A Case-Based Meta-Learning Algorithm Boosts the Performance of Structure-Based Virtual Screening

Xi Yun
Dept. of Computer Science &The Graduate Center
The City University of New York
New York City, U. S. A.
xyun@gc.cuny.edu

Susan L. Epstein
Dept. of Computer Science & The Graduate Center
Hunter College, The City University of New York
New York City, U. S. A.
susan.epstein@hunter.cuny.edu

Weiwei Han
Key Lab for Mol Enzym & Eng of Ministry of Education
Jilin University
Changchun, P. R. China
weiweihan@jlu.edu.cn

Lei Xie*
Dept. of Computer Science & The Graduate Center
Hunter College, The City University of New York
New York City, U. S. A.
lei.xie@hunter.cuny.edu

***ABSTRACT*—Virtual screening based on protein-ligand docking is widely applied at the early stage of drug discovery. Scoring functions from a diverse set of existing protein-ligand docking tools, however, often poorly distinguish bioactive compounds from inactive ones. As a result, considerable effort has been devoted to the combination of multiple scoring functions for more reliable evaluation. State-of-the-art consensus scoring or ensemble learning methods assume each scoring function performs uniformly for all cases. Case-based meta-learning (CBML), the method we have developed, is fundamentally different. It identifies the best predictor for a specific new case based on its similarity to old cases and uses that method to predict rather than average the performance of all predictors. Our large-scale benchmark studies clearly indicate that CBML outperforms consensus-based scoring and significantly improves the performance of structure-based virtual screening. The CBML paradigm can be extended to other applications in bioinformatics and chemoinformatics for robust and reliable predictive modeling.**

Keywords—Meta-predictor; case-based meta-learning; protein-ligand docking; virtual screening; consensus scoring

I. INTRODUCTION

The discovery of drug-like lead compounds that bind to a specific disease-causing protein (i.e. drug target) is a central task at the early stage of drug discovery. *In vitro* high-throughput screening (*HTS*) is an established experimental technique for this purpose. HTS is not only costly and time-consuming, but also associated with high false-positive rates. *In silico* virtual high-throughput screening (*VHTS*) is an attractive alternative with the potential to save time, reduce costs, and improve the hit rate. When the three-dimensional (*3D*) structure of a target protein is available, virtual screening based on protein-ligand docking (*PLD*) is widely applied to identify and optimize drug lead compounds. PLD is a molecular modeling technique that evaluates a ligand's binding pose (e.g., orientations and conformations), and the strength of interaction once it is bound to a protein receptor or enzyme (e.g binding free energy). Many PLD programs explore the search space of possible orientations and conformations to identify those with the strongest binding (i.e., minimal binding free energy) in a protein-ligand complex. Binding free-energy prediction is thus critical for PLD, but most PLD software does it poorly.

To leverage state-of-the-art PLD software and to improve the performance of PLD-based VHTS, considerable effort has been devoted to the combination of multiple scoring functions [1-3]. These methods either average scores or take the majority opinion from a set of algorithms that predict the strength with which a protein will bind to a ligand. A fundamental shortcoming of consensus scoring and ensemble learning methods is their assumption that each scoring function performs uniformly well or uniformly poorly for all protein-ligand pairs *(cases*). In reality, this assumption does not hold. The protein-ligand interaction data used to train scoring functions is noisy and biased. Because each individual scoring function introduces its own systematic errors, a scoring function may perform dramatically differently on different cases. As a result, if most scoring functions are inaccurate on a case, consensus scoring will be too, even if some scoring function is highly reliable on that case. Another practical difficulty in PLD-based VHTS is that no individual PLD program or combination of them can assess the quality of its prediction on a specific case. This hinders the application of PLD-based VHTS in real drug discovery.

To address the above challenges, this paper introduces a

[*] Corresponding author

novel algorithm, case-based meta-learning (*CBML*). The premise of CBML is that close predictive accuracy of a single scoring function on similar cases supports reasoning from a set of scoring functions about a new case. To predict on a new case, CBML identifies the most similar cases among its benchmarks, and selects the scoring function that performed best on them. The principal result reported here is that CBML significantly improves predictive accuracy on PLD-based VHTS. To the best of our knowledge, this is the first work that exploits case-based meta-learning to combine multiple protein-ligand docking programs for VHTS applications. Furthermore, we introduce a method to assess the reliability of case-based prediction. Although the ability to report such reliability is essential for hypothesis generation in biological discovery, it is rarely available from most chemoinformatics and bioinformatics methods. In summary, CBML is a promising tool that should support a broad range of applications in bioinformatics and chemoinformatics for reliable predictive modeling.

