FLEXVDW: A MACHINE LEARNING APPROACH TO ACCOUNT FOR PROTEIN FLEXIBILITY IN LIGAND DOCKING

Patricia Suriana, Joseph M. Paggi, Ron O. Dror

Department of Computer Science
Stanford University
psuriana@stanford.edu, jpaggi@stanford.edu,
rondror@cs.stanford.edu

ABSTRACT

Most widely used ligand docking methods assume a rigid protein structure. This leads to problems when the structure of the target protein deforms upon ligand binding. In particular, the ligand's true binding pose is often scored very unfavorably due to apparent clashes between ligand and protein atoms, which lead to extremely high values of the calculated van der Waals energy term. Traditionally, this problem has been addressed by explicitly searching for receptor conformations to account for the flexibility of the receptor in ligand binding. Here we present a deep learning model trained to take receptor flexibility into account implicitly when predicting van der Waals energy. We show that incorporating this machine-learned energy term into a state-of-the-art physics-based scoring function improves small molecule ligand pose prediction results in cases with substantial protein deformation, without degrading performance in cases with minimal protein deformation. This work demonstrates the feasibility of learning effects of protein flexibility on ligand binding without explicitly modeling changes in protein structure.

1 INTRODUCTION

A critical problem in rational drug discovery is prediction of the position, orientation, and conformation of a ligand (e.g., a drug candidate) when bound to a target protein—i.e., the ligand's "binding pose." Protein-ligand docking methods, which are used to predict ligand binding poses, are key tools in drug discovery and molecular modeling applications (Kitchen et al., 2004; Ferreira et al., 2015).

The most widely used protein-ligand docking techniques assume a rigid protein (i.e., the positions of all protein atoms are fixed), which is often referred to as "rigid docking" (Verdonk et al., 2003; Friesner et al., 2004; Allen et al., 2015; Forli et al., 2016). Although this assumption of a rigid protein often works, rigid docking often fails to produce a near-native ligand pose (i.e., one that is close to the experimentally determined, or native, pose) when the shape of the protein's binding pocket must change for the ligand to bind. In such cases, atoms of the ligand in its native pose typically overlap ("clash") with atoms in the protein structure used for docking (Figure 1). Atoms that overlap experience extremely strong van der Waals repulsion. Rigid ligand docking methods thus predict that such poses will be extremely unfavorable energetically and generally rank them lower than any pose without such clashes—even when the clashes could have been easily resolved by minor changes in the structure of the protein's binding pocket. Such cases occur frequently in drug discovery, particularly when one is investigating novel ligands that differ substantially from ligands present in experimentally determined protein structures.

A variety of flexible protein docking techniques attempt to solve this problem by allowing the protein's binding pocket to deform during docking (Jones et al., 1997; Lemmon & Meiler, 2012; Miller et al., 2021). This approach is very computationally intensive, however, and has sometimes proven less accurate than rigid docking (Ravindranath et al., 2015; Bender et al., 2021). Likewise, ensemble docking techniques in which each ligand is docked to multiple protein structures have met with mixed success, as selecting the protein structures and determining their relative favorability has

proven difficult (Totrov & Abagyan, 2008; Novoa et al., 2010; Amaro et al., 2018; Evangelista Falcon et al., 2019; Korb et al., 2012).

In this work, we explore an alternative approach: rigid docking with a scoring function that has been adapted, through machine learning, to implicitly account for protein flexibility. In particular, we use an end-to-end machine learning approach to design a predictor of protein-ligand van der Waals (VDW) interaction energies. Given a single protein structure, our predictor is trained to recognize which types of deformations the protein's binding pocket can easily undergo, and to distinguish those from less favorable deformations. We name our machine-learned predictor of VDW interaction energies FLEXVDW.

We show that incorporating FLEXVDW into an industry-standard docking package (Glide) improves ligand binding pose prediction results in cases where ligand binding requires significant protein deformation, without compromising performance in cases with minimal protein deformation. Our work demonstrates the feasibility of learning effects of protein flexibility on ligand binding without explicitly modeling changes in protein structure.

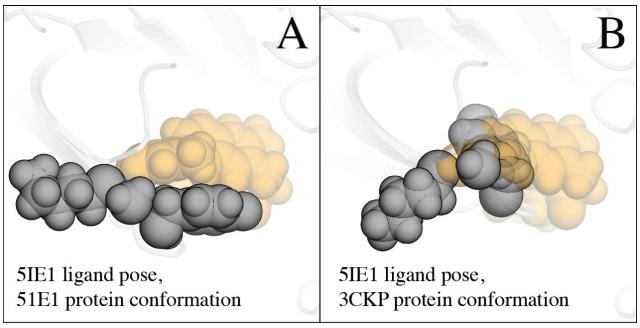

Figure 1: The native binding pose of a ligand often clashes with the experimentally determined structure of its target protein when that structure has a different ligand bound. Panel A shows the structure of the protein β-secretase (BACE-1) —- a major drug target — bound to a ligand known as "compound 5" (PDB entry 5IE1 (Jordan et al., 2016)). The ligand (orange spheres, with each sphere representing one atom) packs favorably against two amino acids in the protein binding pocket (gray spheres) without any clashes (i.e., ligand atoms do not overlap with protein atoms). Panel B shows the same ligand (compound 5) in exactly the same geometry, but superimposed on a structure of BACE-1 that was determined in the presence of a different ligand (PDB entry 3CKP, (Park et al., 2008)). Here, the same two amino acids (gray spheres) assume different positions and therefore clash (overlap) substantially with the ligand atoms (orange spheres).

2 RELATED WORKS

In general, protein-ligand docking involves two challenges (Elokely & Doerksen, 2013; Guedes et al., 2014): (1) designing a sampling algorithm to generate a large number of candidate ligand poses given a query ligand (i.e. ligand of interest) and a target protein structure, at least one of which should be close to the experimentally determined pose, and (2) designing a scoring function that ranks these candidate poses to select the best ones (i.e., those predicted to be closest to the native pose) as the final output. In this paper, we focus on the scoring challenge. Protein-ligand docking scoring function can be loosely categorized into two classes: (1) physics-based scoring functions, and (2) machine-learning scoring functions.

Physics-based scoring functions (Friesner et al., 2004; Verdonk et al., 2003; Trott & Olson, 2010; Coleman et al., 2013; Allen et al., 2015) characterize the binding of protein-ligand complexes based on a set of weighted scoring terms, which correspond to different physical effects such as VDW interactions, electrostatic interactions, hydrophobic interactions, and hydrogen bonds. The weights of these terms are typically determined by fitting experimental data using linear regression. It should be emphasized that the terms were carefully engineered to capture effects known to be important in determining ligand binding energy and represent decades of work.

Machine learning (ML) scoring functions (Khamis et al., 2015) allow for a more general functional form. Progress has been made in these areas, including end-to-end learning without hand-crafted features using deep learning methods (Shen et al., 2020; Ragoza et al., 2017; Morrone et al., 2020; McNutt et al., 2021). Nevertheless, physics-based scoring functions such as Glide (Friesner et al., 2004) or DOCK (Coleman et al., 2013; Allen et al., 2015) have proven to be more generalizable to different drug target families, and especially to new drug targets not present in the training set, than ML-based functions and remain most widely used in drug discovery (Bender et al., 2021).

