). **c**, Test topology, at least one train set example with similar protein topology (T level of CATH classification[56]), none with similar protein superfamily. **d**, Test none, none of the above. **e**,**f**, Illustrations of predicted PPBS for an enzyme (barnase, PDB ID 1BRS, ref. [57], Val. Homology dataset) with its inhibitor overlaid (**e**) and an homodimer (glutamic acid decarboxylase GAD67, PDB ID 2OKJ, ref. [58], test topology dataset) (**f**). The molecular surface of the query protein is shown with coloring based on predicted probability, ranging from low (white) to high (dark blue). The partner protein is shown in cartoon representation (gray transparent). Visualization software, ChimeraX[59].

the origin corresponds to the central atom and the z axis and xz plane are oriented according to its covalent bonds. Figures 3a–f and 4a–f each show one pattern (left), together with a maximally activating neighborhood (right) taken from the validation set and the remaining patterns are provided in Supplementary Data 1, 2. The atomic pattern shown in Fig. 3a has two main components: a NH group located at the center and an oxygen located few Ångstroms away, in the $(x<0, y<0, z<0)$ quadrant, that is, opposite from the two covalent bonds. It is the well-known signature of a N–H–O hydrogen bond, ubiquitous in protein backbones. The corresponding maximally activating atom is indeed a backbone nitrogen within a beta sheet. Patterns may have more than two components, and several possible groups per location. The atomic pattern shown in Fig. 3b features two oxygen atoms and three NH groups in a specific arrangement; the corresponding maximally activating neighborhoods are backbone nitrogens located at contact zones between two helical fragments (right of Fig. 3b and Supplementary Fig. 8). Patterns shown in Fig. 3c,d focus on sidechains. The pattern in Fig. 3c is defined as a carbon in the vicinity of a methyl group and an aromatic ring. The pattern in Fig. 3d consists of SH or NH_2 groups—two sidechain-located hydrogen donors—surrounded by oxygen atoms. Last, patterns may include prescribed absence of atoms in specific regions. The pattern in Fig. 3e is defined by a backbone carbon or oxygen without any NH groups in its vicinity, meaning that it identifies backbones available for hydrogen bonding. The pattern in Fig. 3f identifies a methionine sidechain with one solvent-exposed side, and is associated with high PPBS probability. Together, the filters collectively define a rich representation capturing various properties of a neighborhood, as seen from the 2D t-distributed stochastic neighbor embedding (t-SNE) projections colored by properties (Fig. 3g,h). In the space of filter activities, atoms cluster by coordination number (number of other atoms in the range of van der Waals interactions) and electrostatic potential (calculated with the Adaptive Poisson–Boltzmann Solver[48]).

The amino acid scale patterns can be similarly analyzed: the origin, z axis and xz plane are, respectively, defined by the C_α, sidechain and backbone orientation of the central amino acid. Neighborhoods are shown as backbone segments, with position–weight matrices as attributes; the learned attributes pooled from the atomic scale are not shown. Each Gaussian component of a pattern is characterized

Table 1 | Performance evaluation for prediction of PPBs. AUCPR is shown. Proteins of the test set are subdivided into four nonoverlapping groups as described in Fig. 2. For the Masif-site, only the aggregated performance is shown since its training set differs from ours. See Supplementary Tables 1–6 for additional evaluation metrics and variance estimates. Bold entries indicate the best performance

Algorithm	Test (70%)	Test (homology)	Test (topology)	Test (none)	Test (all)
Structural homology baseline	**0.828**	0.696	0.535	0.387	0.613
Handcrafted features baseline	0.596	0.567	0.568	0.432	0.537
Masif-site[36]	NA	NA	NA	NA	0.533
ScanNet	0.732	**0.712**	**0.735**	**0.605**	**0.694**
ScanNet (no evolutionary information)	0.672	0.648	0.685	0.565	0.639
ScanNet (no atomic information)	0.697	0.672	0.689	0.547	0.648
ScanNet (no regularization)	0.756	0.702	0.701	0.572	0.678
ScanNet (no reweighting)	0.702	0.668	0.683	0.553	0.648

by a complex specificity in attribute space. We represent it by the distributions of amino acid types and accessible surface areas of its top 1% maximally activating residues. Patterns in Fig. 4a,b focus only on the central amino acid, that is, they recombine and propagate features from the previous layers. The pattern in Fig. 4a consists of solvent-exposed residues of type frequently encountered in protein–protein interfaces such as leucine or arginine. It is positively correlated with the output probability ($r=0.31$). Conversely, the pattern in Fig. 4b, which consists of buried hydrophobic amino acids, is activated by residues within the protein cores and is negatively correlated with the output ($r=-0.32$).

Multi-component patterns are also found: the pattern in Fig. 4c consists of an exposed glycine together with an exposed aromatic or leucine amino acid, and is correlated with binding ($r=0.18$). The pattern in Fig. 4d is constituted by an exposed hydrophobic amino acid surrounded by exposed, charged amino acids and is strongly correlated with binding ($r=0.29$). It is similar to the hotspot O-ring architecture previously described by Bogan and Thorn[20]. Conversely, the pattern in Fig. 4e, which consists of a central cysteine (possibly involved in a disulfide bond) surrounded by exposed lysines is negatively correlated with binding ($r=-0.13$).

Distributed patterns such as that in Fig. 4f are found and hypothetically contribute to prediction by identifying domain-level context. The pattern in Fig. 4f, which consists of multiple aromatic and hydrophobic components, is strongly activated by transmembrane helical domains. Identification of transmembrane domain is indeed required for accurate prediction as the hydrophobic core/hydrophilic rim rule is reversed within membranes. Inversely, we expect that for disordered regions, only the filters with patterns focusing on a single amino acid such as Fig. 4a,b or a linear stretch such as Fig. 4c will contribute to the prediction, whereas the others will be silent. ScanNet will thus effectively behave as a convolutional sequence model with a short kernel width.

Finally, the two-dimensional t-SNE projections of the representation (Fig. 4f,g and Extended Data Fig. 7) show that the filter activities encompass various amino acid-level handcrafted features, including amino acid type, secondary structure, accessible surface area, surface convexity and evolutionary conservation.

Overall, these findings support the hypothesis that ScanNet learns some of the underlying physico-chemical principles of PPIs. To consolidate these findings, we compared ScanNet predictions to experimental alanine scans and residue contributions to the binding energy using Rosetta (Methods and Extended Data Fig. 8). We found that among the binding residues, the ones with higher binding probability and larger attention coefficients tend to contribute more to the binding free energy. Additionally, the amino acid filter activities reflected the type of interaction (van der Waals, electrostatic and so on) involved in binding.

ScanNet for prediction of BCEs. BCE are defined as residues directly involved in a antibody–antigen complex. Although a priori every surface residue is potentially immunogenic, some are preferred in the sense that it is easier to mature antibodies targeting them with high affinity and specificity. Exhaustive, high-throughput experimental determination of BCEs is challenging because they can span across multiple noncontiguous protein fragments. Prediction is challenging owing to their instability throughout evolution and the lack of exhaustive epitope mappings for a given antigen. In silico prediction of BCE can be leveraged for constructing epitope-based vaccines and for designing nonimmunogenic therapeutic proteins.

We derived from the SabDab database[49] a dataset of 3,756 protein chains (796, 95% sequence identity clusters) with annotated BCE. Here, 8.9% of the residues were labeled as BCE, likely an underestimation of the true fraction. The dataset was split into five subsets for cross-validation training, with no more than 70% sequence identity between pairs of sequences from different subsets. We evaluated ScanNet in three settings: trained from scratch, trained for PPBS prediction without finetuning and trained via transfer learning using the PPBS network as starting point. We compared it with the handcrafted features baseline, structural homology baseline and Discotope, a popular tool based on geometric features and propensity scores[50]. We also report the performance of ScanNet without evolutionary data, of the null predictor and of a predictor based on solvent accessibility only. ScanNet trained via transfer learning outperformed the other models, with an AUCPR of 0.178 and a positive predicted value at L/10 of 27.5% (Fig. 5a and Supplementary Table 5). This represents an enrichment of respectively 143, 153 and 309% over Discotope, solvent accessibility-based and null prediction. ScanNet performed equally well with or without evolutionary information unlike for PPBS. Visualization of representative spatio-chemical patterns associated with high BCE probability sheds light on the similarities and differences between PPBS and BCE (Fig. 5b–e, the remaining filters are provided in Supplementary Data 3). We find asparagine and arginine-containing patterns (Fig. 5b,c) as well as linear epitopes (Fig. 5c, shared with PPBS). The pattern in Fig. 5d consists of exposed residues with alternate charges, and putatively indicates availability for salt-bridge formation. Finally, pattern Fig. 5e is composed of an exposed, charged amino acid in the vicinity of two cysteines forming a disulfide bond. A possible explanation is that disulfide bond-rich regions are more structurally stable, hence it is easier to recognize with high affinity and specificity.

We next predicted and visualized BCE of the SARS-CoV-2 spike protein. Predictions are shown with representative antibodies superimposed for the trimer with one open receptor binding domain (RBD) (Fig. 5e) and for the isolated RBD and N-terminal domain (NTD) (Supplementary Fig. 9). For the spike protein, the RBD was correctly identified as a major antigenic site. The six main epitopes previously described[51] all had high probabilities, including the cryptic epitope CR3022 (exposed in the open conformation). The tip of the NTD was also correctly identified as a highly antigenic site. Two linear epitopes located

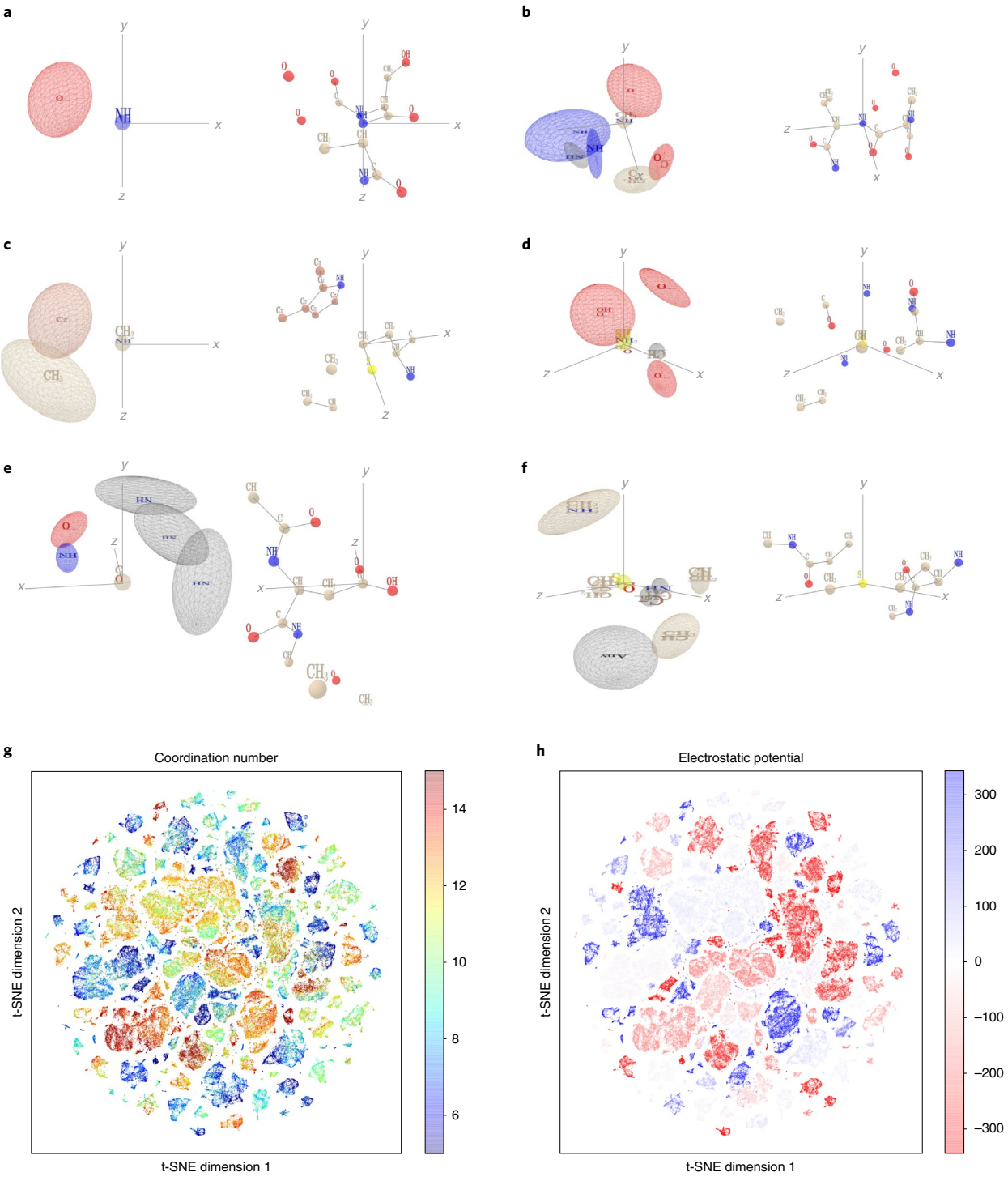


Fig. 3 | Visualization of the learned atomic representation. a–f, Each panel shows one of the 128 learned spatio-chemical patterns on the left and one corresponding top-activating neighborhood on the right. Each pattern is depicted as follows: only the Gaussian kernels relevant to the pattern are shown; they are represented by their unit ellipsoid. The corresponding location-wise attribute specificity is depicted as a weight logo inside the ellipsoid, similar to a position–weight matrix: attributes with nonzero weights are stacked on top of one another with letter height proportional to their algebraic weight value, sorted from strongest positive (top) to strongest negative (bottom, reversed letters). The unit ellipsoid is colored based on the maximally activating attribute type if it is positive, or gray otherwise. Color code is carbon (beige), oxygen (red), nitrogen (blue) and sulfur (yellow). The frame is overlaid in gray, with axes extending over 3.75 Å. Each filter/neighborhood pair is oriented independently for clarity. Visualizations created with pythreejs. **g**,**h**, Two-dimensional projection of the learned atomic-scale representation using t-SNE[60]. Each point corresponds to one atom of a representative set of proteins. Coloring is based on atom coordination index (**g**) or electrostatic potential at the atom location, computed using APBS[48] (**h**).