II. RELATED WORK

Our new method, CBML, addresses PLD-based VHS with a meta-learning method based on case-based reasoning (*CBR*), CBR is a machine-learning paradigm that retrieves and uses knowledge about previously experienced examples (i.e., cases) to solve a new problem. A recent CBR system, for example, diagnosed a patient based on diagnoses for the most similar previous patients [4].

CBML differs from conventional ensemble learning. Ensemble learning applies a single algorithm to a subset of data to build multiple predictors and uses their consensus as its final prediction. CBML does not seek a consensus among multiple algorithms or predictors, but instead identifies the best single predictor for a specific case.

III. METHODS

PLD software

The work reported here takes scoring functions from three orthogonal PLD tools: eHiTS [5], Autodock Vina [6], and Autodock [7]. Each has its own strategies for scoring. Although Autodock and Autodock Vina bear similar name, they have developed different scoring functions. Autodock's scoring function applies a force-field-based approach derived from physical phenomena. Autodock Vina's empirical scoring functions sum individual energy terms, and then train parameters on co-crystallized protein-ligand complexes with experimentally determined binding affinities. Finally, eHiTS combines empirical and knowledge-based scores trained from known protein-ligand complexes. Because of their different algorithms and training data, PLD methods often have dramatically different performance on the same data set. No single method consistently outperforms the others. We do not include all available PLD tools in this study. Autodock and Autodock Vina are two of the most used open source docking tools. In recent benchmark studies, eHiTS was in general the best among 19 scoring functions [8], and ranked among the top 7 docking tools [9]. Thus eHiTS can serve as a baseline from which to evaluate CBML. If CBML outperforms eHiTS, it is likely that CBML would outperform other PLD tools as well.

```
Algorithm 1: CBML ($e, C, d, F$)
(1) Select a subset $M$ of cases in $C$ most similar to $e$.
(2) Calculate $s(e,j)$ for all $F_j \in F$.
(3) Combine $s(e, j)$ for all $F_j \in F$ to predict a score for $e$.
```

CBML algorithm

Each example here is a chemical compound, represented for CBML by its 2D *fingerprint,* which is calculated from openbabel [10]. The similarity metric between chemicals is defined by the Tanimoto coefficient.

Let F be a set of scoring functions, where each scoring function $F_j \in F$ predicts score $s(i, j)$ on chemical c_i, and let $p(i, j)$ denote the predictive accuracy of F_j on c_i. CBML, our case-based meta-learning for example e, case set C, similarity metric d, and scoring functions $F,$ appears in Algorithm 1.

Step 1 of Algorithm 1 assembles M, a set of cases most similar to the new example e. In step 2, each scoring function F_j predicts a score for e based on F_j's predictions on M. The prediction of F_j for e is calculated as a linear combination of F_j's scores for all the cases in M:

$$s\left(e,j\right)=\sum_{c_i\in M} w_i s(i,j) \tag{1}$$

where weight w_i quantifies the similarity between e and c_i. In this work, w_i is determined by Tanimoto Coefficient, but another d or computation from properties of e alone would be a reasonable alternative.

Step 3 in Algorithm 1 combines the scores from all F_j in F on the cases in M to make a final prediction for e. In CBML, a prediction from a scoring function that performs better on M have more influence on the final prediction. Let $p(M, j)$ denote a set-based performance measurement for the overall predictive accuracy of F_j on the cases in M. CBML emphasizes cases that are more similar to e, with the same weights used in equation (1):

$$p(M,j)=\sum\nolimits_{c_i\in M} w_i p\left(i,j\right) \tag{2}$$

Finally, we use a winner-take-all approach to combining the $s(e,j)$ scores based on multiple predictions $p(M, j)$, applicable to both discrete and continuous values:

$$s(e,F^*) = s(e,\ \mathrm{argmax}_{a_j}\, p(M,j)) \tag{3}$$

Case base and benchmark

We test CBML on protein-ligand docking with examples drawn from *DUD*, a set of benchmarks for virtual screening [11]. Along with each ligand, DUD includes 36 decoys intended to challenge a PLD algorithm.