3 METHODS

3.1 INCORPORATING IMPLICIT PROTEIN FLEXIBILITY INTO THE SCORING FUNCTION FOR LIGAND DOCKING

Our goal in this work is to demonstrate the feasibility of creating a VDW interaction energy predictor that implicitly accounts for protein flexibility. We therefore develop a neural network, FLEXVDW, that predicts VDW interaction energy. To demonstrate its effectiveness, we integrate FLEXVDW into Glide (Friesner et al., 2004), which is among the most widely used protein-ligand docking packages in the pharmaceutical industry. In particular, we replace the VDW term in Glide's physics-based scoring function with FLEXVDW. Although in this work we chose Glide to incorporate our machine-learned VDW term, in principle our approach can be integrated with any existing physics-based scoring function, not limited to Glide, with refitting to the particular physics-based scoring function of interest.

When training our neural network (but not when using it to predict ligand docking poses), we take advantage of the fact that, for certain proteins, multiple experimentally determined structures are available, with a different ligand bound to the same protein in each structure. Adopting terminology from structural biology, We refer to each of these ligand-bound structures as a "holo" structure. The set of holo structures for a given protein captures multiple shapes the protein's binding pocket can adopt and thus provides information about the binding pocket's flexibility.

More concretely, the input to our ML model is a single protein structure to be used for docking a ligand, where the protein structure was determined in the presence of a different ligand, or with no ligand present at all. Our training labels, on the other hand, are generated by taking into account all available holo structures (see Figure S2). Importantly, our model can be used to predicting ligand binding to proteins different from those used in training, including proteins for which only a single structure is available. Indeed, when evaluating the performance of our model (Section 4.1), we consider only proteins that were not used in training. For many of these proteins, only a single structure is available.

To assign a label to each training input, we first use Glide to calculate the VDW score (i.e., VDW interaction energy) for the candidate pose superimposed on each available holo structure for the given protein. We then determine the minimum value across these scores — that is, the most favorable score. We use this minimum value as the label (see Figure S2).

Formally, we define the minimum VDW score as

$$VDW'(L) = \min_{\forall p_i \in \{p_1, \ldots, p_N\}} VDW(L, p_i) \tag{1}$$

where $VDW(L, p_i)$ is the Glide VDW score of a candidate ligand pose L with respect to a target protein structure p_i.

When testing our model — and when deploying it for drug discovery and biology applications — we are given only a single structure of the target protein. Because of how the model is trained (on different proteins), however, it effectively predicts what the most favorable VDW score of that pose would be if multiple structures of the target protein were available. In other words, our model implicitly predicts flexibility of a protein's binding pocket given only a single structure of the protein.

3.2 DATASETS

Our training, validation, and test datasets consist of sets of poses of ligands docked to protein structures. The protein structures and small molecule ligands used to generate our ligand pose

datasets were obtained from the PDBBind 2019 refined dataset (Liu et al., 2015), a collection of protein-ligand complex structures with high resolution. The protein-ligand complex structures are categorized based on the protein (i.e., holo structures of a target protein are grouped together), and those proteins that have at least two holo structures are selected. See Figure S3 for distribution of the number of holo structures per unique protein used to generate the labels and ligand docking poses in the training and validation sets. In addition, we also included the benchmark set from Paggi et al. (2021) in our test dataset to ensure good coverage of major drug target protein families: GPCRs, kinases, ion channels and nuclear receptors (Santos et al., 2017). To ensure no data leakage, we split the proteins for training, validation, and testing such that no protein in the test dataset had more than 30% sequence identity with any protein in training or validation datasets. There are 228, 85 and 73 unique proteins in the training, validation and test datasets, respectively.

Next, candidate ligand poses for training and validation are generated using Glide SP (Friesner et al., 2004) with default parameters and then overlaid with a randomly selected holo structure of the same protein to generate poses with and without clashes with the receptor. For each query ligand, a maximum of five protein structures were randomly selected for docking, and 25 poses were randomly selected from each docking result. We follow the procedures described in Paggi et al. (2021) for preparing protein-ligand complex structures and ligands for docking.

Unlike in training/validation, in testing we are given only a single structure of the protein target on which to dock the query ligand. We can only use this one protein structure to generate candidate binding poses for the ligand. Therefore, in addition to (1) generating poses with Glide SP and normal (default) VDW parameters (VDW radius scaling of 1.0/0.8 for receptor/ligand), we ran (2) Glide SP with softened VDW parameters (VDW radius scaling of 0.6/0.5 for receptor/ligand) with extended sampling to generate candidate ligand poses with collisions with the target protein. For each scheme, we set the maximum number of candidate poses to 300 for each protein-ligand pair (referred to here as a "cross-docking pair"). Additionally, we also included a native pose of each query ligand in the candidate pose set, refined with an energy minimization protocol, since otherwise only about 80% of the cross-docking pairs have any near-native poses among the set of candidate poses generated by the two schemes above (see Figure S4). On average, we generate roughly 500 candidate ligand binding poses in total for each cross-docking pair.

For each protein-ligand pair in the test set, we randomly select one protein structure for docking. We ensure that this structure was determined experimentally in the presence of ligand substantially different from the (docked) query ligand — in particular, that the two ligands have a Tanimoto coefficient of less than 0.4, where the Tanimoto coefficient is computed by comparing the Extended-Connectivity Fingerprints (ECFPs) of the two ligands. This results in 615 cross-docking pairs, which are further divided into two cases: (1) "difficult" cross-docking pairs, defined as those for which the native ligand binding pose, after energy minimization in the docking structure, still exhibits severe clashes with protein atoms (specifically, when the ratio of the distance between two atoms and the sum of their VDW radii is ≤ 0.75), or where the ligand pose drifts significantly during energy minimization such that it exhibits an RMSD > 2.0Å relative to the original (experimentally determined) ligand pose; (2) "other" cross-docking pairs, defined as the remaining ones. In the "difficult" cases, we expect significant deformation of the protein upon ligand binding, while we expect less protein deformation in the "other" cases.

3.3 ARCHITECTURE

The input to our ML model is a candidate pose for a ligand and a single protein structure to be used for docking (see Figure S2). We also provide our model with the corresponding Glide VDW score. Although we utilize all available holo structures of a target protein to create our training labels (i.e., to calculate VDW', as described in equation 1), we do not use these other structures in any way to make the prediction. This reflects the situation in practice, where often only one structure is available to dock the ligand of interest.

Our architecture has two main components: (1) the embedding unit (see Figure 2: green block) and (2) the pairwise unit (see Figure 2: blue block). The embedding unit learns an embedding of a protein-ligand pose structure, which is then passed to the pairwise unit. At the core of the embedding unit are 3D equivariant convolution layers (ENN Layers 1 and 2) that operate on a 3D atomic point cloud. This point representation in 3D space allows us to accurately represent the relative

positioning of atoms in the protein-ligand complex, which is important for capturing the interactions between protein atoms and ligand atoms. Each ENN layer consists of the sequential application of self-interaction, point convolution, point normalization, self-interaction, and nonlinearity (Eismann et al., 2020). Each atom/point in 3D is associated with a feature vector. At input, the model takes as features the basic element type of the atom (C, O, N, P, S, polar H, or halogen (F/Cl/Br)) encoded as a one-hot vector, the secondary structure (if applicable), the partial charge of each atom, and a Boolean flag indicating whether the atom belongs to the ligand or the protein. The point-wise feature vectors are updated through the ENN layers by aggregating local information of the nearest 50 neighboring points.

