Modeling and Predicting Protein Reactivity: A Comprehensive Approach with Tree-Based Models and Deep Convolutional Networks

Jackson Kaunismaa[a], **Julia L. Wang**[a], **Ryan Chen**[a], **Tony Fu**[a], **Vishnu Akundi**[a]
[a]University of Toronto, Faculty of Applied Science and Engineering, Department of Engineering Science

Abstract. This study focuses on developing machine learning (ML) models for predicting the covalent reactivity of amino acid active sites, with a primary focus on serine. Leveraging a diverse set of ML models, including Logistic Regression, Random Forest, Gradient-Boosted Trees, 3D Convolutional Neural Networks (3D CNNs), and Deep Convolutional Neural Networks (Deep CNNs), we present a comprehensive analysis of model performance. The serine-specific models reveal the efficacy of tree-based approaches, with Random Forest and XGBoost emerging as robust performers. Residue-agnostic models extend the applicability of Deep CNNs across various amino acids, showcasing promising generalizability. Our findings underscore the need for tailored models for different amino acids and contribute valuable insights to the intersection of machine learning and drug discovery, particularly in the context of Targeted Covalent Inhibitors (TCIs). This research aims to optimize drug design processes by refining predictive models for covalent reactivity in diverse biological contexts. Overall, this study provides nuanced insights into ML model dynamics, offering avenues for further refinement and optimization in the pursuit of innovative and impactful drug therapies.

Keywords: TCI, drug discovery, machine learning, serine, amino acids.

1 Introduction

In the landscape of drug discovery, ML algorithms play a crucial role, particularly in the intricate processes of target-specific drug design. The development of targeted covalent inhibitors (TCIs) stands out as an innovative approach, wherein these therapeutic agents form irreversible covalent bonds with disease-associated proteins, offering precise and tailored interventions [1, 2]. Covalent reactivity, representing the ability to form such bonds, determines the druggability of the site. ML integration has significantly improved TCI identification and optimization, with applications ranging from support vector machines [3] to graph neural networks [4]. This report addresses the importance of ML in drug discovery, emphasizing its impact on resource conservation and expediting therapeutic development.

The advent of the robust and continually advancing AlphaFold2 (AF2) [5] further underscores the potential of predictive structure-based ML models in TCI discovery campaigns. Specifically, in the context of the amino acid serine, ML models contribute valuable insights into sites that may elude traditional chemoproteomic methods, aiding in understanding structural changes due to phosphorylation [6]. Here we detail the rigorous development and validation of both serine-specific ML models and residue-agnostic models trained and tested on all amino acids. Building upon successful models for cysteine [3, 4, 6], the models aim to meet specific requirements, prioritizing area-under-curve (AUC) metrics and precision on held-out test data. These metrics are chosen with the recognition that false positives are more costly than false negatives in the context of TCI. In the pursuit of innovative drug therapies, false positives can potentially lead to resource-intensive laboratory experiments, consuming our clients' valuable time and materials. A false-positive prediction could prompt unnecessary experimental validations for a non-reactive site, resulting in an inefficient allocation of laboratory resources. Therefore, prioritizing metrics

that minimize false positives, such as precision and area-under-curve (AUC), becomes paramount to ensure the judicious use of experimental resources in drug discovery scenarios.

In addition to optimizing AUC and precision, the models prioritize generalizability to unseen test data. This aspect is crucial for practical drug discovery scenarios where identifying potential drug candidates requires models that can perform effectively on new, unobserved data. The project aligns with the imperative to bridge computational modelling with laboratory experimentation, saving valuable resources and time in the pursuit of groundbreaking drug therapies. As we delve into the results and discussions, the emphasis remains on meeting these requirements and furthering our understanding of ML models' performance in predicting covalent reactivity, contributing to the refinement and optimization of predictive models for diverse biological contexts.