Typically, different PLD scoring functions predict on incomparable scales, a concern for a meta-predictor that relies upon multiple scoring functions. We therefore use a simple but robust *rank-regression scoring* mechanism that uniformly maps the raw scores from any $F_j \in F$ to a normalized rank score. The scores from F_j thereby become independent of its scale; they reflect only the preference of F_j for one case over another. More formally, given a set C of n reference cases, CBML calculates rank-regression scores as follows. For each $F_j \in F$, CBML sorts the $s(i,j)$ raw scores for $c_i \in C$ in ascending order, replaces the scores with their rank, and then normalizes that rank in [0,1]. Note that this process assigns smaller scores to higher-ranked cases, to coincide with the premise that a smaller binding-energy score is better. The accuracy of the algorithm F_j on example c_i is

$$p(i,j) = \begin{cases} \dfrac{\left|\{c_k \in D \mid s(c_k,j) > s(c_i,j)\}\right|}{\left|\{c_k \in D\}\right|} & \text{if } c_i \in L \\ \dfrac{\left|\{c_k \in L \mid s(c_k,j) < s(c_i,j)\}\right|}{\left|\{c_k \in L\}\right|} & \text{if } c_i \in D \end{cases} \tag{4}$$

Experimental design

Each of our experiments has a predictor that predicts the score of a chemical e to a receptor. We examine the predictive accuracy of five predictors: three individual predictors (eHiTS, AutoDock Vina, AutoDock) and two meta-predictors, CBML and RankSum, detailed below.

For CBML, F was {eHiTS, AutoDock Vina, AutoDock}. We first computed the similarities between all pairs of chemicals in DUD for the same receptor, and recorded the five chemicals most similar to each chemical, with their scores. Next, we evaluated the three individual predictors with leave-one-out validation.

RankSum is a typical bioinformatics meta-predictor. Each individual predictor ranks chemicals with respect to their rank-regression score. To predict the score on example *e,* RankSum totals the ranks from the three predictors, where a lower score is better. Note that RankSum requires scores from all predictors for each chemical.

We measure the performance of a scoring function by its *enrichment ratio*, the ratio of the number of true positive (i.e., active) compounds to the number among compounds ranked in the top 5% overall.

IV. RESULTS and DISCUSSION

CBML considerably outperforms both consensus scoring and individual predictors

We report first on CBML-1N, a simple but effective version of Algorithm 1, where M is only a single case c, the one most similar to e. Thus, to predict a score for e, CBML need only compute $p(c,\ j)$ for each $F_j \in F$. As Figure 1 shows, CBML-1N clearly outperforms both consensus scoring and each of the individual PLD tools. Nearly 70% of all receptors achieve an enrichment ratio above 40% with CBML-1N, almost twice the performance of RankSum.

The superior performance of CBML-1N comes from its ability to consider and exploit protein-ligand pairs case by case. Because CBML-1N identifies such chemical types, it can more accurately determine which predictor should be used to rank a specific chemical. Note, however, that an algorithm that always selects the best performer (here, eHiTS) cannot ever exceed that performance. We believe that reliance on similar cases makes CBML more resilient than consensus scoring to occasional poor predictions from individual predictors. Of course, were all of F consistently poor on all examples, CBML would not succeed, but we assume that the individual predictors were proved successful to some degree by other researchers.

CBML quantifies the confidence of predictions

A less addressed, unresolved issue in machine learning in general, and in PLD-based VHTS in particular, is how to quantify the reliability of the prediction. A reliable estimate of confidence in a prediction would greatly facilitate follow-up wet-lab experiments and decision making for VHTS. CBML provides a general framework for prediction confidence.

Our confidence analysis considers three kinds of predictions, based on chemical similarity and scoring function accuracy on M. Two chemicals are termed *similar* if and only if their similarity is greater than t_1 (here, 0.8),

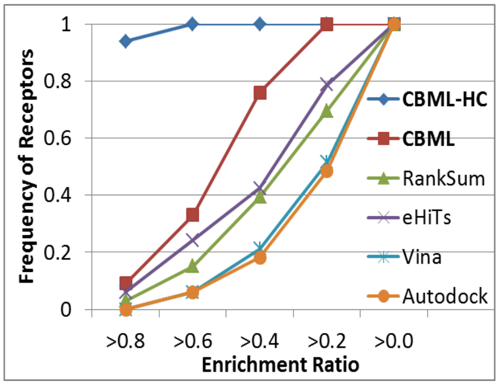

Figure 1. Cumulative distribution of receptors vs. the enrichment ratio for chemicals ranked in the top 5%

and *dissimilar* otherwise. A *reliable* predictor is one whose performance, as calculated by equation (4), is greater than t_2 (here, 0.9); otherwise it is *unreliable*. Together t_1 and t_2 define three categories of predictive ability for a scoring function F_j that predicts on testing example *e*. A prediction has *high confidence* if *e*'s closest neighbor *c* is similar to *e* and F_j is reliable on *c*. A prediction has *low confidence* if *c* is dissimilar to *e* and F_j is unreliable on *c*. In all other situations cases, a prediction has *normal confidence.*

The superior prediction power of high-confidence CBML-1N (here called *CBML-HC*) spans all 34 receptors tested, as shown in Figure 1. The percentage of CBML-HC varies from one receptor to the next, but averages 46.4%. Thus, when a chemical is predicted as active with high confidence, it is very likely to be a real active compound, and worthy of further experimental validation.