To regularize our networks, we downsample the protein from all atoms to the α carbon (CA) of each amino acid residue in the last ENN layer (ENN Layer 2) of the embedding unit and apply the same learned function to each protein CA–ligand atom pair (i.e., the pairwise unit) to mimic the pairwise form of physical VDW interactions. More concretely, for each protein CA–ligand atom pair, their embeddings from the previous embedding unit are concatenated as input to the pairwise unit, a series of dense neural network layers, to compute their pairwise "interaction" features. These pairwise interaction features are averaged over all pairs (see Figure 2: Mean Pooling) and passed through the final dense neural network layer (see Figure 2: Final Dense Layer) to obtain a single scalar prediction. Inspired by Wang et al. (2019); Husic et al. (2020), which use a prior energy for learning molecular dynamics force fields, we use additional information from the Glide VDW score and pass it as input to the `min()` function in the last layer along with the output from the Final Dense Layer in order to make the final prediction.

For details on the architecture and the hyperparameters used for each component of the architecture, see Supplement S1 and Figure S1

3.4 Training

We formulate the training as a regression task aimed at predicting VDW', the minimum of the candidate ligand's VDW score over several available holo structures of the protein. The MSE loss between the actual and predicted values of VDW' is used as a loss function. To prevent loss explosion during training, the training label is capped at 100; otherwise, it could occasionally be on the order of a million or more. We train with the Adam optimizer in PyTorch (Paszke et al. (2019)) with a learning rate of 0.00005 and a batch size of 4 for 10 epochs and monitor the loss on the validation set at every epoch. In the first 5 epochs, the input Glide VDW score is ignored, in order to prevent the model from overfitting to the Glide VDW score instead of learning about protein flexibility. In the next 5 epochs, the Glide VDW score is added. The weights of the network that performs best on the validation set are then used to evaluate the predictions on the test set. We train the models on one NVIDIA GeForce RTX 3090 GPU for around 20 hours.

4 Results

4.1 Evaluation of cross-docking results on test set

To evaluate the strength of our machine-learned scoring function, FlexVDW, in terms of docking accuracy, we evaluate the top-N near-native hit rate, which is defined as the fraction of cross-docking cases for which a near-native pose is included in the first N poses when the poses are ranked by the docking score. Here, we consider a pose to be near native if its root mean square deviation (RMSD) from the experimentally determined pose is less than or equal to 2.0Å (a threshold commonly used in practice (Kontoyianni et al., 2004; Cole et al., 2005)).

The evaluation is performed on the candidate ligand poses generated for the 615 cross-docking pairs in the test dataset (see Section 3.2). During testing, only a single protein structure is provided to our ML model. We compare performance of the Glide scoring function with its original VDW term and with that term replaced by FlexVDW. As can be seen in Figure 3, incorporation of FlexVDW into Glide improves performance in "difficult" cross-docking cases (middle panel), where significant deformation of the protein is typically required upon ligand binding. At the same time, FlexVDW achieves performance similar to that of Glide's original VDW term for the "other" cross-docking cases where less protein deformation is typically required (right panel).

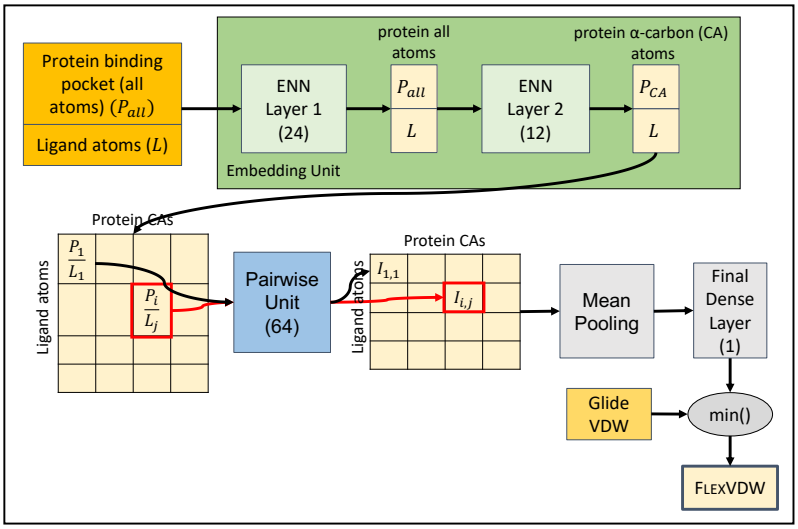

Figure 2: Schematic of the architecture of FLEXVDW network. The output dimensions of the individual layer are indicated in parentheses. At input, the model takes in a single protein structure and a single candidate ligand pose (orange block) to predict the VDW interaction energy of the ligand pose with respect to the protein structure. The model featurizes the input into basic element type of the atom (C, O, N, P, S, polar H, and F/Cl/Br), secondary structure (if applicable), partial charge and a protein/ligand Boolean flag for each atom. FLEXVDW consists of two main components: (1) embedding unit (green block) and (2) pairwise unit (blue block). At the core of the embedding unit are 3D equivariant convolution layers (ENN Layers 1 and 2; light green blocks) that operate on the atomic point cloud to learn the embedding of protein/ligand atoms, which is then used to predict the VDW score of the ligand pose. To regularize our networks, we downsample the protein from all atoms (P_{all}) to α-carbons (P_{CA}) at the last layer of the embedding unit (ENN Layer 2), and apply the same learned function to each protein CA–ligand atom pair (i.e., the pairwise unit) to mimic the pairwise form of physical VDW interactions. In addition, we calculate the Glide VDW score and pass it as input to the `min()` function in the last layer to make the final prediction. For details on the architecture and the hyperparameters used for each component of the architecture, see Supplement S1 and Figure S1.

In addition, as a baseline, we evaluate the accuracy of a scoring function in which we simply remove the VDW term while keeping the other terms of the Glide scoring function. As we can see in Figure 3, although eliminating the VDW term leads to a better top-N near-native hit rate for "difficult" cases compared to FLEXVDW, overall performance deteriorates (especially for the top-1 near-native hit rate), which shows the importance of including a VDW term in the docking score. As we allow more ligand poses with severe collisions with protein backbones ("garbage poses") in the candidate pose set, the performance of FLEXVDW decreases, but the performance of the docking score without a VDW term decreases even more, showing that our approach is able to generalize to some extent even if we never train the model with "garbage" poses, and further highlighting the importance of the VDW term in the docking score (see Figure S5).