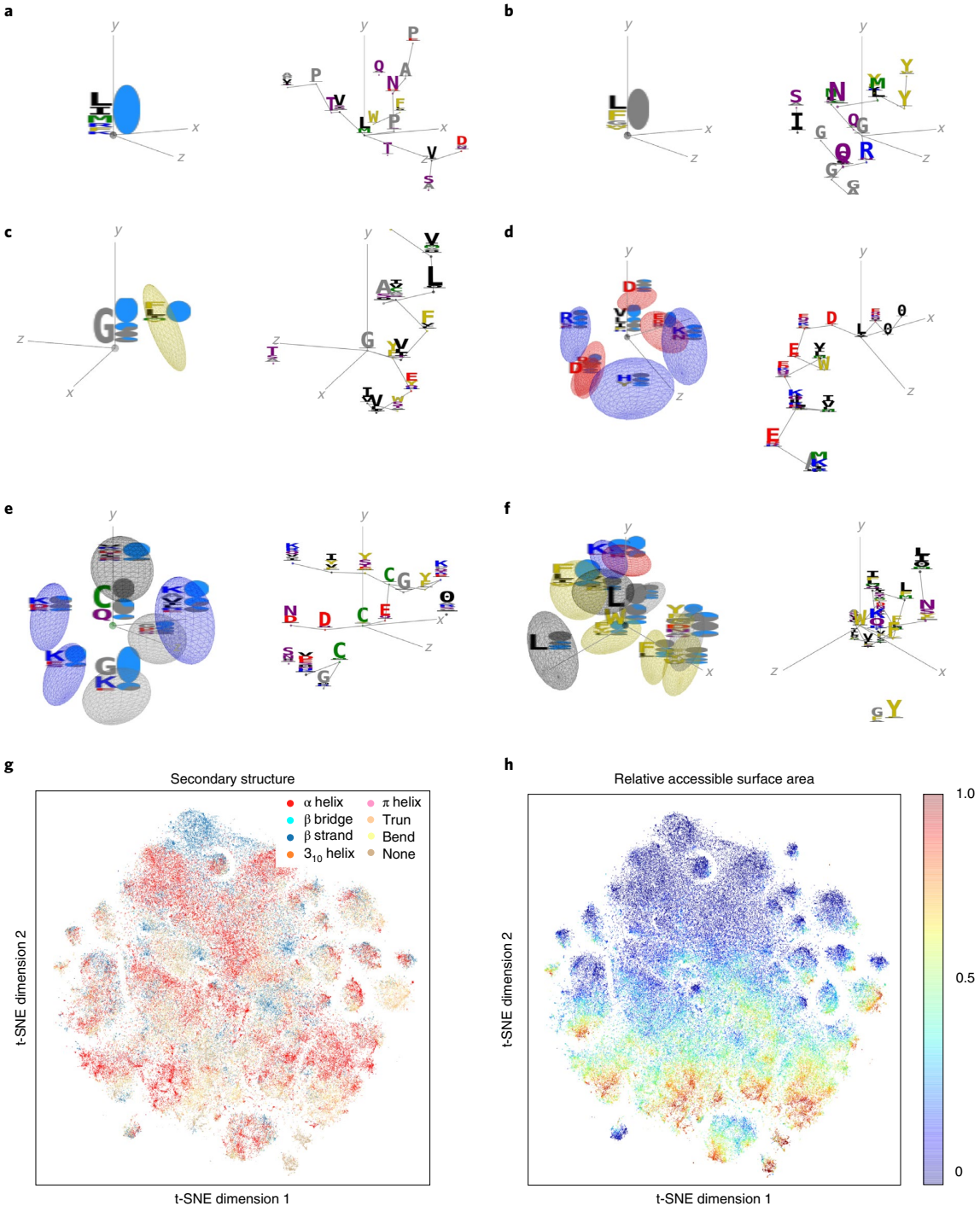


Fig. 4 | Visualization of the learned amino acid representation. a–f, Each panel shows one of the 128 learned spatio-chemical patterns on the left and one corresponding top-activating neighborhood on the right. Gaussian kernels are depicted similarly to those in Fig. 3. Since the input attributes are learned, each component of a pattern is characterized by a complex specificity in attribute space. We represent it by the distributions of amino acid types and accessible surface areas of its top 1% maximally activating residues. The distributions are shown as a logo (each letter or symbol is proportional to the probability), with a total height proportional to the mean activation of the set. Accessible surface area values are discretized into four quartiles and represented as pie charts (from full gray meaning buried to full blue meaning accessible). Amino acids are colored by chemical properties: negatively charged (red), positively charged (blue), polar (purple), hydrophobic (black), sulfur-containing (green), aromatic (gold) and tiny/proline (gray). The frame is overlaid in gray, with axes extending over 9 Å. Each filter/neighborhood pair is oriented independently for clarity. Visualizations created with pythreejs. **g**,**h**, Two-dimensional projection of the learned amino acid scale representation using t-SNE[60]. Each point corresponds to one amino acid of a representative set of proteins. Coloring based on secondary structure (**g**) or accessible surface area (**h**) calculated with DSSP[61]. Additional t-SNE plots are available in Extended Data Fig. 7.

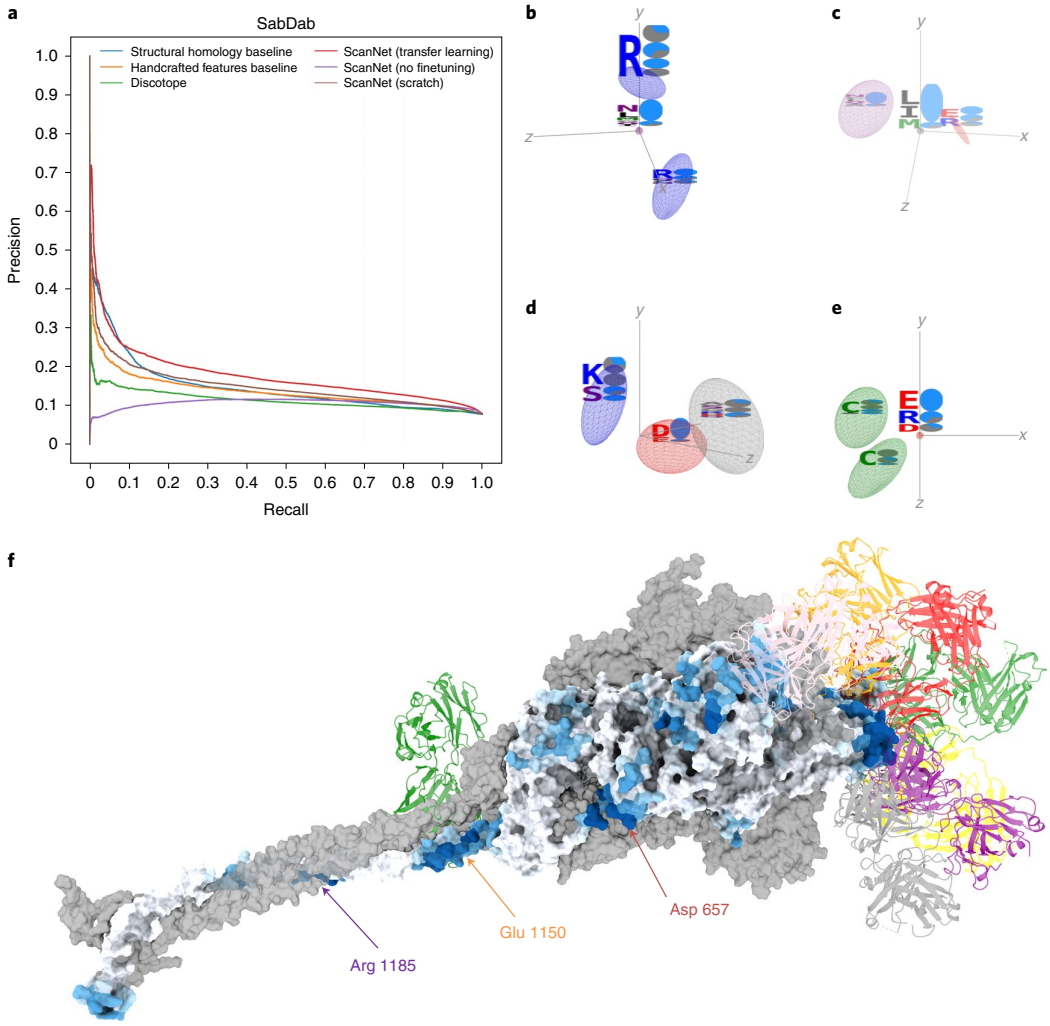


Fig. 5 | Prediction of BCEs with ScanNet. a, Precision-recall curve of BCE prediction for baseline methods, Discotope[50] and ScanNet. Epitope database constructed from SabDab (timestamp 19 April 2021)[49]; fivefold cross-validation performance is shown. **b**–**e**, Each panel shows one learned spatio-chemical pattern whose activity is positively correlated with epitope probability. Same visualization as Fig. 4. **f**, Application to spike protein of SARS-CoV-2. Predictions performed on a Molecular Dynamics snapshot of the spike trimer with one RBD open[62]. The monomer with open conformation is represented as a molecular surface with colors corresponding to BCE probability, from white (low) to dark blue (high). Representative antibodies binding the main epitopes are superimposed in color, cartoon representation, see the full list in Supplementary Table 6.

in the S2 fusion machinery are also predicted around Glu 1150 and Arg 1185, respectively. Previously, Shrock et al.[52] reported that both regions were targeted by antibodies from patients who had recovered from COVID-19. For the first one, a broadly neutralizing mAB targeting this epitope was recently isolated[53] and shown to neutralize several beta-coronaviruses but not SARS-CoV-2. Finally, the network predicted with high confidence one previously unreported conformational epitope constituted by three fragments in the vicinity of the glycosylated[54] Asn 657. Provided that the network is correct, and since the presence of the glycosyl group is unknown at runtime but can be imputed by ScanNet from the Asn-X-Ser/Thr linear motif, two interpretations are possible: either the glycosyl group shields an otherwise highly immunogenic region from antibodies, or it directly induces immune response via glycosyl-binding antibodies. Similarly, we found two additional cryptic epitopes of the NTD that are centered on glycosylated asparagine when performing prediction on the NTD domain alone (Supplementary Fig. 9b).

Overall, ScanNet predictions are in excellent agreement with the known antigenic profile of the spike protein and predict a new epitope that could not be detected via high-throughput linear epitope scanning. We additionally predicted BCE for three other viral protein: HIV envelope protein, influenza HA-1 and influenza HA-3 hemagglutinin (Supplementary Fig. 10). We notably found that the hemagglutinin epitope predictions differed between the HA-1 and HA-3 strand despite the similar fold, suggesting that ScanNet could be suitable for studying antigenic drift.

Discussion

Protein function is borne of a diverse set of structural motifs. These motifs, characterized by their complex spatio-chemical arrangements of atoms and amino acids, cannot be fully encompassed by handcrafted features. Conversely, detection via comparative modeling is challenging because their invariants, that is, the set of function-preserving sequence/conformational perturbations, are unknown. ScanNet is an end-to-end geometric deep learning

model capable of learning such motifs together with their invariants directly from raw structural data by backpropagation. We demonstrated, through a detailed comparison of newly compiled datasets of annotated PPBSs and BCEs, that it efficiently leverages these motifs to outperform feature-based methods, comparative modeling and surface-based geometric deep learning. ScanNet reaches an accuracy of 87.7% for PPBS prediction and a positive prediction value at L/10 of 27.5% for BCE prediction. Through appropriate parameterization and regularization, the spatio-chemical patterns learned by the model can be explicitly visualized and interpreted as previously known motifs and as new ones. A breakthrough was recently achieved in protein structure prediction using deep learning[2], leading to the release of a vast set of accurate protein structure models[3]. We anticipate that ScanNet will prove insightful for analyzing these proteins, of which little is known regarding their function. A webserver is made available at http://bioinfo3d.cs.tau.ac.il/ScanNet/ and linked to both the Protein Data Bank (PDB) and AlphaFoldDB for ease of use. Very recently, Evans et al.[55] introduced AlphaFold-multimer, a new approach for prediction of protein complexes from paired MSAs and demonstrated impressive performance. We further compared ScanNet to AlphaFold-mutimer for prediction of partner-specific PPBSs, partner-agnostic PPBSs and BCEs (Methods and Extended Data Fig. 9). We found that AlphaFold outperformed ScanNet for partner-specific PPBS, whereas both performed comparably for partner-agnostic PPBS. For BCEs, ScanNet could identify all the main epitopes of the RBD of the SARS-CoV-2 spike protein, whereas AlphaFold-multimer could only identify one. This showcases the complementarity between MSA-based, partner-specific and structure-based, partner-agnostic approaches. Owing to its generality, it is straightforward to extend ScanNet to other classes of binding sites provided that sufficient training data is available. Extension to partner-specific binding prediction for prediction of interactions and guiding molecular docking is a promising future direction, as the amino acid filter activities are correlated between interacting binding sites (Extended Data Fig. 10). Meanwhile, the learned atom-wise and amino acid-wise representations can be readily used as drop-in replacement for handcrafted features in any structure-based machine learning pipeline. A second class of applications is protein design: ScanNet, which is differentiable with respect to its inputs and does not require evolutionary information, could be used in conjunction with structure prediction tools to guide design of proteins with prescribed binding or nonbinding properties (for example, nonimmunogenic therapeutic proteins).

Finally, interpretable, end-to-end learning, combined with self-supervised learning techniques could pave the way toward a complete dictionary of function-bearing structural motifs found in nature, deepening our understanding of the core principles underlying protein function.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41592-022-01490-7.

Received: 5 September 2021; Accepted: 12 April 2022;
Published online: 30 May 2022

References

1. Kühlbrandt, W. The resolution revolution. *Science* **343**, 1443 (2014).
2. Jumper, J. et al. Highly accurate protein structure prediction with alphafold. *Nature* **596**, 583–589 (2021).
3. Tunyasuvunakool, K. et al. Highly accurate protein structure prediction for the human proteome. *Nature* **596**, 590–596 (2021).
4. Chruszcz, M., Domagalski, M., Osinski, T., Wlodawer, A. & Minor, W. Unmet challenges of structural genomics. *Curr. Opin. Struct, Biol.* **20**, 587 (2010).
5. Shulman-Peleg, A., Nussinov, R. & Wolfson, H. J. Site engines: recognition and comparison of binding sites and protein–protein interfaces. *Nucleic Acids Res.* **33**, W337 (2005).
6. Carl, N., Konc, J., Vehar, B. & Janezic, D. Protein-protein binding site prediction by local structural alignment. *J. Chem. Info. Model.* **50**, 1906 (2010).
7. Zhang, Q. C., Petrey, D., Norel, R. & Honig, B. H. Protein interface conservation across structure space. *Proc. Natl Acad. Sci. USA* **107**, 10896 (2010).
8. Xue, L. C., Dobbs, D. & Honavar, V. HOMPPI: a class of sequence homology based protein-protein interface prediction methods. *BMC Bioinformatics* **12**, 1 (2011).
9. Shoemaker, B. A. et al. IBIS (inferred biomolecular interaction server) reports, predicts and integrates multiple types of conserved interactions for proteins. *Nucleic Acids Res.* **40**, D834 (2012).
10. Jordan, R. A., Yasser, E.-M., Dobbs, D. & Honavar, V. Predicting protein-protein interface residues using local surface structural similarity. *BMC Bioinformatics* **13**, 1 (2012).
11. Esmaielbeiki, R. & Nebel, J.C. Unbiased Protein Interface Prediction Based on Ligand Diversity Quantification, in *Proc. German Conference on Bioinformatics 119*; 19-22 Sep 2012, Jena, Germany. (OASICS, no. Vol. 26) ISSN (print) 2190-6807 ISBN 9783939897446. Editors: S. Bocker, F. Hufsky, K. Scheubert, J. Schleicher and S. Schuster (2012).
12. Xue, L. C., Dobbs, D., Bonvin, A. M. & Honavar, V. Computational prediction of protein interfaces: a review of data driven methods. *FEBS Lett.* **589**, 3516 (2015).
13. Esmaielbeiki, R., Krawczyk, K., Knapp, B., Nebel, J.-C. & Deane, C. M. Progress and challenges in predicting protein interfaces. *Brief. Bioinform.* **17**, 117 (2016).
14. Neuvirth, H., Raz, R. & Schreiber, G. Promate: a structure based prediction program to identify the location of protein–protein binding sites. *J. Mol. Biol.* **338**, 181 (2004).
15. Chung, J.-L., Wang, W. & Bourne, P. E. Exploiting sequence and structure homologs to identify protein–protein binding sites. *Proteins* **62**, 630 (2006).
16. Porollo, A. & Meller, J. Prediction-based fingerprints of protein–protein interactions. *Proteins* **66**, 630 (2007).
17. Sweredoski, M. J. & Baldi, P. Pepito: improved discontinuous B-cell epitope prediction using multiple distance thresholds and half sphere exposure. *Bioinformatics* **24**, 1459 (2008).
18. Mishra, S. K., Kandoi, G. & Jernigan, R. L. Coupling dynamics and evolutionary information with structure to identify protein regulatory and functional binding sites. *Proteins.* **87**, 850 (2019).
19. Klug, A. & Rhodes, D. 'Zinc fingers': a novel protein motif for nucleic acid recognition. *Trends Biochem. Sci.* **12**, 464 (1987).
20. Bogan, A. A. & Thorn, K. S. Anatomy of hot spots in protein interfaces. *J. Mol. Biol.* **280**, 1–9 (1998).
21. Wensien, M. et al. A lysine–cysteine redox switch with an NOS bridge regulates enzyme function. *Nature* **593**, 460 (2021).
22. Elnaggar, A. et al. ProtTrans: Towards Cracking the Language of Lifes Code Through Self-Supervised Deep Learning and High Performance Computing. IEEE Transactions on Pattern Analysis and Machine Intelligence. (2021)
23. Riveset, A. al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl Acad. Sci. USA* https://doi.org/10.1073/pnas.2016239118 (2021).
24. Ingraham, J., Riesselman, A., Sander, C. & Marks, D. Learning protein structure with a differentiable simulator, in *Proc. International Conference on Learning Representations* (2018). Venue: Vancouver, Canada. Editors: Y. Bengio, Y. LeCun, T. Saintath, I. Murray, M.A. Ranzato, O. Vinyals, A. Courville & H. Larochelle.
25. Ingraham, J., Garg, V., Barzilay, R. & Jaakkola, T. Generative models for graph-based protein design. In. *Proc. Advances in Neural Information Processing Systems 32* (NeurIPS, 2019).
26. Jing, X., & Xu, J. (2021). Fast and effective protein model refinement using deep graph neural networks. Nature Computational Science, 1(7), 462-469.
27. Baldassarre, F., Menéndez Hurtado, D., Elofsson, A. & Azizpour, H. Graphqa: protein model quality assessment using graph convolutional networks. *Bioinformatics* **37**, 360 (2021).
28. Wallach, I., Dzamba, M. & Heifets, A. Atomnet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. Preprint at arXiv:1510.02855 (2015).
29. Ragoza, M., Hochuli, J., Idrobo, E., Sunseri, J. & Koes, D. R. Protein–ligand scoring with convolutional neural networks. *J. Chem. Inform. Model.* **57**, 942 (2017).
30. Pagès, G., Charmettant, B. & Grudinin, S. Protein model quality assessment using 3D oriented convolutional neural networks. *Bioinformatics* **35**, 3313 (2019).
31. Townshend, R., Bedi, R., Suriana, P. & Dror, R. End-to-end learning on 3D protein structure for interface prediction. *Adv. Neural Inform. Proc. Syst.* **32**, 15642 (2019).