2 Data

The data utilized consists of various proteins from the RCSB Protein Data Bank (PDB) [7]. In the PDB, each protein entry has an associated ID and is organized hierarchically, providing details about the sequence of residues, including structural and positional data for each atom in each residue, where coordinates are given in angstroms. As such, it contains atom-level and residue-level data for each protein. Notably, the dataset includes a total of 3875 reactive sites, with 1410 corresponding to the amino acid serine.

During data processing, it was noted that multiple proteins shared identical amino acid sequences. This occurrence stemmed from the inclusion of various conformations of the same protein in the dataset. Conformations represent distinct spatial arrangements of constituent atoms, defining the overall shape of the protein [8]. However, opting to exclude different conformations would result in a substantial 85% reduction in the dataset, an undesirable outcome. Therefore, the strategy employed for train-test splits ensured that identical conformations of a protein did not appear in both the training and test datasets. Two variations of each model were developed: one for serine-specific training and another for residue-agnostic training, encompassing all amino acids.

The client supplied positive and negative labels for residues, along with their corresponding covalent reactivity information. The features employed varied based on the model, and the extraction process drew inspiration from previous work by Liu et al. [6], who constructed models to classify the reactivity of cysteine, another amino acid. The following sections delineate our feature extraction methodologies, aiming to derive features offering pivotal insights into protein structure and bonding capabilities.

2.1 Tabular Feature Extraction

A pipeline to extract 36 tabular-based categories of features was built for each target serine within a PDB, outlined below.

Closest type features is a broad feature that represents the distance of the x closest atoms in some category to the target serine. **Amount type** features return an integer value for the number of atoms within a certain distance of the serine. These features contain information about potential hydrogen bonding and electrostatic interactions and are listed in Appendix 8.1.

The **secondary protein structure** of each residue in the PDB is identified using the Dictionary of Protein Secondary Structure (DSSP) [9]. This is 3D information about how the protein chain curves in space, which was computed for every residue in a small window centred on the target residue. As such, this feature considers the 3D arrangement of amino acids. These secondary

protein structures are extracted for serine at Ser-4, Ser-2, Ser, Ser+2, and Ser+4, where a similar methodology is employed for all other amino acids.

PredyFlexy [10] software is used to extract **residue flexibility**, which provides a nuanced perspective on a protein's flexibility at the molecular level. This is achieved by predicting flexibility scores for each residue through the utilization of Root Mean Square Fluctuation (RMSF) and normalized B-Factor. Flexibility features allow for discerning regions of the protein that are more prone to conformational changes, although this feature extraction is computationally expensive.

The **fpocket** [11] program is utilized to identify and extract crucial information about potential binding sites known as pockets. Pockets are surface indentations on proteins that serve as promising sites for ligand bonding in drug design. The extraction process includes determining the distance to these identified pockets which enables pinpointing regions on proteins likely to interact with ligands, enhancing the precision of drug discovery efforts by focusing on key binding sites.

The calculation of **Solvent-Accessible Surface Area** (SASA) was incorporated using Biopython [12] for each atom, resulting in 2 features. This assessment allowed for the evaluation of the exposure of serine residues from their surrounding environment, providing crucial insights into their accessibility and, consequently, their potential for ligand bonding. The SASA metric enhances understanding of the structural dynamics of proteins, guiding the drug design efforts by pinpointing regions of interest for effective ligand interactions. By considering SASA, the models gain a comprehensive understanding of the molecular landscape, and we sum the SASA values for all sidechain atoms and subsequently for all mainchain atoms.

2.2 Voxel Feature Extraction

Spatial data plays a pivotal role in determining the covalent reactivity of a targeted active site residue since surrounding atoms and their chemical properties significantly influence the ligandability of the active site. CNNs utilize this spatial information by evaluating three-dimensional pockets around the center atom, capturing the impact of neighbouring atoms. Recognizing the variability in protein orientations observed by researchers, we addressed this challenge by developing a rotationally invariant model. In data processing, we employ random angle sampling from 4400 valid angles to augment the data, ensuring a diverse representation of the protein's candidate target residue. The valid angles shown in Figure 1 were determined by testing over 10 different serine amino acids without duplicates, centring on the alpha carbons. This methodology is similarly employed for the residue agnostic data for all amino acids.