4.2 COMPARISON OF GLIDE AND FLEXVDW PREDICTED SCORES AND TOP-1 POSES

Next, we compare the FLEXVDW and Glide VDW scores for the native ligand poses when superimposed on structures of the target protein determined with other ligands bound. In Figure 4A-C, the native ligand poses clash with the docking structures. Glide assigns very high (unfavorable) VDW scores, preventing it from predicting these poses. Indeed, in these cases, Glide's top-ranked (top-1) ligand pose predictions differ substantially from the native pose(see Figure S6A-C). In contrast, our machine-learned predictor, FLEXVDW, handles these cases better. In two of the three cases, it ranks near-native poses first (top-1) (see Figure S6A and C). In the third case (Figure S6B), even though FLEXVDW predicts a negative VDW score for the native ligand pose (see Figure 4B), the near-native poses are eventually rejected due to the high electrostatic repulsion energy, thus FLEXVDW fails to

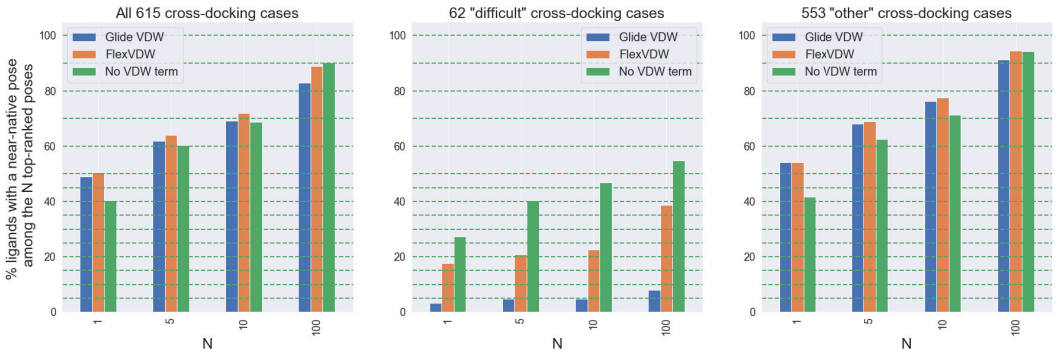

Figure 3: Percentage of cases for which a near-native pose is included in the top N poses sorted by docking score (higher is better). Our approach significantly improves over Glide performance in "difficult" cross-docking cases where significant deformation of the protein is expected upon ligand binding, while maintaining performance in "other" cross-docking cases where minimal deformation of the protein is expected. Although the absence of a VDW term in the docking score leads to better performance in "difficult" cases, it worsens overall performance (especially for the top-ranked pose), showing the importance of including a VDW term in the docking score.

select the near-native pose as the top-1 pose. When there is no clash between the ligand pose and the protein structure used for docking, FLEXVDW is comparable to Glide in ranking near-native ligand pose highly (see Figure 4D and S6D).

5 DISCUSSION

We have demonstrated the feasibility of learning a scoring function that accounts for protein flexibility in ligand docking without explicitly modeling changes in protein structure. Given a protein structure and a candidate ligand pose, FLEXVDW predicts the VDW value of this ligand pose, taking into account the flexibility of the protein.

To evaluate the strength of our machine-learned scoring function in terms of docking accuracy, we evaluate the top-N near-native hit rate of cross-docking protein-ligand pairs. To ensure the generalizability of our methods to different protein families, we select our test cases to cover the major drug target protein families, including GPCRs, kinases, ion channels, nuclear receptors, and others. We show that incorporating this machine-learned VDW term into Glide, a state-of-the-art physics-based scoring function, improves docking accuracy in cases with substantial protein deformation upon ligand binding, without degrading performance in cases with minimal protein deformation upon ligand binding. Our approach could be integrated with any existing physics-based scoring function, not limited to Glide, with refitting to the particular physics-based scoring function of interest.

There are several limitations to our approach. First, we formulate our learning task in terms of predicting the global VDW score. Reformulating the learning task in terms of predicting VDW interaction energies between individual pairs of atoms could potentially provide a better signal for which parts of the protein are flexible upon ligand binding. Additionally, we consider only VDW interactions and ignore the electrostatic interaction. In some cases, a near-native ligand pose is eliminated not only due to a high VDW energy, but also due to a high electrostatic repulsion energy. Future work is necessary to address these issues.

Second, because we assign training labels using the minimum VDW score across multiple holo structures as a proxy for protein flexibility, the extent to which our model can learn about protein flexibility is limited by the diversity of available holo structures. This could potentially be improved by including snapshots from molecular dynamics simulations as additional protein structures when determining the training labels.

Note that when using our model to predict ligand binding poses, we use only a single structure of the target protein — because often only one a single structure of a given protein is available. Indeed,

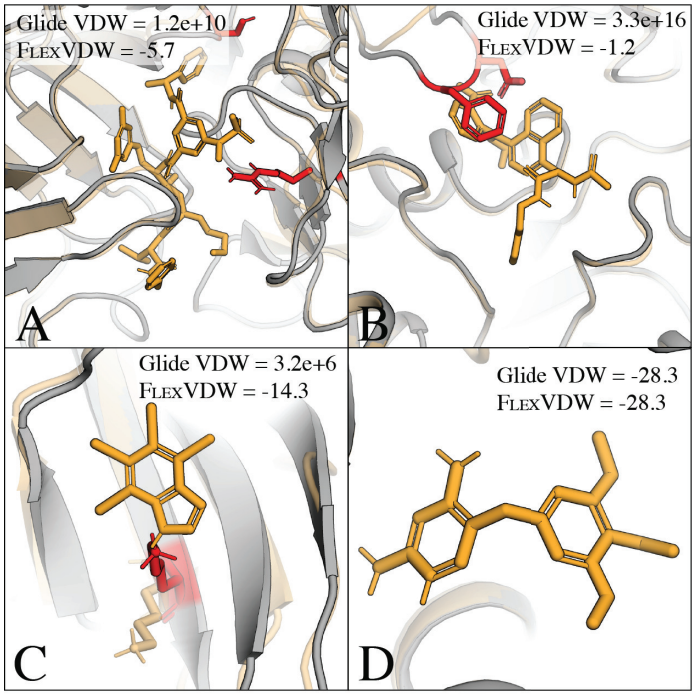

Figure 4: Comparison of the Glide VDW and FlexVDW score of the native ligand poses (orange) when docked to other holo structures of the same protein (grey). The protein side chains that clash with the native ligand poses are shown in red. (A): Native ligand pose of 3I25 docked to the protein structure of 1FKN (Uniprot ID: P56817). The native ligand pose clashes with several residue side chains. (B): Native ligand pose of 1NL9 docked to the protein structure of 2F6T (Uniprot ID: P18031). The native ligand pose clashes with the loop that should have been located further up in the crystal structure of 2F6T (transparent light orange). (C): Native ligand pose of 2OXY docked to the protein structure of 2PVJ (Uniprot ID: P28523). The native ligand pose clashes with the lysine side chain (red) in the docking structure. In the native crystal structure, the lysine side chain points downward (transparent light orange). (D): Native ligand pose of 2W3A docked to the protein structure of 1BOZ (Uniprot ID: P00374). Here, there is no clash between the native ligand pose and the protein. Both Glide and FlexVDW give the same VDW scores of -28.3.

when when evaluating the performance of our model, we use only a single structure for each protein, and the proteins used for evaluation are all substantially different from those used to train the model.

In summary, our work is a step toward incorporating implicit protein flexibility into ligand docking, which will improve the accuracy of ligand binding pose prediction.

Funding Information

PS was supported by a Graduate Research Fellowship from the US National Science Foundation (NSF). JMP was supported by a Stanford Graduate Fellowship.