The performance of low-confidence CBML provides a measurement of the underlying inaccuracy of CBML, which is separate from that of the PLD tool itself. If CBML cannot correctly identify the suitable cases, its performance degrades. Thus the accuracy of CMBL critically depends on the performance of its case similarity metric (measured here as the chemical fingerprint similarity). On the other hand, if CBML identifies the correct case, but no PLD tools perform well, neither will CBML. In such a situation, the inclusion of more PLD tools should increase CBML's performance still further, as long as some PLD tool provides accurate scoring.

VI. CONCLUSION

CBML is a case-based meta-predictor, applied here to improve compound virtual screening using PLD. Results here suggest that CBML outperforms any individual PLD predictor, as well as conventional consensus scoring. Furthermore, a method is proposed to estimate reliability in CBML predictions. This approach makes it possible to apply PLD to solve real drug-discovery problems. In practice, experimental design can focus on high-confidence predictions, which promise a high success rate.

Given a domain-specific similarity metric that compensates for individual predictors by its focus on additional relevant features, CBML is applicable to other bioinformatics and chemoinformatics problems. Examples include protein structure prediction, protein-protein interaction, protein-nucleotide interaction, disease-causing mutation, and the functional roles of non-coding DNA.

Ligand-based VHTS (e.g., 3D Quantitative Structure-Activity Relationship) is also widely applied in drug discovery. PLD-based and ligand-based VHTS use fundamentally different algorithms. Although it would be interesting to compare the performance of PLD-based methods with that of ligand-based methods in VHTS, such comparison is beyond the scope of this paper. It will be explored in future work.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under grant IIS-1242451, CNS-0958379 and CNS-0855217 and the City University of New York High Performance Computing Center at the College of Staten Island.

REFERENCES

[1] R. Wang and S. Wang, "How Does Consensus Scoring Work for Virtual Library Screening? An Idealized Computer Experiment," *J Chem Inf Comput Sci,* vol. 41, pp. 1422-1426, 2001.

[2] M. A. Miteva*, et al.*, "Fast Structure-Based Virtual Ligand Screening Combining FRED, DOCK, and Surflex," *J Med Chem,* vol. 48, pp. 6012-6022, 2005.

[3] H. Fukunishi*, et al.*, "Bootstrap-based consensus scoring method for protein-ligand docking," *J Chem Inf Model,* vol. 48, pp. 988-996, 2008.

[4] C. Pous*, et al.*, "Diagnosing patients with a combination of principal component analysis and case based reasoning," *International Journal of Hybrid Intelligent Systems,* vol. 6, pp. 111-122, 2009.

[5] Z. Zsoldos*, et al.*, "eHiTS: a new fast, exhaustive flexible ligand docking system," *J Mol Graph Model,* vol. 26, pp. 198-212, 2007.

[6] O. Trott and A. J. Olson, "AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading," *J Comput Chem,* vol. 31, pp. 455-61, 2010.

[7] D. S. Goodsell*, et al.*, "Automated docking of flexible ligands: applications of AutoDock," *J Mol Recognit,* vol. 9, pp. 1-5, 1996.

[8] P. Englebienne and N. Moitessier, "Docking ligands into flexible and solvated macromolecules. 4. Are popular scoring functions accurate for this class of proteins?," *J Chem Inf Model,* vol. 49, pp. 1568-80, 2009.

[9] D. Plewczynski*, et al.*, "Can we trust docking results? Evaluation of seven commonly used programs on PDBbind database," *J Comput Chem,* vol. 32, pp. 742-55, 2011.

[10] N. M. O'Boyle*, et al.*, "Pybel: a Python wrapper for the OpenBabel cheminformatics toolkit," *Chem Cent J,* vol. 2, p. 5, 2008.

[11] N. Huang*, et al.*, "Benchmarking Sets for Molecular Docking," *J Med Chem,* vol. 49, pp. 6789–6801, 2006.