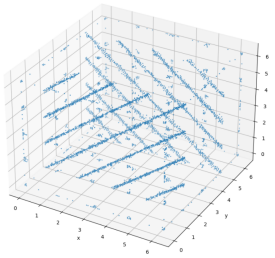


Figure 1: Valid Angles

The input data is transformed into a 4-dimensional array for CNN analysis, incorporating three spatial dimensions and features. To enrich this representation, structural and chemical atom data are extracted using Pybel (OpenBabel) [13], while atom residue data is extracted using Biopython [12]. To provide a more refined spatial context, a 3D cube grid is formed, centred around the alpha carbon of the protein, resulting in a $36 \times 36 \times 36 \times n$ array with a 1 Angstrom resolution (see Figure 2) where n is the number of features. For serine, $n = 20$, where $n = 35$ for the residue agnostic model for generalization to all amino acids. This approach allows for comprehensive rotation within the localized 3D space, ensuring that no atoms are rotated out of the analyzed area.

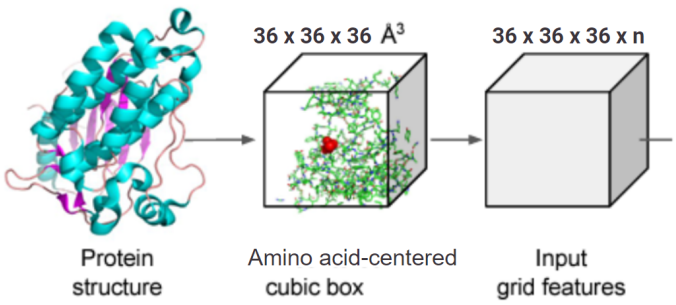


Figure 2: 3D CNN Data Processing

The chosen features for each atom predominantly encompass chemical and structural properties, as these factors frequently serve as the most pertinent indicators and predictors of an atom's covalent reactivity. Among these, crucial chemical properties include atom charge polarity and valence features. A detailed list of the 20 features employed in representing protein data for covalent reactivity classification can be found in Appendix 8.2.

3 Methodology

We developed a diverse set of 5 models to comprehensively assess and predict protein reactivity. Logistic Regression, Random Forest, and Gradient-Boosted Trees leveraged tabular features from Section 2.1 in their training and evaluation processes. In parallel, 3D CNN and Deep CNN were designed to operate on the voxel features outlined in Section 2.2. All models were trained and tested on serine only, as well as on all residues for the residue-agnostic models, ensuring a comprehensive evaluation across different amino acids. This approach allowed us to assess the models' generalization capabilities beyond serine-specific reactivity prediction.

3.1 Logistic Regression

Logistic regression is applicable to the classification of covalently reactive sites due to its simplicity and effectiveness in predicting binary outcomes using a logistic function. While it is a linear model and may not perform as well on inherently non-linear data, it aligns with the nature of the binary classification of reactive sites.

3.2 Random Forest

Random Forest [14] is well-suited for the classification of sites that are covalently reactive for TCI. As a classification algorithm, Random Forest excels in handling the complexities inherent in identifying covalent reactivity. The ensemble structure, comprising diverse decision trees trained on random subsets of data, enables the model to capture the intricate patterns and features associated with covalent binding sites. By aggregating the predictions of individual trees, Random Forest enhances accuracy and mitigates overfitting, which is significant in the context of TCI classification where the identification of specific reactive sites demands a nuanced understanding of structural and chemical characteristics.