References

William J Allen, Trent E Balius, Sudipto Mukherjee, Scott R Brozell, Demetri T Moustakas, P Therese Lang, David A Case, Irwin D Kuntz, and Robert C Rizzo. Dock 6: Impact of new features and current docking performance. *Journal of computational chemistry*, 36(15):1132–1156, 2015.

Rommie E Amaro, Jerome Baudry, John Chodera, Özlem Demir, J Andrew McCammon, Yinglong Miao, and Jeremy C Smith. Ensemble docking in drug discovery. *Biophysical journal*, 114(10): 2271–2278, 2018.

Brian J Bender, Stefan Gahbauer, Andreas Luttens, Jiankun Lyu, Chase M Webb, Reed M Stein, Elissa A Fink, Trent E Balius, Jens Carlsson, John J Irwin, et al. A practical guide to large-scale docking. *Nature protocols*, 16(10):4799–4832, 2021.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv: Learning*, 2016.

Jason C Cole, Christopher W Murray, J Willem M Nissink, Richard D Taylor, and Robin Taylor. Comparing protein–ligand docking programs is difficult. *Proteins: Structure, Function, and Bioinformatics*, 60(3):325–332, 2005.

Ryan G Coleman, Michael Carchia, Teague Sterling, John J Irwin, and Brian K Shoichet. Ligand pose and orientational sampling in molecular docking. *PloS one*, 8(10):e75992, 2013.

Stephan Eismann, Patricia Suriana, Bowen Jing, Raphael J. L. Townshend, and Ron O. Dror. Protein model quality assessment using rotation-equivariant, hierarchical neural networks. *arXiv preprint arXiv:2011.13557*, 2020.

Stephan Eismann, Raphael J.L. Townshend, Nathaniel Thomas, Milind Jagota, Bowen Jing, and Ron O. Dror. Hierarchical, rotation-equivariant neural networks to select structural models of protein complexes. *Proteins: Structure, Function, and Bioinformatics*, 89(5):493–501, 2021. doi: https://doi.org/10.1002/prot.26033. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/prot.26033`.

Khaled M Elokely and Robert J Doerksen. Docking challenge: protein sampling and molecular docking performance. *Journal of chemical information and modeling*, 53(8):1934–1945, 2013.

Wilfredo Evangelista Falcon, Sally R Ellingson, Jeremy C Smith, and Jerome Baudry. Ensemble docking in drug discovery: how many protein configurations from molecular dynamics simulations are needed to reproduce known ligand binding? *The Journal of Physical Chemistry B*, 123(25): 5189–5195, 2019.

Leonardo G Ferreira, Ricardo N Dos Santos, Glaucius Oliva, and Adriano D Andricopulo. Molecular docking and structure-based drug design strategies. *Molecules*, 20(7):13384–13421, 2015.

Stefano Forli, Ruth Huey, Michael E Pique, Michel F Sanner, David S Goodsell, and Arthur J Olson. Computational protein–ligand docking and virtual drug screening with the autodock suite. *Nature protocols*, 11(5):905–919, 2016.

Richard A Friesner, Jay L Banks, Robert B Murphy, Thomas A Halgren, Jasna J Klicic, Daniel T Mainz, Matthew P Repasky, Eric H Knoll, Mee Shelley, Jason K Perry, et al. Glide: a new approach for rapid, accurate docking and scoring. 1. method and assessment of docking accuracy. *Journal of medicinal chemistry*, 47(7):1739–1749, 2004.

Isabella A Guedes, Camila S de Magalhães, and Laurent E Dardenne. Receptor–ligand molecular docking. *Biophysical reviews*, 6:75–87, 2014.

Brooke E Husic, Nicholas E Charron, Dominik Lemm, Jiang Wang, Adrià Pérez, Maciej Majewski, Andreas Krämer, Yaoyi Chen, Simon Olsson, Gianni de Fabritiis, et al. Coarse graining molecular dynamics with graph neural networks. *The Journal of chemical physics*, 153(19):194101, 2020.

Gareth Jones, Peter Willett, Robert C Glen, Andrew R Leach, and Robin Taylor. Development and validation of a genetic algorithm for flexible docking. *Journal of molecular biology*, 267(3): 727–748, 1997.

John B Jordan, Douglas A Whittington, Michael D Bartberger, E Allen Sickmier, Kui Chen, Yuan Cheng, and Ted Judd. Fragment-linking approach using 19f nmr spectroscopy to obtain highly potent and selective inhibitors of β-secretase. *Journal of Medicinal Chemistry*, 59(8):3732–3749, 2016.

Mohamed A Khamis, Walid Gomaa, and Walaa F Ahmed. Machine learning in computational docking. *Artificial intelligence in medicine*, 63(3):135–152, 2015.

Douglas B Kitchen, Hélène Decornez, John R Furr, and Jürgen Bajorath. Docking and scoring in virtual screening for drug discovery: methods and applications. *Nature reviews Drug discovery*, 3 (11):935–949, 2004.

Maria Kontoyianni, Laura M McClellan, and Glenn S Sokol. Evaluation of docking performance: comparative data on docking algorithms. *Journal of medicinal chemistry*, 47(3):558–565, 2004.

Oliver Korb, Tjelvar SG Olsson, Simon J Bowden, Richard J Hall, Marcel L Verdonk, John W Liebeschuetz, and Jason C Cole. Potential and limitations of ensemble docking. *Journal of chemical information and modeling*, 52(5):1262–1274, 2012.

Gordon Lemmon and Jens Meiler. Rosetta ligand docking with flexible xml protocols. *Computational Drug Discovery and Design*, pp. 143–155, 2012.

Zhihai Liu, Yan Li, Li Han, Jie Li, Jie Liu, Zhixiong Zhao, Wei Nie, Yuchen Liu, and Renxiao Wang. PDB-wide collection of binding data: current status of the PDBbind database. *Bioinformatics*, 31 (3):405–412, February 2015.

Andrew T McNutt, Paul Francoeur, Rishal Aggarwal, Tomohide Masuda, Rocco Meli, Matthew Ragoza, Jocelyn Sunseri, and David Ryan Koes. Gnina 1.0: molecular docking with deep learning. *Journal of cheminformatics*, 13(1):1–20, 2021.

Edward B Miller, Robert B Murphy, Daniel Sindhikara, Kenneth W Borrelli, Matthew J Grisewood, Fabio Ranalli, Steven L Dixon, Steven Jerome, Nicholas A Boyles, Tyler Day, et al. Reliable and accurate solution to the induced fit docking problem for protein–ligand binding. *Journal of Chemical Theory and Computation*, 17(4):2630–2639, 2021.

Joseph A Morrone, Jeffrey K Weber, Tien Huynh, Heng Luo, and Wendy D Cornell. Combining docking pose rank and structure with deep learning improves protein–ligand binding mode prediction over a baseline docking approach. *Journal of chemical information and modeling*, 60(9): 4170–4179, 2020.

Eva Maria Novoa, Lluis Ribas de Pouplana, Xavier Barril, and Modesto Orozco. Ensemble docking from homology models. *Journal of Chemical Theory and Computation*, 6(8):2547–2557, 2010.