32. Wang, X., Terashi, G., Christoffer, C. W., Zhu, M. & Kihara, D. Protein docking model evaluation by 3D deep convolutional neural networks. *Bioinformatics* **36**, 2113 (2020).
33. Igashov, I., Olechnovič, K., Kadukova, M., Venclovas, Č., & Grudinin, S. VoroCNN: deep convolutional neural network built on 3D Voronoi tessellation of protein structures. *Bioinformatics*, 37(16), 2332-2339. (2021)
34. DeepRank: a deep learning framework for data mining 3D protein-protein interfaces. *Nature communications*, 12(1), 1-8 (2021).
35. Eismann, S., Suriana, P., Jing, B., Townshend, R. J. & Dror, R. O. Protein model quality assessment using rotation-equivariant, hierarchical neural networks. Preprint at arXiv:2011.13557 (2020).
36. Gainza, P. et al. Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning. *Nat. Methods* **17**, 184 (2020).
37. Sverrisson, F., Feydy, J., Correia, B. E. & Bronstein, M. M. Fast end-to-end learning on protein surfaces, in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 15272–15281 (IEEE, 2021). Venue: Virtual. Editors: M. S. Brown, R. Sukthankar, T. Tan & L. Zelnik
38. Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A. & Vandergheynst, P. Geometric deep learning: going beyond Euclidean data. *IEEE Sig. Process.* **34**, 18–42 (2017).
39. Bronstein, M. M., Bruna, J., Cohen, T. & Veličković, P. Geometric deep learning: grids, groups, graphs, geodesics, and gauges. Preprint at arXiv:2104.13478 (2021).
40. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry, in *Proc. International Conference on Machine Learning* 1263–1272 (PMLR, 2017). Venue: Sydney, Australia. Editors: D. Precup, Y. W. Teh
41. Veličković, P. et al. Graph attention networks. Preprint at arXiv:1710.10903 (2017).
42. Keskin, O., Ma, B. & Nussinov, R. Hot regions in protein–protein interactions: the organization and contribution of structurally conserved hot spot residues. *J. Mol. Biol.* **345**, 1281 (2005).
43. Ofran, Y. & Rost, B. Protein–protein interaction hotspots carved into sequences. *PLoS Comput. Biol.* **3**, e119 (2007).
44. Dey, S., Ritchie, D. W. & Levy, E. D. PBD-wide identification of biological assemblies from conserved quaternary structure geometry. *Nat. Methods* **15**, 67 (2018).
45. Kundrotas, P. J. et al. Dockground: a comprehensive data resource for modeling of protein complexes. *Protein Sci.* **27**, 172 (2018).
46. Chen, T. & Guestrin, C. Xgboost: a scalable tree boosting system, in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 785–794 (ACM, 2016). Venue: San Francisco, CA, USA. Editors: R. Krishnapuram, M. Shah, A. Smola, C. Aggarwal, D. Shen & R. Rastogi
47. Shatsky, M., Nussinov, R. & Wolfson, H. J. Multiprot–a multiple protein structural alignment algorithm, in Proc. *International Workshop on Algorithms in Bioinformatics* 235–250 (Springer, 2002). Venue: Rome, Italy. Editors: R. Guigó & D. Gusfield
48. Jurrus, E. Improvements to the apbs biomolecular solvation software suite. *Protein Sci,* **27**, 112 (2018).
49. Dunbar, J. et al. Sabdab: the structural antibody database. *Nucleic Acids Res.* **42**, D1140 (2014).
50. Kringelum, J. V., Lundegaard, C., Lund, O. & Nielsen, M. Reliable b cell epitope predictions: impacts of method development and improved benchmarking. *PLoS Comput. Biol.* **8**, e1002829 (2012).
51. Yuan, M. et al. Structural and functional ramifications of antigenic drift in recent SARS-CoV-2 variants, *Science* **373**, 818–823 (2021).
52. Shrock, E. et al. Viral epitope profiling of COVID-19 patients reveals cross-reactivity and correlates of severity. *Science* **370**, eabd4250 (2020).
53. Sauer, M. M. et al. Structural basis for broad coronavirus neutralization. *Nature Struct. Mol. Biol.* **28**, 478–486 (2021).
54. Watanabe, Y., Allen, J. D., Wrapp, D., McLellan, J. S. & Crispin, M. Site-specific glycan analysis of the SARS-CoV-2 spike. *Science* **369**, 330 (2020).
55. Evans, R. et al. Protein complex prediction with alphafold-multimer. Preprint at *bioRxiv* (2021).
56. Sillitoe, I. et al. CATH: increased structural coverage of functional space. *Nucleic Acids Res.* **49**, D266 (2021).
57. Buckle, A. M., Schreiber, G. & Fersht, A. R. Protein-protein recognition: crystal structural analysis of a Barnase-Barstar complex at 2.0-. ang. resolution. *Biochemistry* **33**, 8878 (1994).
58. Fenalti, G. et al. Gaba production by glutamic acid decarboxylase is regulated by a dynamic catalytic loop. *Nat. Struct. Mol. Biol.* **14**, 280 (2007).
59. Goddard, T. D. et al. UCSF chimerax: meeting modern challenges in visualization and analysis. *Protein Science* **27**, 14 (2018).
60. Van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *J. Mach. Learning Res.* **9**, 2579–2605 (2008).
61. Kabsch, W. & Sander, C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers: Original Res. Biomolecules* **22**, 2577 (1983).
62. Amaro, R. & Mulholland, A. Biomolecular simulations in the time of COVID19, and after. *Comput. Sci. Eng.* **22**, 30–36 (2020).

Methods

This section is organized as follows. The first subsection provides all mathematical and implementation details for ScanNet. The next subsection is dedicated to the baseline methods. Then the dataset construction, partition and sample weights are covered. After that, we evaluate the impact of induced fit on changes on ScanNet predictions. We then compare ScanNet to AlphaFold-multimer. The link between ScanNet prediction and binding site predictions is then covered and finally the additional results for the PPBS and BCE prediction tasks are discussed.

ScanNet network. *Preprocessing.* For PDB parsing, the PDB files are parsed using Biopython[63]. We gather, for each chain, the amino acid sequence and the point cloud of heavy atoms, formally a list of triplets $\{(\text{coordinates}_l, \text{residueid}_l, \text{atomid}_l)\ l \in [[1, N_{\text{atoms}}]]\}$, for example, ([10.1, 101.3, −12.6], 97, CA). Only atoms belonging to classical residues are considered; exotic residues, additional molecules bound to the chain (for example, heme, ATP, glycosyl groups, ions...) are excluded.

Toward definition of a local reference frame for each atom, we reconstruct the molecular graph (that is, atom as nodes and covalent bonds as edges) using the residue and atom IDs. Each heavy atom has one, two or three neighbors on the molecular graph; if it has only one (for example, for methyl group CH_3), a virtual hydrogen atom is appended to the graph. Two neighbors are selected to define a triplet of points $(l, i_{\mathcal{N}_1(l)}, i_{\mathcal{N}_2(l)})$ from which a frame can be derived. The coordinates of atoms l, $i_{\mathcal{N}_1(l)}$ and $i_{\mathcal{N}_2(l)}$ respectively define the center, xz plane and z direction, see later about the frame computation module (FCM) and equation (3). The first ('previous') neighbor is chosen as the closest from the N-terminal nitrogen. For the second ('next') neighbor, if the atom has three neighbors, the furthest from the C-terminal carbon among the remaining two is used. If both are equally far away, for example, for isoleucine, we choose the first one according to the residue ID. For instance, the two neighbors of the C atom of residue l are the C_α of the residue l and the N of the residue $l+1$. The two neighbors of the C_β atom are the C_α atom and the C_γ atom of the sidechain.

Also based on the molecular graph, an attribute is assigned to each heavy atom based on its type and the number of bound hydrogens. Twelve categories are defined: C, CH, CH_2, CH_3, C_π (aromatic carbon), O, OH, N, NH, NH_2, S and SH. Overall, four atomic arrays are constructed:

- The point cloud of atoms and virtual atoms (float, size $(N_{\text{atoms}} + N_{\text{virtualatoms}}, 3)$).
- The triplets of indices for constructing atomic local frames (integer, size $(N_{\text{atoms}}, 3)$).
- The atom groups (integer, size $(N_{\text{atoms}},)$).
- The residue index of each atom (integer, size $(N_{\text{atoms}},)$).

For the amino acid level, four similar arrays are constructed. The point cloud consists of the C_α and the sidechain centers of mass (SCoM) of each amino acid. For glycines—which do not have a sidechain—a virtual SCoM is defined as $\mathbf{x}_{\text{SCoM}} = 3\mathbf{x}_{C_\alpha} - \mathbf{x}_C - \mathbf{x}_N$, where $\mathbf{x}_{C_\alpha}, \mathbf{x}_C, \mathbf{x}_N$ denote the vector coordinates of the C_α, N and C atom of the residue, respectively. The reference frame of each amino acid is defined by the C_α (center), previous C_α along the backbone (xz plane) and SCoM (z axis). Previous works[24,30] considered other amino acid frames constructed from the backbone atoms only. Here, our rationale was that neighboring amino acids located in the opposite direction from the sidechain (that is, the interior of the protein) should not matter for functionality. It also facilitates filter interpretation, as for exposed residues the sidechain points toward the exterior of the protein. We also experimented with frames constructed from consecutive C_α and found no difference performance-wise, but have not visualized the corresponding filters.

The per-residue attribute is given by the position–weight matrix (21-dimensional probability distribution, below) or the one-hot-encoded sequence for the models without evolutionary information.

For the derivation of the position–weight matrix, given the sequence, we first construct a MSA by homology search using HHblits 2 (four iterations, default values of other parameters)[64] on the UniClust30_2018_06 database[65] (except for the SARS-Cov-2 spike protein for which we used the UniRef30_2020_06). Next, a sequence dependent weight $w(S)$ was computed so as to (1) address sampling redundancy[66] and (2) focus the alignment around the wild type (WT)[67]:

$$w(S) = \frac{1}{\text{Number of 90\% sequence identity homologs}} \times \exp\left(-\frac{D_{\text{Hamming}}(S,WT)}{d_0}\right), \quad (1)$$

where d_0 is adjusted such that the effective number of samples is $B_{\text{eff}} \equiv \sum_S w(S) = 500$. If the alignment is initially too small, $d_0 = \infty$ is used. Focusing the alignments allows to detect local evolutionary conservation patterns as opposed to family-level conservation patterns; this is relevant as protein–protein interfaces are not always conserved at the superfamily level.

ScanNet modules. The following notations are used throughout presentation of the modules: x, global coordinates; f, frames; x^ℓ, local coordinates; a, attributes; a^ℓ, local attributes; L, size of point set; K, number of points in a neighborhood; D, dimension of coordinates; N or M, dimension of attributes and G, number of Gaussian kernels. All upper case letters are integer dimension numbers. The corresponding lower case letter denote running indices, for example, a_{ln} denotes the nth ($n \in ((1,N))$) attribute of the lth ($l \in ((1,L))$) point of the point cloud and x^ℓ_{lkd} is the dth local coordinate of the kth neighbor of point i. Bold letters denote vectors.

The attribute embedding module (AEM) applies an element-wise nonlinear transformation to the attributes a_{ln} of each point. Here, we used a element-wise dense layer, that is, a matrix product followed by ReLU nonlinearity for all AEM except for the initial atomic AEM, for which the input is a categorical variable and a one-hot encoding layer is applied. The equation for the AEM is written as:

$$a'_{lm} = \text{ReLU}\left[\sum_n a_{ln} w_{nm} + \theta_m\right] \quad (2)$$

The FCM takes as input a point cloud x_{ld} and a set of triplets of indices (i_{l1}, i_{l2}, i_{l3}) and calculates, for every triplet, a frame f'_{ldd} of size $[L, 4, 3])$, constituted by the center and the three unit vectors. The equation is written as:

$$\begin{aligned}
\mathbf{f}_{l1}\ (\text{center}) &= \mathbf{x}_{i_{l1}} \\
\mathbf{f}_{l4}\ (\text{z axis}) &= \frac{\mathbf{x}_{i_{l3}} - \mathbf{x}_{i_{l1}}}{||\mathbf{x}_{i_{l3}} - \mathbf{x}_{i_{l1}}||} \\
\mathbf{f}_{l3}\ (\text{y axis}) &= \frac{\mathbf{f}_{l4} \times (\mathbf{x}_{i_{l2}} - \mathbf{x}_{i_{l1}})}{||\mathbf{f}_{l4} \times (\mathbf{x}_{i_{l2}} - \mathbf{x}_{i_{l1}})||} \\
\mathbf{f}_{l2}\ (\text{x axis}) &= \frac{\mathbf{f}_{l3} \times \mathbf{f}_{l4}}{||\mathbf{f}_{l3} \times \mathbf{f}_{l4}||}
\end{aligned} \quad (3)$$

where $\times$ denotes the cross-product. Examples of frames overlaid on a protein structure are shown in Extended Data Fig. 1a,b. The FCM has no trainable parameters.

The neighborhood computation module determines, for each point, its K closest neighbors in space (including itself), computes their local coordinates and duplicates their attributes. Its inputs are a set of frames f_{lid} and attributes a_{ln}, and outputs are the neighborhoods x^ℓ_{lkd}, a^ℓ_{lkn}. The nearest neighbor search is implemented naively by computing distances between all pairs of frame centers. For the atomic and amino acid neighborhoods, we use as local coordinates the three Euclidean coordinates of the second frame center in the first frame and take $K = 16$. For the neighborhood attention module (NAM), we take $K = 32$ and use five coordinates: the distance between both frame centers $\| \mathbf{f}_{l1} - \mathbf{f}'_{l1} \|$, the dot product between the sidechain directions $\mathbf{f}_{l4}.\mathbf{f}'_{l4}$, the dot product between the sidechain directions and the center to center vectors $\mathbf{f}_{l4}.\frac{\mathbf{f}'_{l1} - \mathbf{f}_{l1}}{\|\mathbf{f}'_{l1} - \mathbf{f}_{l1}\|}$ (and symmetric) and the distance between amino acids along the sequence (clipped at $d_{\max} = 8$). They are shown as $d, \omega, \theta, \theta', d_{\text{sequence}}$ in Extended Data Fig. 1c. The neighborhood computation module has no trainable parameters.

The neighborhood embedding module (NEM) is the core module of ScanNet. NEM convolves each neighborhood with a set of trainable spatio-chemical filters, akin to convolutional filters in image CNNs (Fig. 1). Its inputs are a set of K points with local coordinates x^ℓ_{kd} and attributes a^ℓ_{kn}, where $k \in [1, K]$, $d \in [1, D]$ and $n \in [1, N]$, respectively, denote neighbor, coordinate and attribute indices. NEM outputs a set of M filter activities y_m. It is parameterized using $G = 32$ Gaussian kernels (as in ref. [68]) and a bilinear product as follows:

$$y_m = \text{ReLU}\left[\sum_{k,g,n} W^{sc}_{mgn} \mathcal{G}(\mu_g, \Sigma_g, x_k) a_{kn} + \sum_{k,g} W^s_{mg} \mathcal{G}(\mu_g, \Sigma_g, x_k) + W^b_m\right] \quad (4)$$

where $\mathcal{G}(\mu, \Sigma, \mathbf{x}) = \exp\left[-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu)\right]$ is a Gaussian kernel of center μ and (full) covariance matrix Σ, and W^{sc}, W^s, W^b are trainable tensors of sizes $[M,G,N]$, $[M,G]$, $[M,]$. See a graphical sketch in Extended Data Fig. 2d. The Gaussian kernels are trainable and shared between all filters of a given layer, see the implementation in Extended Data Fig. 2e.

The above parameterization offers several advantages over other choices such as multilayer perceptrons[25,69,70] or spherical harmonics[35,71]. First, it is straightforward to interpret: a filter m with large entries of the tensor W^{sc} for some g,n is positively activated by points having attribute n and located near the center of the Gaussian g. Similarly, the matrix W^s encodes attribute-independent spatial sensitivity and $\mathbf{W}^b$ is a bias vector. Second, localized filters, that is, filters detecting only one or few combinations of point/attributes can be obtained by simply enforcing sparsity of the weights W^{sc} and W^s via a regularization penalty. Third, the filters are guaranteed to have an almost compact support, as the Gaussian functions decay rapidly as $\|x\| \to \infty$). This ensures that the diameter of the neighborhood is effectively capped irrespective of the local point density—in particular for unpacked or disordered regions. Last but not least, the Gaussian kernels can be initialized using unsupervised learning, thereby improving performance and limiting run-to-run performance variance (initialization protocol detailed below).

For the sparsity regularization, we use the following combination of cost function and norm constraint:

$$\begin{aligned}\mathcal{R}_1^2(W^{sc}) &= \frac{\lambda_1^2}{2GN}\sum_m\left(\sum_{gn}|W_{mgn}^{sc}|\right)^2\\ \mathcal{R}_1^2(W^{s}) &= \frac{\lambda_1^2}{2G}\sum_m\left(\sum_{g}|W_{mg}^{s}|\right)^2\\ \sqrt{\sum_{gn}(W_{mgn}^{sc})^2} &= \sqrt{\frac{G}{K}},\forall m\end{aligned} \tag{5}$$

The so-called L_1^2 regularization (as previously described in ref. [72]) is a variant of the L_1 regularization ($\mathcal{R}_1(W^1)=\sum_{mgn}|W_{mgn}^1|$) that promotes homogeneity of the filter sparsity values. This can be seen from the expression of the gradients, which is written as:

$$\begin{aligned}\frac{\partial\mathcal{R}_1^2}{\partial W_{mgn}^{sc}} &= \left(\frac{\lambda_1^2}{GN}\sum_{gn}|W_{mgn}^{sc}|\right)\mathrm{sign}(W_{mgn}^{sc})\\ \frac{\partial\mathcal{R}_1}{\partial W_{mgn}^{sc}} &= \lambda_1\mathrm{sign}(W_{mgn}^{sc})\end{aligned} \tag{6}$$

The L_1^2 regularization is effectively a L_1 regularization with a filter-dependent regularization strength: filters that are sparse (respectively not sparse) have a small (respectively large) L_1 norm, hence a small (resp. large) effective L_1 regularization strength; which in turn further relaxes or tightens the sparsity constraint. The L_2 filter norm constraint is necessary to ensure a well-defined optimization problem because of the downstream batch norm layers. Indeed, the operation $W_{mgn}^1 \to \rho_m W_{mgn}^1$ leaves the final output invariant, as it is exactly compensated by the covariation of the slope of the subsequent batch norm layer through $\alpha_m \to \frac{\alpha_m}{\rho_m}$ (using notations from ref. [73]). Therefore, without constraint the optimum would be the asymptote $W_{mgn}^1 \to 0, \alpha_m \to \infty$ with $W_{mgn}^1 \times \alpha_m = W_{mgn}^{1\star}$, the optimum weight value without any regularization. The norm value is chosen such that the filter output y_m (equation (4)) has roughly variance 1 when the attributes have variance 1.

To determine the value of the regularization penalty λ_1^2, we searched for a satisfying compromise between interpretability (localized filters) and classification performance. We first determined the order of magnitude of λ_1^2 as follows: assuming filters weights W^1 with sparse entries (a fraction p of nonzero weight, with typical weight value W), the L_2 norm is written $\|W\|_2 = \sqrt{pGNW} \equiv \sqrt{G/K}$, that is, $W \sim \frac{1}{\sqrt{KNp}}$ and $\mathcal{R}_1^2 \sim \frac{\lambda pGM}{2K}$. Further assuming that the regularization penalties and cross-entropy variations (about 10^{-2} per site in our experiments) should approximately balance each other, and with $G/K=2, M=128$ for both atomic and amino acid filters, we find that $\lambda \cong 10^{-2}/pM$. With a target $p \cong 10^{-2}$, we conclude that $\lambda_1^2 \sim 10^{-2}$. After experimentation, we chose $\lambda_1^2 = 2.10^{-3}$ for both atomic and amino acid filters, as this value yielded the most satisfactory filter visualizations and prediction performances.

For atomic to amino acid pooling, toward calculation of residue-wise outputs, the learned atomic-scale representation must be aggregated at the amino acid scale. We recall that the constituting atoms of an amino acid may play different functional roles, hence symmetric pooling operations may not be sufficiently expressive. ScanNet instead uses a trainable multi-headed attention pooling. It is written as:

$$y_m^{\text{amino acid}} = \sum_{\text{atom},n} B_{mn}y_n^{\text{atom}}\frac{\exp\left[\sum_n A_{mn}y_n^{\text{atom}}\right]}{\sum_{\text{atom}}\exp\left[\sum_n A_{mn}y_n^{\text{atom}}\right]} \tag{7}$$

where B, A are trainable projection and attention weighting) matrices. Equation (7) generalizes the average pooling ($A_m=0$) and maximum pooling ($A_m=\alpha B_m$ with large α) operations. A sparsity regularization is also used for both B, A to simplify correspondence between atomic and amino acid filters.

The NAM computes spatially coherent, residue-wise output probabilities from amino acid frames and spatio-chemical filter activities. The computation is done in four stages (Extended Data Fig. 2a). First, local amino acid scale neighborhoods of size $K=32$ are constructed, with graph-type local coordinates: distances, angles and sequence distances (Extended Data Fig. 1c). Second, the five-dimensional edges are projected element-wise into a single algebraic value using trainable Gaussian kernels followed by a dense layer with linear activation function. No bias is used for the dense layer, such that the edge value decays to zero as the distance increases. Third, the filter activities are projected to scalar values and locally averaged using attention-based weights. Our expression of the weighting coefficients slightly differs from the graph attention network formulation of ref. [41] as follows: each node is characterized by a trainable output feature (unnormalized binding site probability), self-attention ('passenger' residues should have weak self-attention), cross-attention (hotspots should have strong cross-attention) and contrast coefficients (residues can follow either the majority or the hotspot residue). The weights may also take negative values depending on the edge values. Finally, a logistic function is applied to obtain normalized probabilities. Intuitively, the purpose of the NAM is to smooth out the probabilities such that if a residue has high binding propensity, its solvent-exposed neighbors should too. To this end, the NAM learns (1) a diffusion kernel on the residue–residue graph (the algebraic edges) and (2) importance coefficients for each node.

Full architecture. A diagram showing the architecture of the network is shown in Extended Data Fig. 2b and a table listing each module with its input(s) and output(s) sizes and comments is provided as Supplementary Table. In total, the network contains 475,000 parameters, of which about 200,000 are nonzero.

Training. For initialization, for the NEMs, the Gaussian kernels were initialized by unsupervised learning; using a subset of the training set, we computed atomic and amino acid neighborhoods and fitted the spatial point density using a Gaussian mixture model (as implemented in Scikit-learn[74], best of ten runs with *K*means++ initialization, full covariance matrix and 10^{-1} covariance matrix regularization). For the trainable graph edges of the NAM computed from distances and angles, we initialized them as a least square parametric fit of the label autocorrelation function (normalized):

$$A(\text{distance, angles, ...}) = \frac{\left(\mathbb{E}\left[Y_iY_j|d_{ij}=\text{distance, ...}\right]-\mathbb{E}[Y_i]^2\right)}{\mathbb{E}\left[Y_i\right]-\mathbb{E}[Y_i]^2} \tag{8}$$

Intuitively, this initialization choice corresponds to a diffusion kernel over the residue–residue graph. All remaining weights are initialized using symmetric random distributions, see details in the Supplementary Table.

For the padding and protein serialization trick, in our implementation, ScanNet takes as input an entire protein and computes neighborhoods on-the-fly, akin to a fully convolutional segmentation network[75]. Training on GPUs requires fixed size inputs but the lengths of proteins varied by almost two orders of magnitude in our dataset (Extended Data Fig. 3e). To avoid truncating large proteins or wasting most of the computational power, we used the following protein serialization trick. We choose a relatively large maximal protein length ($L_{\max}$ = 1024, 2120 for the PPBS and BCE datasets), concatenate several proteins into a single example and translate each protein far away from the others, such that no two proteins overlap in space. Since ScanNet exploits only local neighborhoods, the predictions for each protein are fully independent from one another. Before training or prediction, we group proteins in a greedy fashion that minimizes the unused placeholders. Proteins are first sorted by length and the largest ones are first picked; then, we pick among the remaining proteins the largest that fits into the placeholder (if any), concatenate it and continue until the placeholder is full. For the PPBS dataset, we found that about 96% of the amino acids placeholders were used, as opposed to less than 25% with naive padding. This results in a speed-up of about fourfold. Finally, we used masking layers across the network to prevent backpropagating errors for the remaining placeholders that do not contain any residue.

For optimization, the network is trained by minimizing the binary cross-entropy loss function by backpropagation using the ADAM optimizer[76]. We set the maximum number of epochs to 100, the batch size to 1, the learning rate to 10^{-3} (10^{-4} for the transfer learning) and perform learning rate annealing and early stopping based on the validation cross-entropy; the optimal model was usually reached before ten epochs. We used batch normalization layers before each ReLU nonlinearity throughout the network to avoid vanishing gradients. Finally, regarding sample weighting, a complication of the protein serialization trick is that residues of a single example may have different sample weight as they come from different proteins. To account for this, we formally replaced the binary cross-entropy loss function and logistic nonlinearity with a categorical cross-entropy and softmax function with two output classes; training labels are multiplied by their weight so as to replicate the weighted loss function.

Regarding software and runtime, the model was implemented in Python using the following scientific computing and machine learning packages: Python v.3.6.12; numpy v.1.19.5 (ref. [77]); h5py v.2.10.0; keras v.2.2.5 (ref. [78]); tensorflow v.1.14.0 (ref. [79]); biopython v.1.78 (ref. [63]); numba v.0.52.0 (ref. [80]); pandas v.1.1.5; scipy v.1.5.4 (ref. [81]); matplotlib v.3.3.3 and scikit-learn v.0.24.2 (ref. [74]). Training was completed in about 1–2 h using a single Nvidia V100 GPU. The inference time is dominated by the construction of the MSA and the calculation of the position–weight matrix—it is of the order of one to a few minutes depending on sequence length and MSA depth.

Baseline methods. *Handcrafted features baseline.* For the handcrafted features baseline, we computed for each amino acid geometric, chemical and evolutionary features as described in recent works on prediction of protein–protein/protein–antibody binding sites[12–18]. The following features were computed:

- Amino acid type (one-hot encoded, 20 dimensions).
- Secondary structure type (one-hot encoded, eight dimensions); computed with DSSP[61].
- Relative accessible surface area (one dimension); computed with DSSP[61].
- Coordination number (one dimension), defined as the number of C_α atoms in a ball of radius 13 center around the C_α atom of the amino acid.
- Half-sphere exposure index[82] (one dimension), defined as follows: let N_1 be the coordination number, and N_2 the number of C_α atoms in the intersection of a ball of radius 13 center and above the plane defined by the $C_\alpha - C_\beta$ vector. The half-sphere exposure index is $\frac{2N_2-N_1}{N1} \in [-1, 1]$.

- Backbone and sidechain depth[83] (two dimensions). The molecular surface was computed using MSMS (probe radius 1.5 Å)[84], and the distance to the surface was computed and averaged for all backbone (resp. sidechain) atoms.
- Surface convexity index (three dimensions)[85]. For each atom, we construct a ball of radius 5, 8 or 11 Å centered on it, and compute the f fraction of its volume located on the inside of molecular surface; the index is given by $2f-1 \in (-1,-1)$. The surface convexity index is averaged at the amino acid level.
- Position–weight matrix (21 dimensions)
- Conservation score $C = \log 21 + \sum_a \log PWM(a)$ (one dimension).

In total, 58 features were used. For classification, we used the xgboost algorithm (boosted trees)[46]. The classifier was trained by cross-entropy minimization, using the same training and validation sets. We used 100 boosting rounds (with early stopping on validation loss), and the following four parameters were determined by grid search: tree depth (5,10,20), minimum child weight (5,10,50,100), γ (0.01,0.1,1.0,5.) and η (0.5, 1.0).

Structural homology baseline. Several approaches leveraging sequence and structure homology were previously developed[5–10,10,11], but were not readily available for large scale benchmarking, which prompted us to develop an in-house structural homology baseline method. It features three key components:

(1) A nonredundant database of template protein chains with known binding sites. We used here as template the training set of ScanNet for a fair comparison. The template database was further clustered at the 90% (resp. 95%) sequence identity for the PPBS and BCE datasets, for speed gain purposes and to simplify alignment weighting (below).
(2) A local pairwise structure comparison engine. Compared to sequence homology or global structural homology, local structural homology were shown to outperform other methods in terms of coverage[7,10]. Here, we used MultiProt[47], an algorithm we previously developed that, given two proteins, outputs a set of local structural alignments.
(3) An alignment weighting scheme. Typically, MultiProt always finds at least few local alignments even when there is no homology between a query and a template protein, albeit with low coverage and low sequence identity. The alignments hence must be weighted so as to give higher importance to the alignments of highest quality[11]. Formally, for a given query protein with length L, MultiProt produces a set of R local alignments $\mathcal{A}_r$, $r \in [1, R]$. Each alignment is characterized by:

- The list of query residues included in the alignment, encoded as a binary vector:

$$\begin{cases} a_{r,l} = 1 & \text{if residue } l \in [1, L] \text{ in local alignment } \mathcal{A}_r \\ a_{r,l} = 0 & \text{otherwise} \end{cases} \quad (9)$$

- The coverage of the local alignment: $\text{Coverage}_r = \frac{1}{L}\sum_l a_{r,l}$
- The average root mean square deviation (r.m.s.d.) between matching pairs of C_α atoms
- The average sequence identity between query and template residues of the local alignment SeqID_r

Combining the alignment and the corresponding binding site labels of the templates, we define the following label alignment matrix:

$$\begin{cases} y_{r,l} = 1 & \text{if } a_{r,l} = 1 \text{ and label of aligned template residue } = 1 \\ y_{r,l} = 0 & \text{otherwise} \end{cases} \quad (10)$$

and write the predicted binding site probability as:

$$P_l = \frac{P_0 + \sum_{r=1}^{R} a_{r,l} y_{r,l} e^{\mathcal{W}(\text{Coverage}_r, \text{SeqID}_r, \text{r.m.s.d.}_r)}}{1 + \sum_{r=1}^{R} a_{r,l} e^{\mathcal{W}(\text{Coverage}_r, \text{SeqID}_r, \text{r.m.s.d.}_r)}} \quad (11)$$

where $\mathcal{W}(\text{Coverage, SeqID, r.m.s.d.})$ is a trainable log-weight function and P_0 is a pseudo-count regularization term, such that $P_l = P_0$ if no alignment is found for a given residue. The log-weight function $\mathcal{W}$ is parameterized by a two-layer perceptron with 20 hidden nodes and hyperbolic tangent activation function and was trained by cross-entropy minimization on a subset of the validation set; after training, we found that $\mathcal{W}$ is a increasing function of both alignment coverage and sequence identity, in agreement with our intuition that high coverage/sequence identity alignments should be favored. For P_0, we use the fraction of interface residues in the train set (resp. 0.22 and 0.09 for the PPBS and BCE train sets). Note that since the labels were already defined using multiple PDB files and redundancy reduction on templates was used, there was no need to further reweight alignments by ligand diversity as described in ref. [11].

As expected, the baseline performed very well when high quality homologs were available, and underperformed otherwise.

Masif-site. We used the Docker image of Masif-site as made available at https://github.com/LPDI-EPFL/masif. Masif-site predicts binding site propensity at the surface vertex level. To aggregate at the amino acid level, we followed the aggregation scheme provided for the Masif versus Sppider comparison (https://github.com/LPDI-EPFL/masif/blob/master/comparison/masif_site/masif_vs_sppider/masif_sppider_Intpred_comp.ipynb): each surface vertex was first assigned to its closest atom and corresponding amino acid and the binding site probability of an amino acid was taken as the maximum binding site probability over all its corresponding vertices. We stress that the comparison with Masif-site should be interpreted with caution, as: (1) Masif-site predicts at surface vertex level rather than amino acid level. Its residue-wise probabilities are therefore not calibrated, resulting in bad likelihood scores (Supplementary Table 2). (2) We did not retrain Masif-site because of limited computational resources and its training set used was smaller than ours. (3) Our test set overlaps with Masif-site training set, hence Masif-site should overperform on a fraction of our test set.

Discotope. We used the Discotope v.1.1 as made available at https://services.healthtech.dtu.dk/software.php. To emulate the behavior of Discotope v.2.0, which processes entire protein assemblies rather than individual protein chains[50], we fused each multi-chain antigens into a single chain, and verified on a few examples that the outputs were consistent with the ones from the Discotope v.2.0 webserver.