Joseph M Paggi, Julia A Belk, Scott A Hollingsworth, Nicolas Villanueva, Alexander S Powers, Mary J Clark, Augustine G Chemparathy, Jonathan E Tynan, Thomas K Lau, Roger K Sunahara, et al. Leveraging nonstructural data to predict structures and affinities of protein–ligand complexes. *Proceedings of the National Academy of Sciences*, 118(51):e2112621118, 2021.

Heuisul Park, Kyeongsik Min, Hyo-Shin Kwak, Ki Dong Koo, Dongchul Lim, Sang-Won Seo, Jae-Ung Choi, Bettina Platt, and Deog-Young Choi. Synthesis, sar, and x-ray structure of human bace-1 inhibitors with cyclic urea derivatives. *Bioorganic & medicinal chemistry letters*, 18(9): 2900–2904, 2008.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Matthew Ragoza, Joshua Hochuli, Elisa Idrobo, Jocelyn Sunseri, and David Ryan Koes. Protein–ligand scoring with convolutional neural networks. *Journal of chemical information and modeling*, 57(4):942–957, 2017.

Pradeep Anand Ravindranath, Stefano Forli, David S Goodsell, Arthur J Olson, and Michel F Sanner. Autodockfr: advances in protein-ligand docking with explicitly specified binding site flexibility. *PLoS computational biology*, 11(12):e1004586, 2015.

Rita Santos, Oleg Ursu, Anna Gaulton, A Patrícia Bento, Ramesh S Donadi, Cristian G Bologa, Anneli Karlsson, Bissan Al-Lazikani, Anne Hersey, Tudor I Oprea, et al. A comprehensive map of molecular drug targets. *Nature reviews Drug discovery*, 16(1):19–34, 2017.

Chao Shen, Junjie Ding, Zhe Wang, Dongsheng Cao, Xiaoqin Ding, and Tingjun Hou. From machine learning to deep learning: Advances in scoring functions for protein–ligand docking. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 10(1):e1429, 2020.

Maxim Totrov and Ruben Abagyan. Flexible ligand docking to multiple receptor conformations: a practical alternative. *Current opinion in structural biology*, 18(2):178–184, 2008.

Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010.

Marcel L Verdonk, Jason C Cole, Michael J Hartshorn, Christopher W Murray, and Richard D Taylor. Improved protein–ligand docking using gold. *Proteins: Structure, Function, and Bioinformatics*, 52(4):609–623, 2003.

Jiang Wang, Simon Olsson, Christoph Wehmeyer, Adrià Pérez, Nicholas E Charron, Gianni De Fabritiis, Frank Noé, and Cecilia Clementi. Machine learning of coarse-grained molecular dynamics force fields. *ACS central science*, 5(5):755–767, 2019.

Supplementary Information

S1 Details on the Architecture

The ENN layers of the embedding unit (see Figure S1A: green block) consists of sequential application of self-interaction, point-convolution, point normalizationm, self-interaction, and nonlinearity (Eismann et al., 2021) (see Figure S1B). We restrict the maximum filter rotation order at each ENN layer to $l = 2$. At each point convolution, it updates the features associated to a given point p based on the features of 50 closest neighboring points in 3D Euclidian space, weighted by their distances to p. We express these weighted distances (W^l) in terms of Gaussian radial basis function (RBF) kernel, as a trainable network of two dense layers with a hidden layer (followed by a ReLU nonlinearity) of size 12. The number of basis and maximum radius of the Gaussian RBF kernel determines the spatial resolution of the kernel, which we chose to be 12 and 12.0Å respectively for our architecture (see Figure S1C).

Starting from a one-hot encoding of the element type, secondary structure, partial charges (from the OPLS force field), and Boolean protein/ligand flags as feature channels at input to the embedding unit ($Z_1, \ldots Z_n$), the first ENN layer of the embedding unit mixes those features and outputs 24 feature channels per rotation order ($l = 0$, $l = 1$, and $l = 2$). The second ENN layer further mixes those features to output 12 feature channels per rotation order (see Figure S1A: green block).

Next, we constructed pairwise features of protein-CA–ligand-atom pairs by concatenating their 0-th rotation order ($l = 0$) embeddings ($E_1, \ldots E_n$), which are then passed to the pairwise unit to calculate the pairwise protein–ligand atom interaction features ($I_{1,1}, \ldots I_{p,q}$). This pairwise unit consists of a series of dense neural network layers (followed by the ELU activation function (Clevert et al., 2016) and a dropout layer). For more details on the layers (including the output feature dimensions of each layer), see Figure S1A: blue block. These pairwise interaction features, $I_{1,1}, \ldots I_{p,q}$, are averaged over all pairs (see Figure 2: Mean Pooling) and passed through the Final Dense Layer (see Figure 2), which consists of a single dense neural network layer, to obtain a single scalar prediction. The final prediction of the network is the minimum of this single scalar prediction and the Glide VDW score.