Data preparation. For the initial database and filtering, we use the Dockground database of protein–protein interfaces[45] (January 2020, full redundant version) as a starting point for our PPBS database. Each unique PDB chain involved in one interface or more was considered as a single example; we excluded chains with sequence length less than 10, chains involved in a protein–antibody complex (as classified in the SabDab database[49]) or designed proteins (identified as having two or more of the following red flags: no UniProt ID, no known CATH class, no sequence homologs found and engineered, synthetic, designed and/or de novo appearing in chain name). We obtained 70,583 unique chains (grouped in 20,025 clusters at 95% sequence identity) from 41,466 distinct PDB files, involved in 240,506 PPIs.

The dataset covers a wide range of complex sizes, types, organism taxonomies, protein lengths (Extended Data Fig. 3a–d). For the BCEs database, we used the SabDab database (timestamp 19 April 2021, ref. [49]) and included all antigens with length of ten or more forming an interface with an antibody with both heavy and light chain appearing in the PDB files. We obtained 3,756 chains (grouped in 796 clusters at 95% sequence identity).

Regarding data partition, for the PPBS database, we investigated the impact of homology between train and test set examples on generalization of ScanNet and our baseline models. We enforced a maximum sequence identity (90%) between a val/test example and any train set example, and grouped validation and test examples into four subgroups based on their degrees of homology (Extended Data Fig. 3g):

(1) Val/Test 70%: at least 70% sequence identity with at least one train set example.
(2) Val/Test homology: at most 70% sequence identity with any train set example, at least one train set example belonging to same protein superfamily (H level of CATH classification[56]).
(3) Val/Test topology: at least one train set example with similar protein topology (T level of CATH classification[56]), none with similar protein superfamily.
(4) Val/Test none: none of the above.

Subgroups are ordered by decreasing degree of homology; generalization is expected to be increasingly difficult. To ensure that the four subsets have approximately equal sizes, the following partitioning algorithm was used. The chains are first iteratively clustered by sequence identity at several levels (100%, 95%, 90% seqID, 70% seqID) using CD-HIT[86] followed by clustering at homology and topology identifiers. If a 70% (resp. homology) cluster contains several distinct homology (resp. topology) categories, these categories are merged into a single one. Next, we constructed the Val/Test none by randomly drawing topology clusters and assigning all its members to either validation and test; this is repeated until Val/Test none are full. The Val/Test topology sets were constructed by randomly drawing from the remaining topology clusters with more than one homology cluster, and assigning half of the homology clusters to train and half to val/test. Similarly, the Val/Test homology and Val/Test 70% are constructed similarly by drawing homology (resp. 70%) clusters with more than one 70% (resp. 90%) sequence identity cluster, and allocating each 70% (resp. 90%) cluster to either train or val/test. Finally, the remaining 90% clusters are randomly allocated to fill the training, validation and test sets (64/16/20% split).

For the BCE, the dataset was subdivided into fivefold for cross-validation. Antigens were clustered at 70% sequence identity, and each cluster was assigned to one fold at random (except for SARS-CoV-2 antigens, which were all assigned to fold 1).

For label computation, an amino acid of a protein chain is labeled as a binding site if at least one of its heavy atoms is within 4 Å of another heavy atom from another chain within the biological assembly[12]. Next, since the same protein may appear in multiple assemblies, we take the union of all its binding sites found across PDB files. This is done by clustering sequences at 95% sequence identity using CD-HIT[86,87], aligning the sequences and labels of each cluster using MAFFT[88]

and propagating the labels along each column. We found that for the PPBS dataset, 91.2% of the binding sites were identified from the original PDB complex file and 8.8% were propagated from other PDB files.

For SabDab, we found that PDB epitopes appeared as accessible in one conformation of the protein and buried in another conformation; labels were propagated from one structure to another only if the residues had similar relative accessible surface area and coordination number (number of amino acids within 13 Å). The propagation criterion is written as:

$$|\mathrm{ASA}_1 - \mathrm{ASA}_2|/\sigma(\mathrm{ASA}) + |\mathrm{Coord}_1 - \mathrm{Coord}_2|/\sigma(\mathrm{Coord}) < 0.5 \qquad (12)$$

For the PPBS, we obtained 22.7% positive labels and 30% when only considering the surface residues, with relative accessible surface ≥25% (distributions shown in Extended Data Fig. 3e,f). For the BCE, we found 8.9% positive labels.

For sample weighting and subsampling, PDB covers unevenly the protein sequence space: many protein families do not have any representative structure, whereas others such as immunoglobulins have tens of thousands. The sampling is also biased within one family, as some genes and/or organisms are more frequently studied than others. To correct for the biases occurring at multiple scales, we apply the following hierarchical reweighting scheme:

$$w = \frac{1}{\text{no. } \mathcal{C}_{100}} \times \frac{1}{\text{no. } \{\mathcal{C}_{100} \in \mathcal{C}_{95}\}} \times \frac{1}{\text{no. } \{\mathcal{C}_{95} \in \mathcal{C}_{90}\}} \times \frac{1}{\text{no. } \{\mathcal{C}_{90} \in \mathcal{C}_{70}\}} \qquad (13)$$

where $\mathcal{C}_T$ denotes the clusters at sequence identity cutoff, T. This choice is such that each cluster at 70% sequence identity contributes a total weight 1; within each 70% cluster, each of the K 90% clusters contributes a total weight $1/K$, and so on. An example of set of weights is illustrated in Extended Data Fig. 3 h.

In addition, this hierarchical choice ensures that the total weight of a cluster is invariant on subsampling at some higher cluster identity level (for example, the total weight of a 90% sequence identity cluster is invariant on subsampling at 100, 95 or 90% sequence identity). For the PPBS dataset, when hierarchical reweighting was used, we found no significant change of performance when training on the full set of chains or on 95% sequence identity representatives and therefore used the 95% sequence identity subset for speed gain purposes. When no reweighting or subsampling was used, performance significantly degraded (Table 1 and Extended Data Fig. 4). For the BCE database, the same approach was followed, without any subsampling—to include as many conformations as possible—and using a 90% sequence identity cutoff for the reweighting scheme, as similar proteins may have different epitopes.

Impact of induced fit on ScanNet predictions. Protein structures undergo induced fit (that is, conformational changes) on binding. The magnitude of conformational changes varies, ranging from minimal rearrangement of sidechain rotamers to extensive allosteric motion. ScanNet is mostly trained on bound chains but applied to unbound ones. Note, however, that for the PPBS dataset, 8.9% of the binding site residues are actually in unbound conformation, as their label was inferred from another PDB file (Data preparation). Owing to its high expressivity, it is a priori capable of picking up signature of bound conformations such as over-stretched sidechains or unpacked helixes (see, for example, 4wwx:B in Supplementary Fig. 7).

We evaluated the predictive performance of ScanNet on unbound chains for two datasets: the Dockground simulated and Dockground X-Ray[45] (available from http://dockground.compbio.ku.edu/). The Dockground simulated dataset consists of chains extracted from complex PDB files and relaxed using Langevin dynamics simulations[89]. Simulating the bound protein structures separately, without the interacting partner for a short time period (1 ns), relaxes the sidechain conformations of the interface residues and reliably approximates the unbound form of the protein if conformational changes are small (<2 Å r.m.s.d.). We considered only the proteins that appeared in our dataset and excluded four tetramers, obtaining 6,012 chains. We used the binding site labels of the PPBS dataset as ground truth (18.5% positive labels).

The Dockground X-Ray consists of chains that are both crystallized alone and in complex with their partner. It features chains undergoing larger conformational changes than the simulated dataset one. We selected $N = 709$ (bound, unbound) pairs with at least 95% sequence identity between chains. As some complex components were multi-chains, there was no direct correspondence with our dataset labels (which included inter-domain, intra-protein binding sites); instead, we used as ground truth labels the interface residues of the complex (6.6% positive labels). The reduction in the fraction of positive labels also stems from the longer length of proteins on average (331 and 221 for X-ray and simulated, respectively).

For both datasets, we computed ScanNet predictions separately for the bound and unbound structures, excluded residues that did not match between the bound and unbound structure and compared both predictions residue-wise. Results are reported in Supplementary Table 5 and Extended Data Fig. 6. We find a good agreement between bound and unbound predictions (Pearson correlations of $r = 0.86$, $r = 0.78$ for simulated and X-ray datasets, respectively). A slight drop in accuracy between bound and unbound structures was found: from 88.3 to 86.6% for the simulated set and from 91.9 to 91.3% for the X-ray set.

To further quantify the impact of global and local conformational changes on prediction, we calculated for each chain the r.m.s.d. between the bound and unbound atomic coordinates and the r.m.s.d between the bound and unbound solvent accessible surface area. Extended Data Fig. 6e,f shows the per-chain Pearson correlation between bound and unbound predictions against the coordinate (resp. solvent accessibility) r.m.s.d. As expected, structures with larger global/local conformational changes tend to exhibit significant changes in binding site predictions. Overall, we conclude that ScanNet predictions are overall robust to conformational changes, although improvements could be obtained by training on unbound structures.

Comparison between ScanNet and AlphaFold2 binding site predictions. AlphaFold-multimer (AF2) is a recently released model for predicting the structure of protein complexes from paired MSAs[55]. It is difficult to compare fairly AF2 and ScanNet, as the first one assumes knowledge of the partner and predicts partner-specific binding sites, whereas the second one does not assume knowledge of the partner and predicts partner-agnostic binding sites. We nonetheless benchmarked both approaches as follows. We considered Benchmark2, a set of 17 recently released dimers that do not appear in the training sets of AF2 and ScanNet[90]. For each of the 34 chains, we determined the ground truth partner-specific binding sites (that is, involved in the complex) and partner-agnostic binding site (that is, the union of all binding sites involved in any complex found among PDB structures with 95% or more sequence identity to the chain). Next, for both ScanNet and AF2, we predicted a single set of binding sites, which was compared against the two ground truths. For AF2, we predicted the structure of the complex given the pair of sequences, obtaining five models (ColabFold implementation, no relaxation[91]). For each residue, the binding site probability was defined as the fraction of models in which it belongs to the interface (taking fractional values $\in \{0, 0.2, 0.4, 0.6, 0.8, 1.0\}$; we also tested continuous values using the predicted alignment error at contacts, but found no improvement). We assumed that the protein binding sites predicted given one known partner were representative of all the protein binding sites of the protein. Although this is not true in general, an exhaustive prediction of all complexes in which the protein is involved is not possible in practice, because not all its partners are known at inference time. Arguably for most of the UniProt proteins, not even one partner is known; this inference setup is therefore realistic and reasonably fair. For ScanNet, we predicted binding site probabilities for each chain separately (average of 11 models). The AUCPR was computed for each chain separately, and for both the partner-specific and partner-agnostic binding sites. Results are reported in Extended Data Fig. 9a–c. We found that for a partner-specific binding site, AF2 outperformed ScanNet in 27 out of 34 chains (Extended Data Fig. 9 a) whereas for partner-agnostic binding, the performances were comparable (19/34 better for AF2 and 15/34 better for ScanNet, Extended Data Fig. 8b). Generically, ScanNet outperformed AF2 when a protein had multiple binding sites, whereas AF2 outperformed ScanNet when only a single binding site was known. Other examples where ScanNet outperformed AF2 were a mammal cell surface protein (6pnq, Extended Data Fig. 8c) and a rice host-pathogen interaction (5zng) for which no paired MSA can be constructed.

For BCE prediction, we tested AF2 on the RBD of the SARS-CoV-2 spike protein as follows. We first selected six representative antibody–antigen complexes spanning all the known epitopes of the RBD, following ref. [51] (Supplementary Table 6). We then predicted their structure with AF2, obtaining $6 \times 5 = 30$ models. The BCE propensity was defined residue-wise as the fraction of all models in which the residue is bound by antibodies. We found that AF2 systematically predicted a single binding mode roughly corresponding to the RBD-C epitope (Extended Data Fig. 8d,e), whereas ScanNet correctly predicted multiple epitopes. We compared AF2 and ScanNet epitope propensity predictions with the empirical antibody hit rate calculated from 290 experimental structures of antibody-spike protein found in the PDB (Extended Data Fig. 8d), and found that the ScanNet profile better correlated (Spearman coefficients of 0.74 and 0.6, respectively). Allegedly, AF2 failure stems from (1) unavailability of a paired MSA, (2) low sensitivity with respect to the antibody sequence and (3) unimodal rather than multimodal prediction.

Link between ScanNet predictions and residue contribution to binding energy. Presumably, residues with high binding probability correspond to hotspots residues, that is, residues with high contribution to the binding free energy of the complexes[43]. To test this hypothesis, we first compared ScanNet predictions to changes in binding affinity $\Delta\Delta G$ measured after mutation of binding residues to alanine. Positive $\Delta\Delta G$ indicate important residues, and hotspots are typically defined as $\Delta\Delta G > 2\,\mathrm{kcal\,mol^{-1}}$. In the SKEMPI v.2.0 database[92], we found 2,122 mutations of binding residues to alanine, spread across 130 complexes. We calculated for each residue its binding site probability p and aggregated attention coefficient a (defined as $\sum_i a_{ij}$ where a is computed as in Extended Data Fig. 2a). The later score quantifies the importance of the residue within the neighborhood; residues with high aggregated attention drive prediction of their neighborhood. Next, we estimated the conditional average $\mathbb{E}[\Delta\Delta G|p, a]$ using a one-layer perceptron with 20 hidden units, hyperbolic tangent activation and nonnegative kernel weights to enforce monotonicity (Extended Data Fig. 8a,b). We indeed find that residues with high binding probability and large attention coefficient tend to be more important for binding.

We next performed a similar analysis using the Benchmark 5.5 dataset[93] (271 dimers, 10,444 binding sites, available from https://zlab.umassmed.edu/benchmark/) and the Rosetta REF15 all-atom energy function[94]. For each dimer, the binding energy was estimated as the difference between the energy of the complex and the sum of the energies of the unbound structures. The FastRelax protocol of PyRosetta[95] was used to remove steric clashes before computation of the energies. We similarly find that residues with high binding probability and large attention coefficient tend to contribute a lower energy (Extended Data Fig. 8c,d).

In addition, Rosetta allows to calculate the contribution of individual energy terms to the residue-wise binding energy. This raises the question of whether the types of interaction involved in binding can be predicted from the intermediate layer activities of ScanNet. We grouped the 19 energy terms into eight groups: solvation (fa_sol+lk_ball_wtd+fa_intra_sol_xover4), van der Waals (fa_atr+fa_rep), Coulomb (fa_elec), backbone–sidechain hydrogen bonds (hbond_bb_sc), sidechain–sidechain hydrogen bonds (hbond_sc), sidechain internal energy (fa_intra_rep+fa_dun+yhh_planarity), backbone internal energy (omega+p_aa_pp+rama_prepro+hbond_sr_bb+hbond_lr_bb) and others (pro_close+dslf_fa13+ref).

Next, we computed for each binding residue the vector of activities of the amino acid spatio-chemical filters. We then performed a least absolute shrinkage and selection operator regression to predict residue-wise the value of each energy term from the filter activities (optimal regularization determined by cross-validation with scikit-learn[74]). The regression and correlation coefficients are shown in Extended Data Fig. 8e. We find several hotspot filters associated with negative binding energies, such as filters 81, 17, 57, 41, 2 and 22 (the O-ring filter represented in the main text). As expected, they are also strongly correlated with binding ($r = 0.47, 0.19, 0.12, 0.31, 0.09, 0.29$, see the filter depiction in Supplementary Data 1).

Each filter displays a distinct energetic profile. For instance, filter 81 is associated with strong van der Waals binding without any cost in solvation energy; consistently, it is activated by hydrophobic residues already fully exposed in the unbound state (see filter depiction in Supplementary Data 1). The O-ring filter 22 is associated with both strong van der Waals and electrostatic energy, but at the expense of a higher solvation cost. Backbone-mediated interactions are also captured; for instance, filter 54, which corresponds to an exposed glycine/lysine tandem, is associated with strong backbone–sidechain hydrogen bonding.

Altogether, the comparative analysis with mutagenesis assays and Rosetta energy supports the claim that ScanNet learns some of the underlying physical principles of binding.

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

Data availability

All raw data resources used are publicly available. They were obtained from the following databases: Protein Data Bank (https://www.rcsb.org), UniProt (https://www.uniprot.org), Dockground (http://dockground.compbio.ku.edu/), SabDab (http://opig.stats.ox.ac.uk/webapps/newsabdab/sabdab/) and SKEMPI (https://life.bsc.es/pid/skempi2). The processed data used for training and testing the models are available from the GitHub repository referenced below.