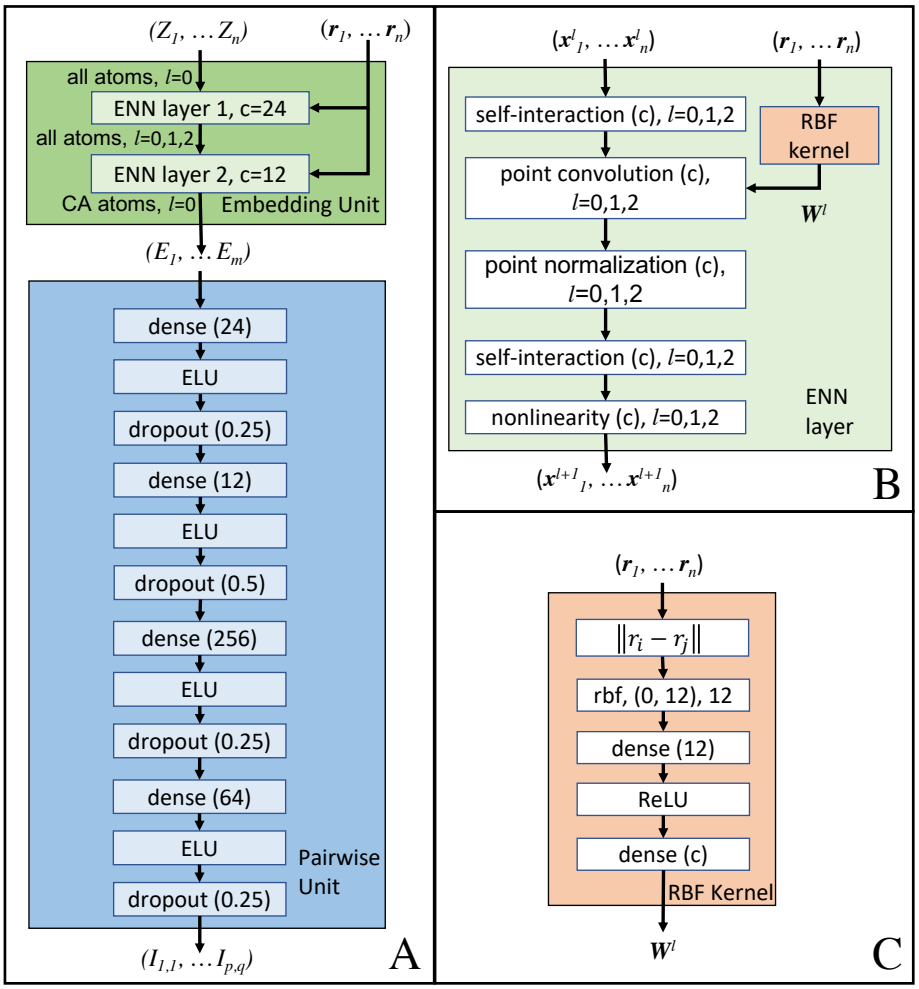

Figure S1: Detail Schematic of the architecture of FLEXVDW network. The output size of each layer and the dropout parameters are indicated in parentheses. Panel A shows the overall architecture of FLEXVDW, which consists of (1) an embedding unit (green block) and (2) a pairwise unit (blue block). We represent a protein-ligand complex as a set of points in 3D with associated scalar features ($Z1, \ldots Z_n$). The embedding unit learns an embedding feature for each point through two layers of ENN layers (panel B), each with output channels of size 24 and 12, respectively. For each protein–ligand atom pair, we concatenate their embeddings and pass it to the pairwise unit (blue block) to calculate the pairwise protein–ligand atom interaction features ($I_{1,1}, \ldots I_{p,q}$). These pairwise interaction features are then averaged over all pairs (see Figure 2: Mean Pooling) and passed through the Final Dense Layer (see Figure 2) to obtain a single scalar prediction. Panel B shows the architecture of the ENN layer. The ENN layer updates the features associated with each point with respect to the features of the 50 nearest neighbor points in 3D Euclidean space, weighted by the distances of the neighbors to the point (W^l). Panel C shows the Gaussian radial basis function (RBF) kernel, a trainable network consisting of a hidden dense layer of size 12, a ReLU nonlinearity, and a final dense layer. This kernel computes the weights based on the distances of each point for updating the point convolution (panel B). The number of basis and maximum radius of the Gaussian RBF kernel determines the spatial resolution of the kernel, which we chose to be 12 and 12.0 Å respectively.

S2 SUPPLEMENTARY FIGURES

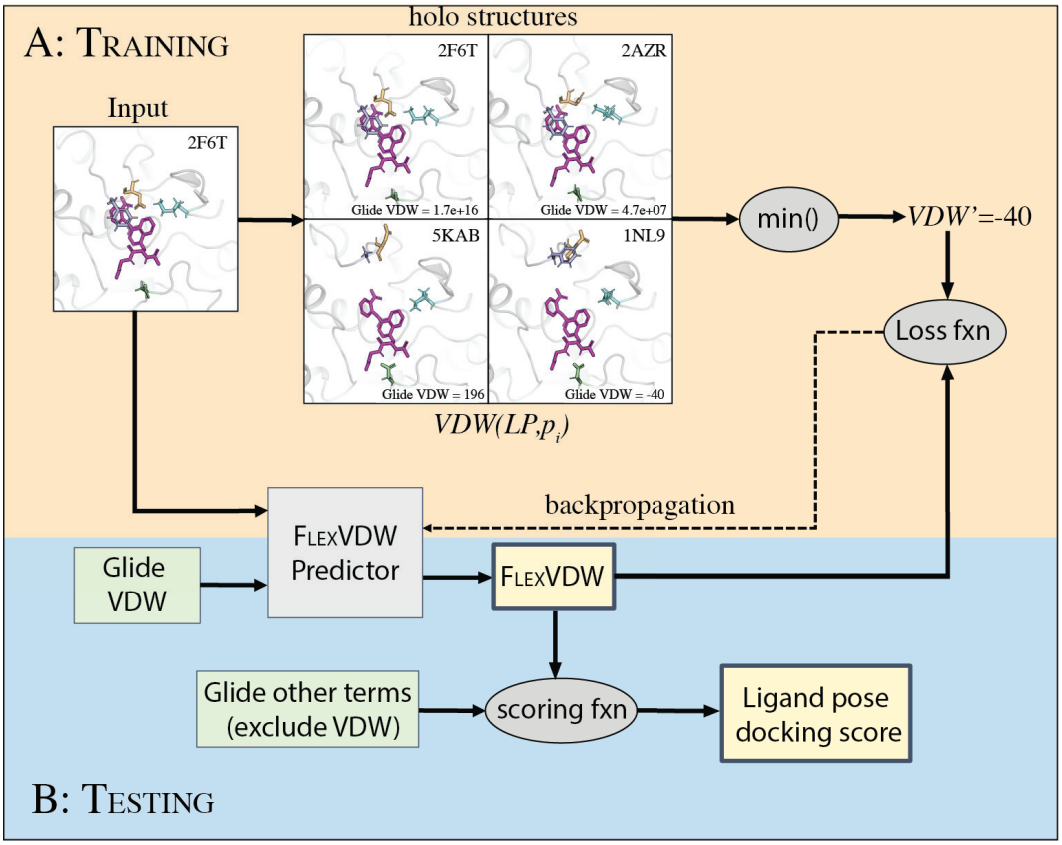

Figure S2: High-level overview of training and testing of FLEXVDW. (A): Given a protein-ligand pose structure and its Glide VDW score, we train FLEXVDW to predict the VDW score of the ligand pose accounting for the protein flexibility by using VDW' as training label. VDW' is defined as the minimum of the VDW scores of the ligand pose when superimposed on other holo structures of the same protein (see equation 1). Here, the input ligand pose clashes with the loop residues PHE182 (light purple) and ASP181 (yellow) of the input docking structure 2F6T. When superimposed on other holo structures, the ligand pose severely clashes with the loop residues in structure 2AZR. This loop, which contains amino acid residues PHE182 and ASP181, is located further upwards in structures 5KAB and 1NL9 compared to in 2F6T and 2AZR. In structure 5KAB, there are only minor clashes of the ligand pose with with residues ASP48 (green) and LYS120 (cyan), and in structure 1NL9, there is no clash at all. The protein conformation in 1NL9 fits the ligand pose, giving a favorable (i.e., low) VDW score of -40, even though the VDW score of the ligand pose with respect to the input docking structure 2F6T is very large (it is 1.7e+16) due to severe clashes with the loop residues in the input docking structure. (B): During testing, FLEXVDW takes only a single protein-ligand pose structure and Glide VDW score as input and outputs a predicted VDW score of the ligand pose with respect to the input protein structure that takes into account implicit protein flexibility. The predicted VDW score is integrated with other Glide terms (excluding the Glide VDW score) to calculate the final docking score of the ligand pose.

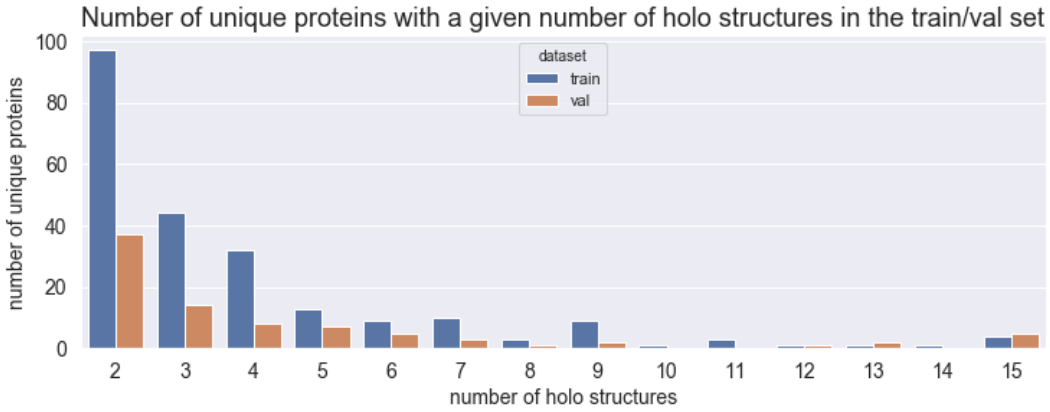

Figure S3: Distribution of the number of holo structures per unique protein used to generate the labels and docking poses in the training and validation sets. We limit the maximum number of holo structures to 15 to prevent overrepresentation of certain proteins.