Code availability

A GitHub repository containing source code and label files for retraining and evaluating ScanNet is available at https://github.com/jertubiana/ScanNet. The original version of the code for reproducing the results of this article is available at https://zenodo.org/record/6521889#.YnPoYS8RpbW

References

63. Cock, P. J. et al. Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25**, 1422 (2009).
64. Remmert, M., Biegert, A., Hauser, A. & Söding, J. Hhblits: lightning-fast iterative protein sequence searching by hmm-hmm alignment. *Nat. Methods* **9**, 173 (2012).
65. Mirdita, M. et al. Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res.* **45**, D170 (2017).
66. Cocco, S., Feinauer, C., Figliuzzi, M., Monasson, R. & Weigt, M. Inverse statistical physics of protein sequences: a key issues review. *Rep. Progress Phys.* **81**, 032601 (2018).
67. Posani, L. *Inference and Modeling of Biological Networks: A Statistical-Physics Approach to Neural Attractors and Protein Fitness Landscapes*. PhD thesis, Univ. Paris, sciences et lettres (2018).
68. Chen, W. et al. Deep rbfnet: point cloud feature learning using radial basis functions. Preprint at arXiv:1812.04302 (2018).
69. Qi, C. R., Su, H., Mo, K. & Guibas, L. J. Pointnet: deep learning on point sets for 3D classification and segmentation, in *Proc. IEEE Conference on Computer Vision and Pattern Recognition* 652–660 (IEEE, 2017). Venue: Honolulu, Hawai. Editors: R. Chellappa, Z. Zhang, A. Hoogs, J. Rehg, Y. Liu, Y. Wu & C. Taylor
70. Qi, C. R., Yi, L., Su, H. & Guibas, L. J. Pointnet++: deep hierarchical feature learning on point sets in a metric space, Preprint at arXiv:1706.02413 (2017).
71. Igashov, I., Pavlichenko, N. & Grudinin, S. Spherical convolutions on molecular graphs for protein model quality assessment. *Mach. Learn.: Sci. Technol.* **2**, 045005 (2021).
72. Tubiana, J., Cocco, S. & Monasson, R. Learning protein constitutive motifs from sequence data. *eLife* **8**, e39397 (2019).
73. Ioffe S. & Szegedy, C. Batch normalization: accelerating deep network training by reducing internal covariate shift, in *Proc. International Conference on Machine Learning* 448–456 (PMLR, 2015). Venue: Lille, France. Editors: F. Bach, D. Blei
74. Pedregosa, F. et al. Scikit-learn: machine learning in python. *J. Mach. Learning Res.* **12**, 2825 (2011).
75. Long, J., Shelhamer, E. & Darrell, T. Fully convolutional networks for semantic segmentation, in *Proc. IEEE conference on Computer Vision and Pattern Recognition* 3431–3440 (IEEE, 2015).
76. Kingma, D. P. & Ba, J. ADAM: a method for stochastic optimization. Preprint at arXiv:1412.6980 (2014).
77. Harris, C. R. et al. Array programming with numpy. *Nature* **585**, 357 (2020).
78. Chollet, F. *Deep Learning with Python* (Simon and Schuster, 2017).
79. Abadi, M. et al. Tensorflow: a system for large-scale machine learning, in *Proc. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16). November 2–4, 2016, Savannah, GA, USA*, 265–283 (USENIX, 2016).
80. Lam, S. K., Pitrou, A. & Seibert, S. Numba: a LLVM-based Python JIT compiler, in *Proc. Second Workshop on the LLVM Compiler Infrastructure in HPC* 1–6 (2015). Venue: Austin, TX, USA. Editor: H. Finkel.
81. Virtanen, P. et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nat. Methods* **17**, 261 (2020).
82. Song, J., Tan, H., Takemoto, K. & Akutsu, T. Hsepred: predict half-sphere exposure from protein sequences. *Bioinformatics* **24**, 1489 (2008).
83. Chakravarty, S. & Varadarajan, R. Residue depth: a novel parameter for the analysis of protein structure and stability. *Structure* **7**, 723 (1999).
84. Sanner, M. F., Olson, A. J. & Spehner, J.-C. Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers* **38**, 305 (1996).
85. Connolly, M. L. Shape complementarity at the hemoglobin $\alpha 1\beta 1$ subunit interface. *Biopolymers* **25**, 1229 (1986).
86. Fu, L., Niu, B., Zhu, Z., Wu, S. & Li, W. CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics* **28**, 3150 (2012).
87. Li, W. & Godzik, A. CD-HIT: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics* **22**, 1658 (2006).
88. Nakamura, T., Yamada, K. D., Tomii, K. & Katoh, K. Parallelization of MAFFT for large-scale multiple sequence alignments. *Bioinformatics* **34**, 2490 (2018).
89. Kirys, T. et al. Simulated unbound structures for benchmarking of protein docking in the dockground resource. *BMC Bioinformatics* **16**, 243 (2015).
90. Ghani, U. et al. Improved docking of protein models by a combination of alphafold2 and cluspro. Preprint at *bioRxiv* (2021).
91. Mirdita, M. et al. Colabfold-making protein folding accessible to all. Preprint at *bioRxiv* (2021).
92. Jankauskaitė, J., Jiménez-García, B., Dapkūnas, J., Fernández-Recio, J. & Moal, I. H. SKEMPI 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics* **35**, 462 (2019).
93. Vreven, T. et al. Updates to the integrated protein–protein interaction benchmarks: docking benchmark version 5 and affinity benchmark version 2. *J. Mol. Biol.* **427**, 3031 (2015).
94. Alford, R. F. et al. The rosetta all-atom energy function for macromolecular modeling and design. *J. Chem. Theory. Comput.* **13**, 3031 (2017).
95. Chaudhury, S., Lyskov, S. & Gray, J. J. Pyrosetta: a script-based interface for implementing molecular modeling algorithms using rosetta. *Bioinformatics* **26**, 689 (2010).

Acknowledgements

J.T. acknowledges financial support from the Edmond J. Safra Center for Bioinformatics at Tel Aviv University and from the Human Frontier Science Program (cross-disciplinary postdoctoral fellowship LT001058/2019-C). D.S.-D. was supported by grant no. ISF 1466/18, Israel ministry of Science and Technology and HUJI-CIDR. This work was supported by L. Blavatnik and the Blavatnik Family Foundation. The funders had no role in study design, data collection and analysis, decision to publish or preparation of the manuscript. We are grateful to S. Lichtenzveig Sela and the CS system team for their technical support. We thank R. Groscot for his help on pythreejs visualizations. We thank L. Naccache, M. Nissan, Y. Lotem, M. Rozanov, L. Bitton, M. Halfon and S. Cohen for helpful discussions.

Author contributions

Conceptualization was carried out by J.T., D.S.-D. and H.J.W. The methodology, software, investigation, data curation and visualization were developed by J.T. Supervision and

project administration were carried out by H.J.W. Funding was acquired by J.T. and H.J.W. Writing of the paper was done by J.T., D.S.-D. and H.J.W.

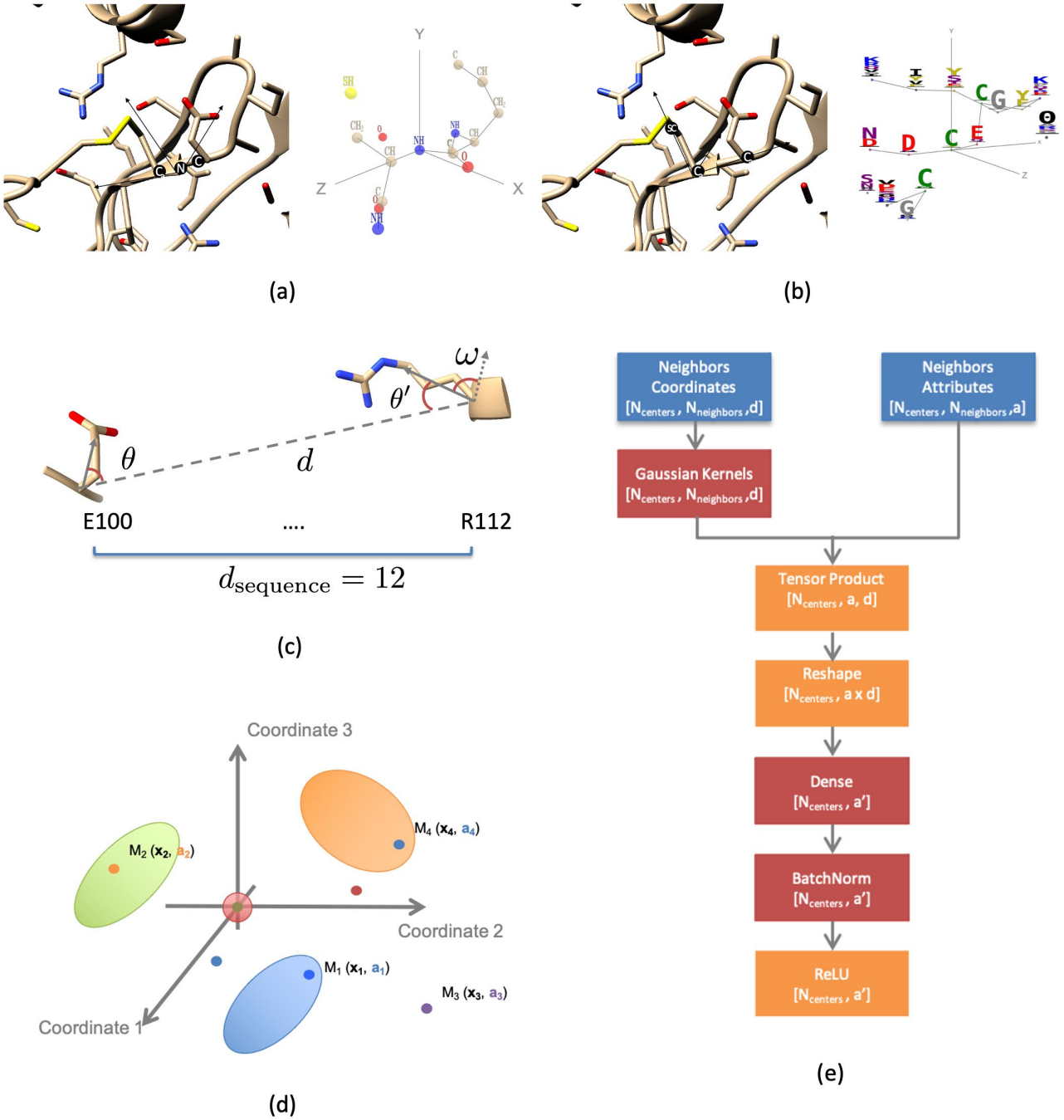
Competing interests
The authors declare no competing interests.

Additional information
Extended data are available for this paper at https://doi.org/10.1038/s41592-022-01490-7.

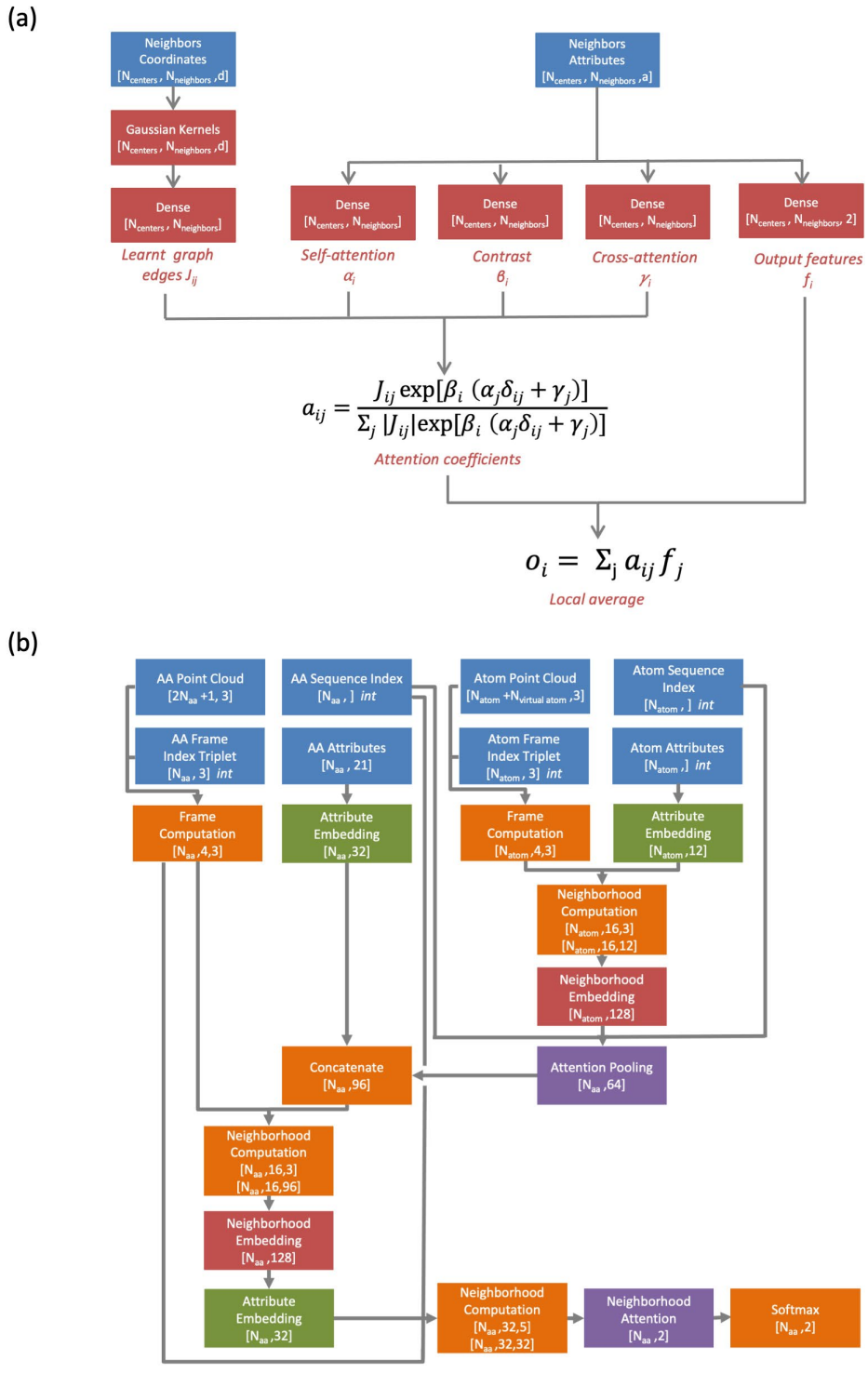
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41592-022-01490-7.

Correspondence and requests for materials should be addressed to Jérôme Tubiana, Dina Schneidman-Duhovny or Haim J. Wolfson.

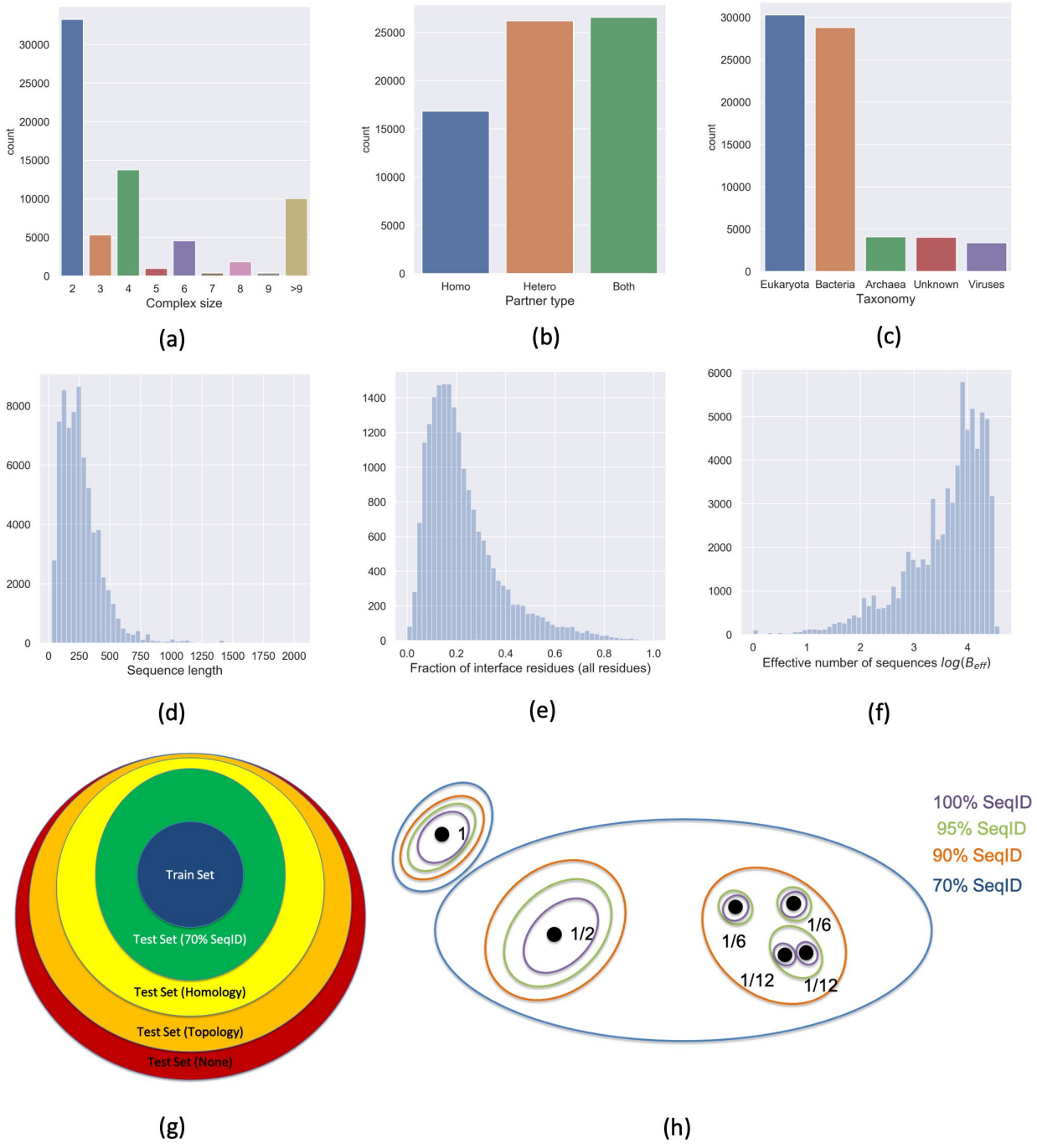
Peer review information *Nature Methods* thanks the anonymous reviewers for their contribution to the peer review of this work. Arunima Singh was the primary editor on this article and managed its editorial process and peer review in collaboration with the rest of the editorial team. Peer reviewer reports are available.



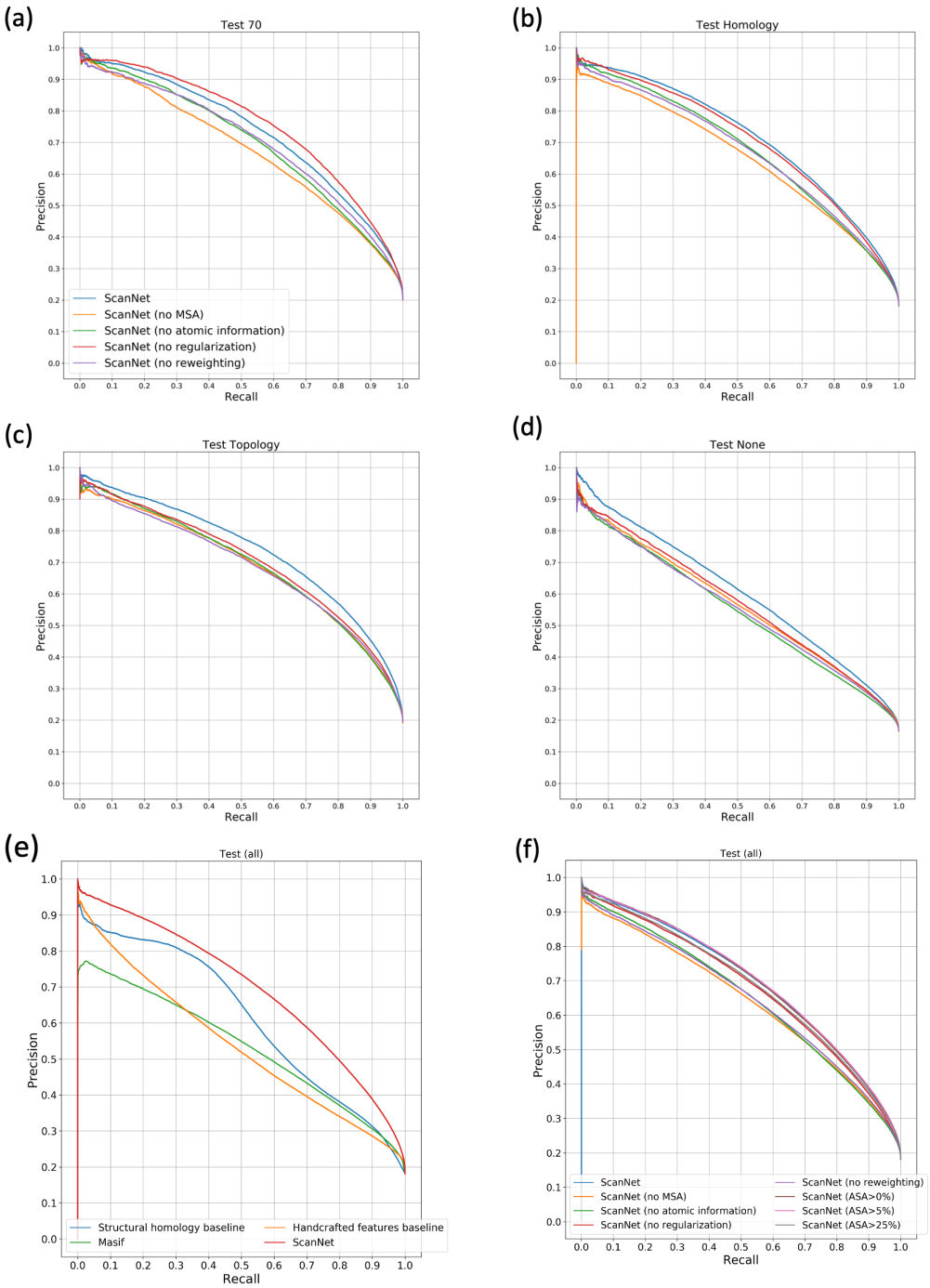
Extended Data Fig. 1 | Overview of the frame computation, neighborhood computation and neighborhood embedding modules. (a) Construction of an atomic neighborhood from structure. For each atom, the $K=16$ closest atoms (including itself) are identified. Next, a frame is constructed from its position and the directions of its covalent bonds. The neighboring atoms are characterized by their coordinates in the local frame and group type (12 subclasses: C,CH,CH2,CH3,CΠ (aromatic ring), O, OH, N, NH, NH2, S,SH. (b) Construction of an amino acid neighborhood from structure. For each amino acid, the $K=16$ closest amino acid (including itself) are identified. Next, a frame is constructed from its C_α atom, sidechain center of mass and the previous C_α atom along the backbone.The neighboring amino acid are represented by their coordinates in the local frame and their attributes learnt from the position weight matrix and pooled atomic filters. (c) Local coordinate system used for the neighborhood attention module (d) Principle of neighborhood embedding module: a generic neighborhood consists of a set of K points M_k characterized by their local coordinates $\mathbf{x}_k$ and attributes $\mathbf{a}_k$; (e) Implementation of the neighborhood embedding module.



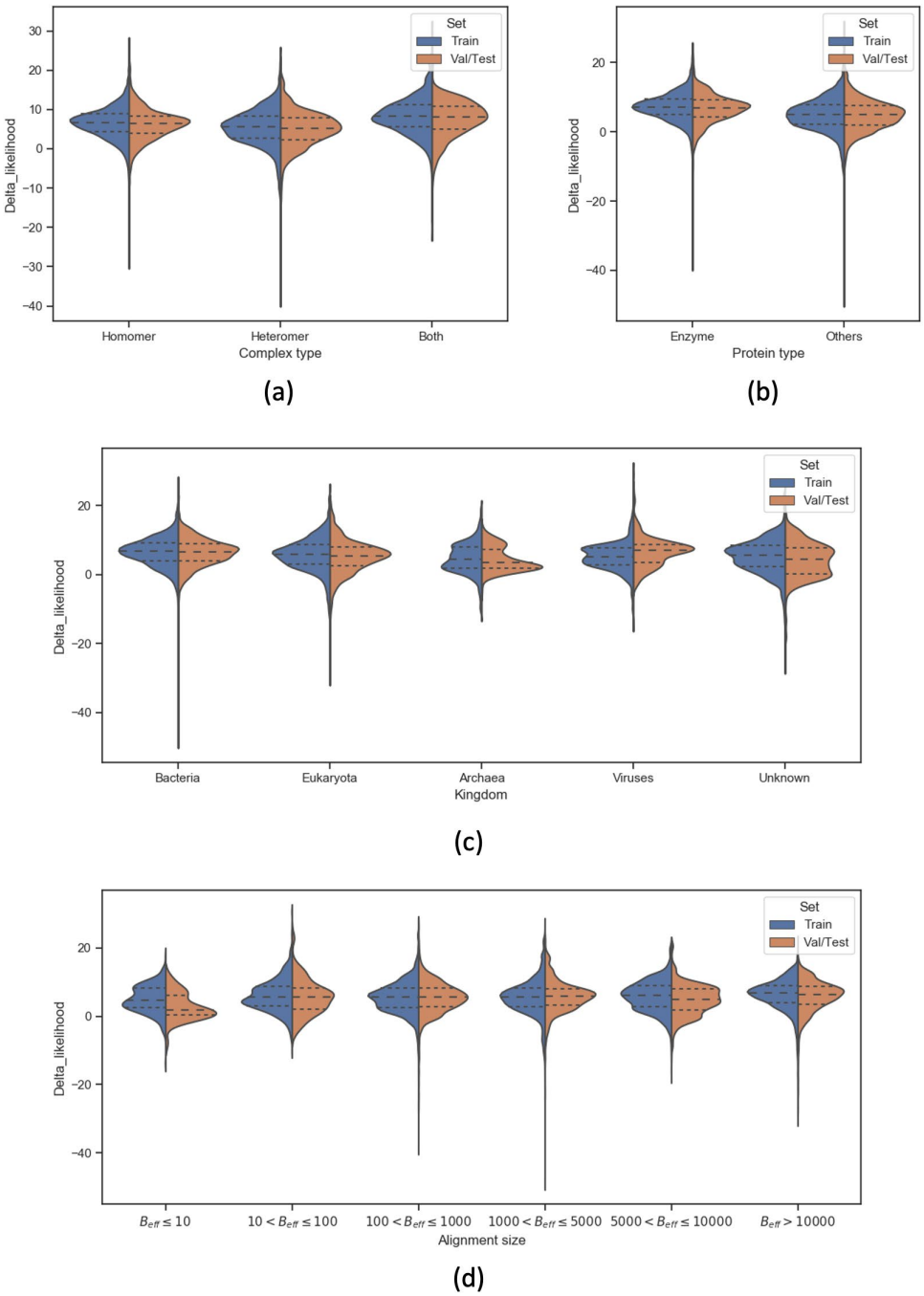
Extended Data Fig. 2 | Overview of the neighborhood attention module and the overall ScanNet architecture. (a) The Neighborhood Attention Module is the final module of ScanNet; its purpose is to locally average predictions to produce spatially consistent predictions. An attention mechanism is included to account for driver/passenger binding sites. $\delta_{ij} = 1 \text{if} i = j$; 0otherwise is the Kronecker symbol. (b) Complete architecture of ScanNet Orange modules are not trainable.



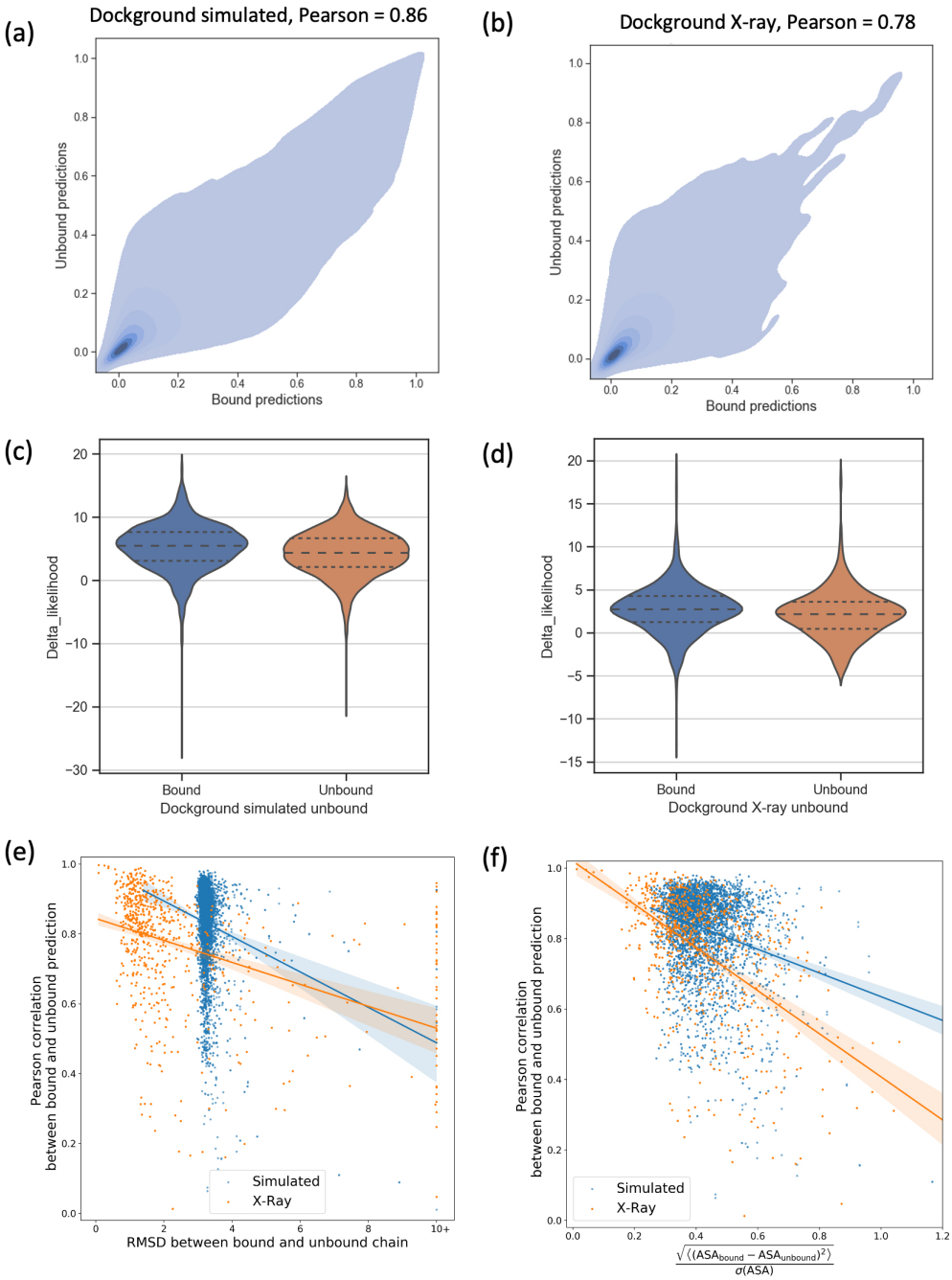
Extended Data Fig. 3 | Overview of the Protein-protein binding sites database. (a-) Distribution of (a) complex sizes (b) complex types (c) source organism taxonomy (d) protein length (e) fraction of interface residues (f) effective number of sequences in corresponding the multiple sequence alignment. (g) Data partition. Proteins of the validation/test set are subdivided into four non-overlapping groups, depending on the degree of similarity with the closest protein found in the train set: (i) ≥ 70% Sequence identity (ii) Same CATH superfamily (iii) Same fold topology CAT. (iv) None of the above. Generalization is increasingly difficult. (h) Illustration of the hierarchical sample reweighting used to counterbalance heterogeneity in the sampling of the protein space at multiple levels. Sequences are first clustered at four sequence identity thresholds (100%, 95%,90%,70%). Each cluster at 70% sequence identity (blue ellipses) contributes an identical total weight of 1 irrespective of its size. Within each 70% cluster, each of the 90% clusters (orange ellipses) contributes an identical total weight 1/Ncluster90, etc. The weight of a sample is: Num(sequences in cluster 100) × Num(cluster 100 in cluster 95) × Num(cluster 95 in cluster 90) × Num(cluster 90 in cluster 70). The weight of each cluster 70% is invariant upon subsampling of the dataset.