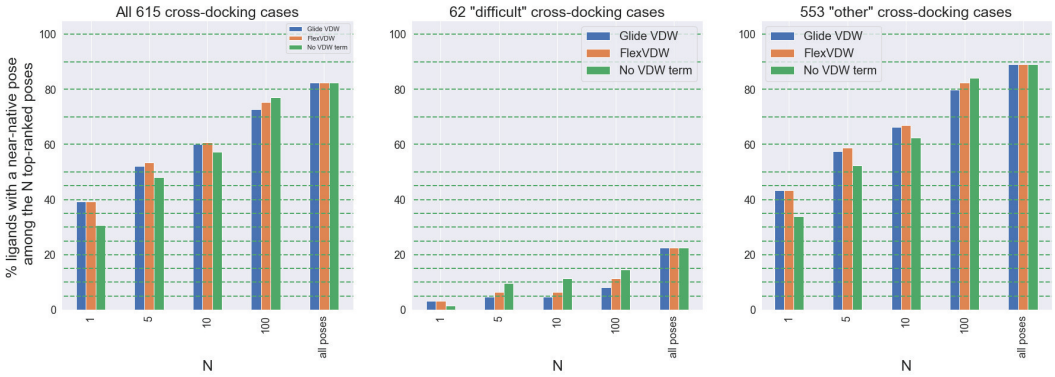

Figure S4: Percentage of cases for which a near-native pose is included in the top-N poses sorted by docking score (higher is better). This analysis differs from that of Figure 3 only in that the refined native ligand pose was not added to the set of poses considered for each ligand. The bars labeled "all poses" show the fraction of cross-docking pairs in which any near-native pose is present in the set of candidate poses. Only about 80% of the cross-docking pairs have at least one near-native pose in the generated candidate poses, which makes the comparison between methods difficult, especially for the "difficult" cases, because for most of those cases, the candidate pose set does not include any near-native pose.

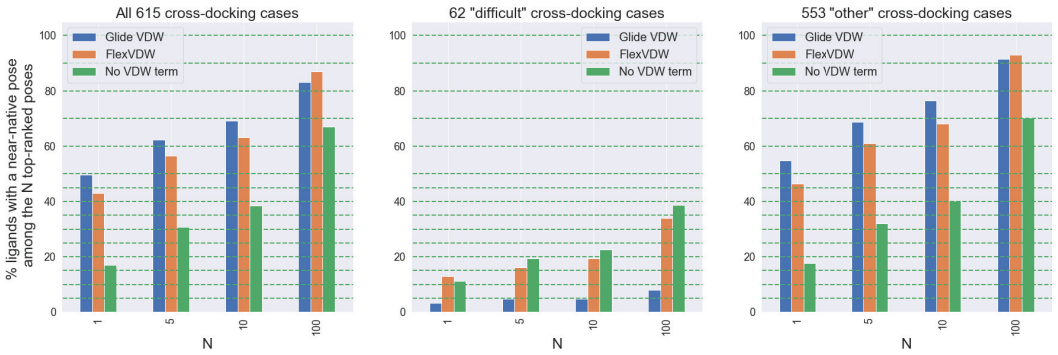

Figure S5: Percentage of cases for which a near-native pose is included in the top-N poses sorted by docking score (higher is better). This analysis differs from that of Figure 3 only in that we use VDW radius scaling of 0.0/0.0 instead of 0.6/0.5 for receptor/ligand for the softened VDW parameters. This further softening of VDW parameters leads to generation more "garbage" poses (i.e., poses with severe collisions with receptor backbones). As can be seen in the figure, the performance of FLEXVDW is worse here compared to in Figure 3. This is to be expected since the ML model never sees such poses during training. However, a scoring function that includes FLEXVDW generalizes better than one that does not include a VDW term.

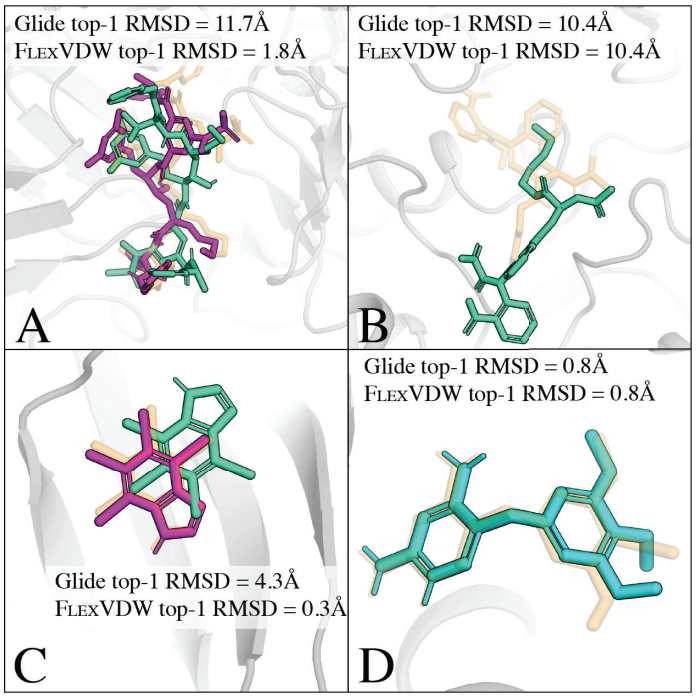

Figure S6: Comparison of the posed ranked first (top-1) by Glide (cyan) and FlexVDW (magenta). The experimentally determined native poses are shown in transparent light orange. (A): Ligand of 3I25 docked to the protein structure of 1FKN (Uniprot ID: P56817). Due to clashes of several residue side chains with the native ligand pose (see Figure 4A), Glide's top-1 ligand pose (cyan) was placed in the opposite orientation to the native ligand pose. FlexVDW, on the other hand, was able to select a near-native ligand pose as the top-1 pose. (B): Ligand of 1N06 docked to the protein structure of 1BZC (Uniprot ID: P18031). Although FlexVDW predicts a negative VDW score for the native ligand pose (see Figure 4B), the near-native poses are eventually rejected due to the high electrostatic repulsion energy (not shown in the figure). Hence, FlexVDW selects the same pose as Glide as top-1. (C): Ligand of 2OXY docked to the protein structure of 2PVJ (Uniprot ID: P28523). FlexVDW selects a near-native pose as top-1, while Glide does not. (D): Ligand of 2W3A is docked to the protein structure of 1BOZ (Uniprot ID: P00374). Both Glide and FlexVDW select the same pose as top-1.