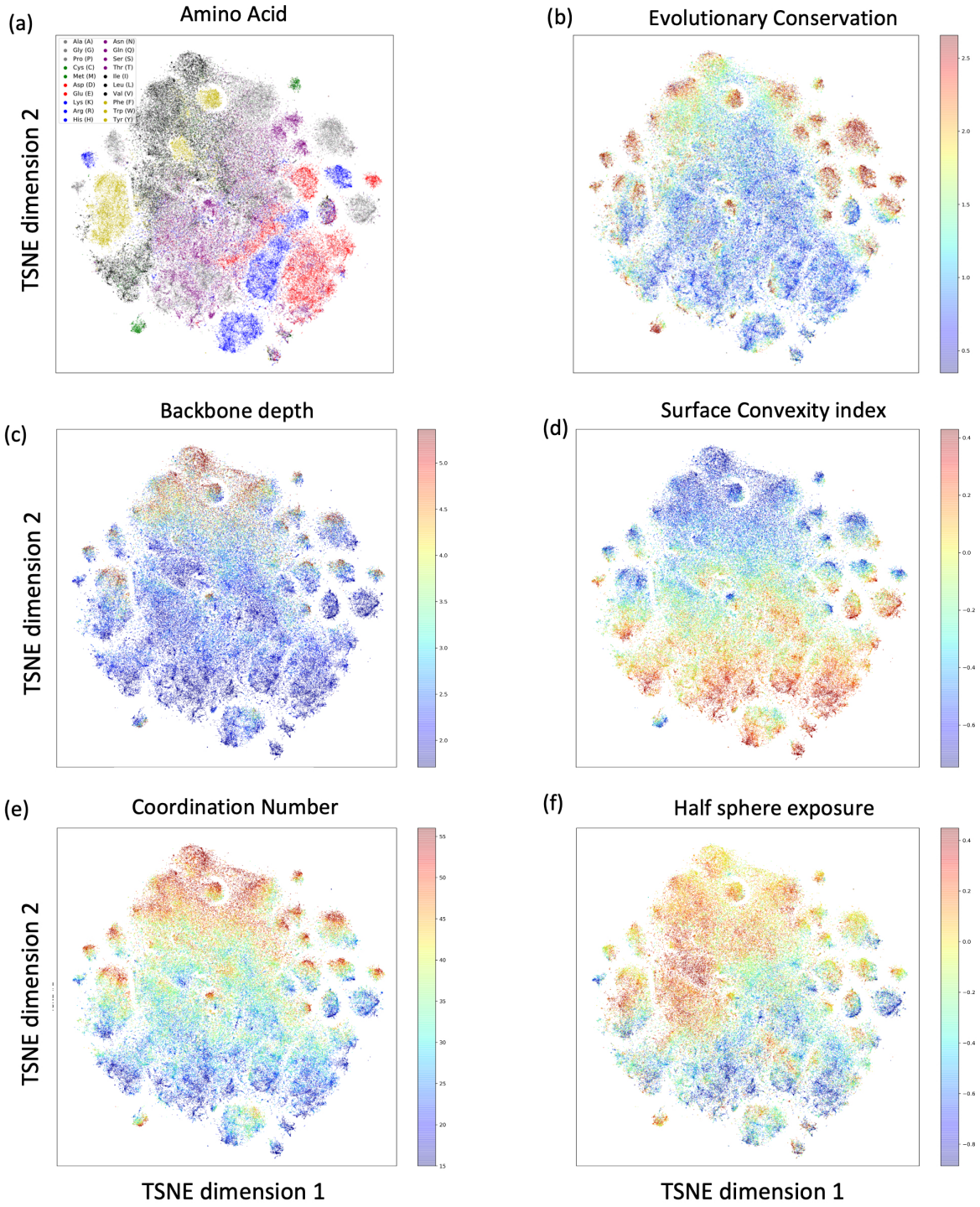
Extended Data Fig. 4 | Performance of Protein-Protein Binding Sites (PPBS) prediction. Performance of Protein-Protein Binding Sites (PPBS) prediction (a-d): Precision-Recall curves on the four test subsets (defined in Fig. 2) for various ablated ScanNet, see description of ablations in main text. (e,f): Precision-Recall curves of PPBS prediction performance, across the entire test set for ScanNet, baseline methods and ScanNet ablations.



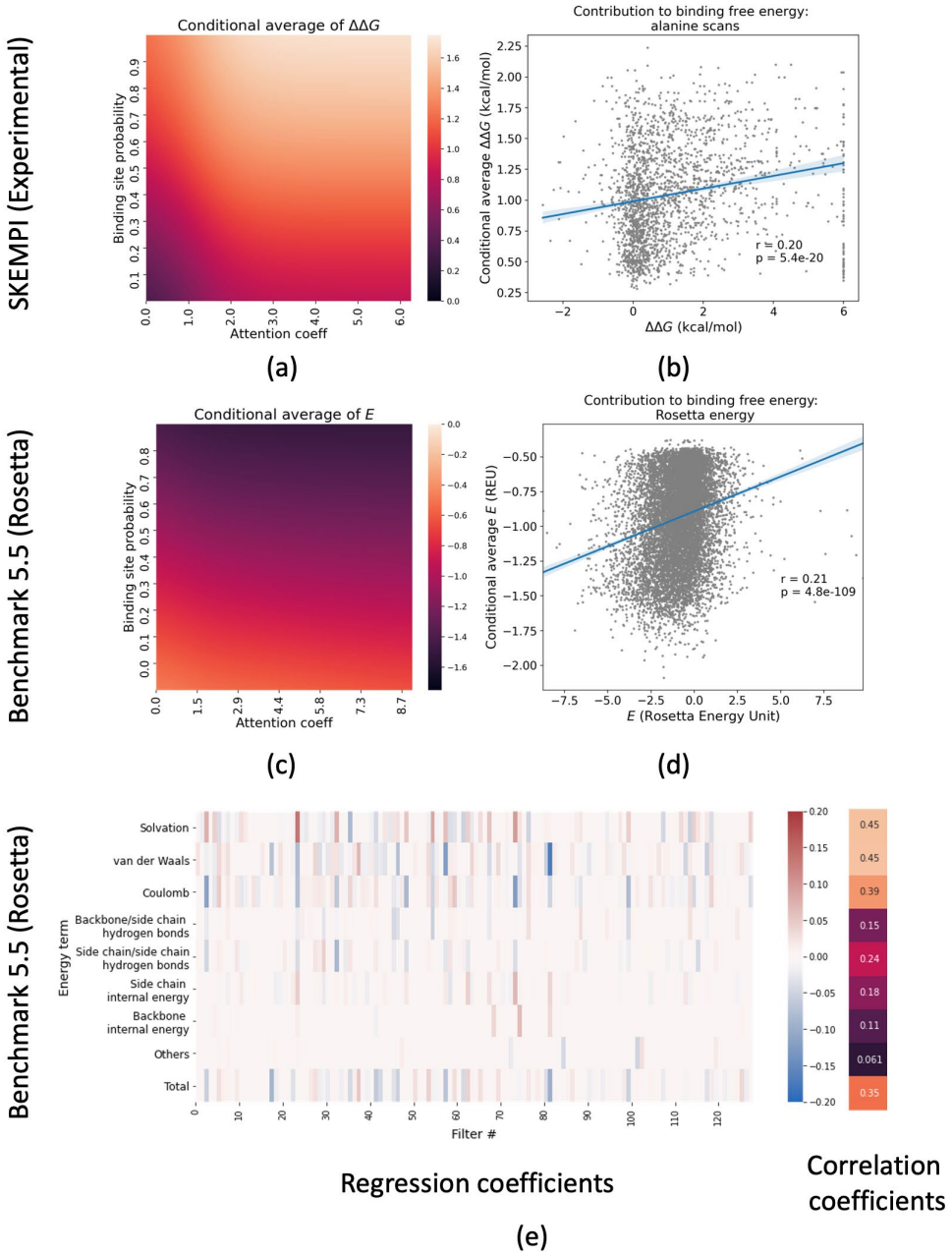
Extended Data Fig. 5 | ScanNet performance by sample type. The metric shown is the difference between the likelihood of the ScanNet and the likelihood of the null predictor (constant probability ~ 0.2); higher is better. Prediction performance is shown against complex type, protein type, source organism and effective alignment size.



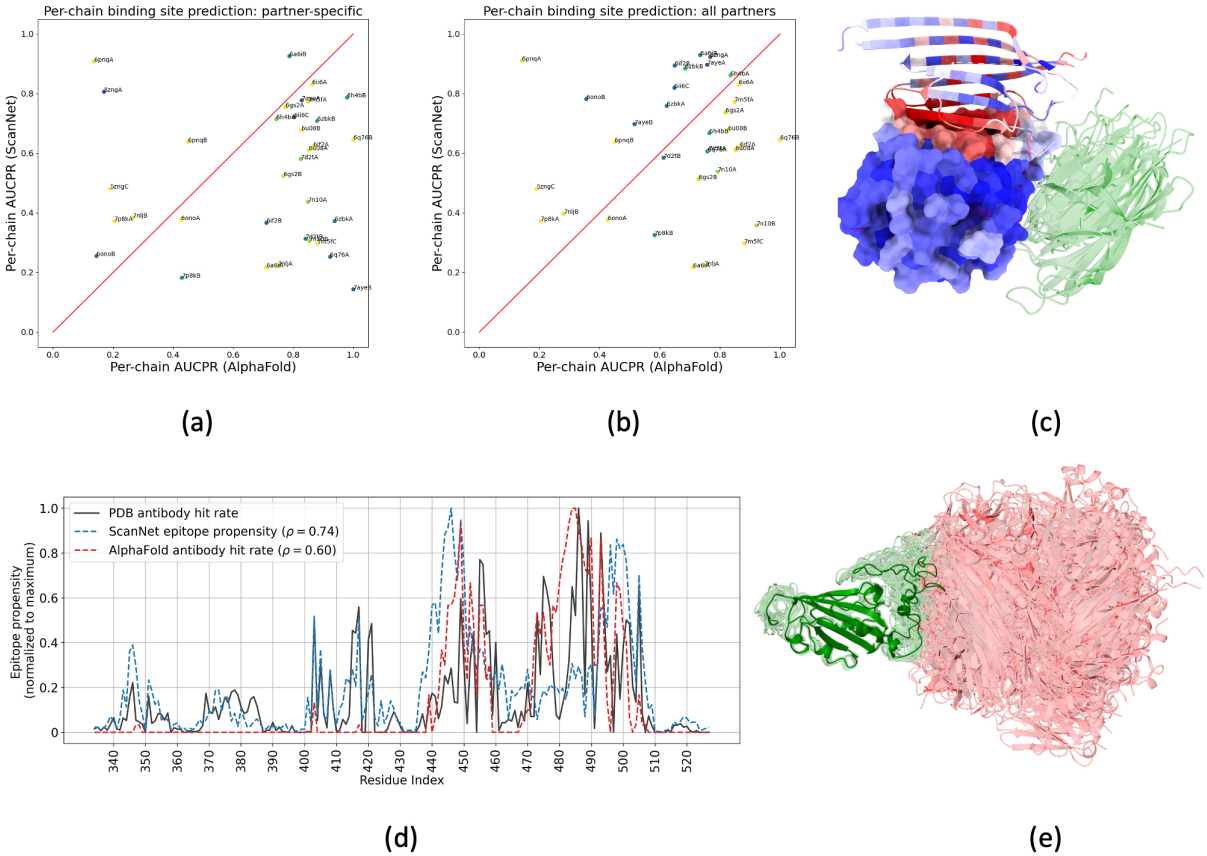
Extended Data Fig. 6 | Comparison between predictions performed on bound and unbound structures. Two data sets of (bound,unbound) pairs of protein structures are considered: the Dockground simulated dataset and Dockground X-ray data set. Panels (a),(b) display 2D-density plots of the distribution of ScanNet predictions on bound and unbound structures for each data set. Panels (c),(d) show for each data set the distributions of protein-wise prediction performance, measured as the difference $\Delta\mathcal{L}$ between the likelihood of ScanNet prediction and null prediction (uniform probability $p \sim 0.2$), divided by the standard deviation of the null model likelihood ($\sqrt{Lp(1-p)}\log\left[\frac{p}{1-p}\right]$). Higher is better. By construction, for a null predictor, $\Delta\mathcal{L}$ has zero variance across the data set whereas other metrics such as likelihood or accuracy have substantial variance owing to the variability of fraction interface residues across proteins, see Figure 3 (g); using $\Delta\mathcal{L}$ therefore facilitates detection of trends. A statistically significant but overall limited drop in performance is observed from bound to unbound. (e),(f) Impact of the degree of global (e) and local (f) conformational changes on the consistency between bound and unbound prediction. The correlation between bound and unbound predictions is represented against the RMSD between bound and unbound atomic coordinates (e), and between bound and unbound relative solvent accessibility values (f). Linear regression fit is shown, shaded area indicate 95% confidence interval determined by bootstrapping.



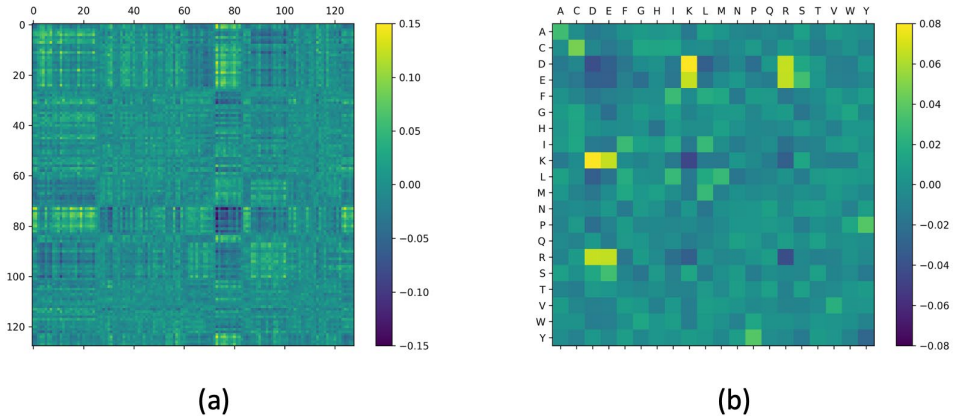
Extended Data Fig. 7 | Two-dimensional projection of the learnt amino acid scale representation using T-SNE[60] ref. [60]. Each point corresponds to one amino acid of a representative set of proteins. Coloring based on (a) Amino acid type (b) evolutionary conservation (c) backbone depth (d) surface convexity index (e) coordination number (f) Half-sphere exposure.



Extended Data Fig. 8 | Link between ScanNet predictions and residue contribution to the binding free energy. (a) Parametric fit of the conditional average of experimentally determined changes of binding affinity upon mutation to alanine $\Delta\Delta G$ (obtained from the SKEMPI v2 database), as function of the predicted binding site propensity p and aggregated attention coefficient a of the residue. (b) Scatter plot of the predicted $\Delta\Delta G$ given p, a to the experimental one (cross-validation predictions). Linear regression fit is shown, shaded area indicate 95% confidence interval determined by bootstrapping; Pearson correlation coefficient is shown along with the corresponding two-tailed p-value under normal distribution assumption. (c) Parametric fit of the conditional average of Rosetta binding energy E (computed from the Benchmark v5.5 database), as function of the predicted binding site propensity p and aggregated attention coefficient a of the residue. (d) Scatter plot of the predicted E given p, a to the experimental one (cross-validation predictions). Linear regression fit is shown, shaded area indicate 95% confidence interval determined by bootstrapping; Pearson correlation coefficient is shown along with the corresponding two-tailed p-value under normal distribution assumption (e) Sparse regression and correlation coefficients of Rosetta energy terms from ScanNet amino acid filter activities. Displayed values are $\frac{\alpha_{ij}\sigma(F_i)}{\sigma(E_j)}$, where Y_j is the j'th energy term, and F_i is the i'th filter activity and α_{ij} is the regression coefficient determined by LASSO regression.



Extended Data Fig. 9 | Comparison of AlphaFold-Multimer and ScanNet for binding site prediction. (a) Scatter plot depiction of the per-chain AUCPR metric for prediction of *partner-specific* binding sites on the Benchmark2 data set. (b) Same for *partner-agnostic* binding sites, determined by taking the union of all binding sites found in related complexes. In both panels, each chain is colored by the ratio of the number of partner-specific binding sites divided by the number of partner-agnostic binding sites (from 0=Blue to 1=Yellow). By definition, the ratio equals one for proteins with only one known partner and is low for multivalent proteins. (c) Depiction of complex 6pnq (chains A and B respectively in surface and cartoon representations), colored by ScanNet binding site probability (from blue=low to red=high). The five models produced by AF2 for chain B are superimposed in green. ScanNet correctly predicts the binding sites of both chains, but not AF2. (d) B-cell epitope propensity profile for the Receptor Binding Domain of the Spike Protein, as estimated from i) available structures in the Protein Data Bank ii) ScanNet B-cell epitope network and iii) AF2-based docking with representative antibodies. AF2 fails to identify all the main epitopes. (e) Depiction of the 30 RBD-antibody complex models predicted by AF2, featuring only a single binding mode.



Extended Data Fig. 10 | Correlation of ScanNet filter activities and amino acid types between interacting binding sites. For each of the 271 dimers of the benchmark 5.5 dataset[93] ref. [93], the pairs of interacting binding sites (defined as < 4 Åbetween any two heavy atoms) are identified. Next, the amino acid ScanNet filter activities are computed for each binding site (128-dimensional vector for each residue), and the cross-correlation is subsequently computed (Panel a). The cross-correlation matrix features significantly large entries ($|r| \sim 0.15$), suggesting that favorable interactions require complementary spatio-chemical patterns. As a control, the cross-correlation between amino acid types is similarly computed and features known complementary amino acid pairs such as K/R and D/E (of opposite electrostatic charge), but lower correlations ($|r| \sim 0.08$).