3.3 Gradient Boosted Decision Trees

Gradient-boosted decision Trees are also applicable to predicting covalent reactivity. We implemented XGBoost [15] which utilizes iterative decision tree ensembling to enhance classification accuracy. Its iterative decision tree ensembling, optimization in the direction of the greatest gradient for loss minimization, and incorporation of the Newton-Raphson method enable it to effectively discern complex patterns associated with amino acids. Additionally, XGBoost's robust algorithmic enhancements, including regularization techniques, make it well-suited for handling the nuanced and intricate nature of covalent reactivity classification.

3.4 3D Convolutional Neural Networks

Prior work by [6] utilized a shallow CNN model that showed marginal improvement over tree-based methods. Seeking to validate these findings, we replicated their model and assessed its performance. For predicting ligandability, we hypothesized that crucial features related to serine exhibit spatial relationships. As such, we propose a 3D CNN as an approach for achieving strong performance. Our 4-dimensional feature representation comprises n feature atoms arranged in a $36 \times 36 \times 36$ localized 3D grid space, described in Section 2.2. The 3D CNN is based upon the adapted Pafnucy model in [6, 16], consisting of two convolutional layers followed by two dense layers. Each of the convolutional layers consists of a 3D convolution operation using $5 \times 5 \times 5$ kernels in 128 channels with "same" padding, followed by a ReLU layer, $2 \times 2 \times 2$ max pooling, and batch normalization. Additionally, an average pooling operation with a kernel size of 3 is conducted on the second convolutional layer's output. 3D features are then flattened into a 1D array in the dense layers. The first linear layer is followed by ReLU activation, batch normalization, and dropout with $p = 0.5$. Final linear layer outputs are binarized with a sigmoid activation. All weights were randomly initialized with zero mean and 0.001 standard deviation.

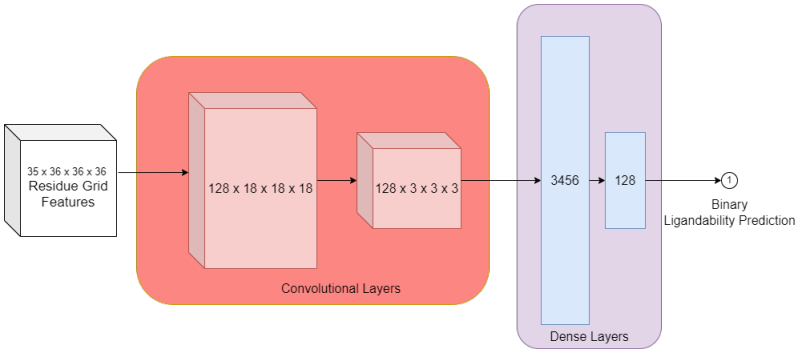


Figure 3: 3D CNN architecture for serine where $n = 35$

3.5 Deep CNN

Recognizing the intricate complexities inherent in protein structures, we propose a deep CNN model as a step forward from the shallow 3D CNN architecture. We hypothesize that a more parameter-heavy design with deeper layers may yield enhanced performance. Through comparative analysis with the 3D CNN, we aim to ascertain the efficacy of this deep CNN approach in capturing intricate features within protein structures, potentially leading to improved ligandability prediction performance. The architecture of this proposed model consists of 5 convolutional layers followed by 3 dense layers. The number of filters is $(100, 200, 400, 400, 400)$ respectively for each of the 5 convolutional layers, with kernel sizes of $(5, 5, 3, 3, 3)$ respectively, "same" padding was used. Each of the first 4 convolutional layers was followed by a ReLU operation, $2 \times 2 \times 2$ max pooling, and batch normalization. The final convolutional layer is activated by ReLU, and outputs are flattened into 1D and inputted into the dense layers. The first two dense layers utilize ReLU and batch normalization, with the second layer additionally using dropout with $p = 0.5$ to combat overfitting. The final linear layer's output is binarized by a sigmoid activation. Model weights were randomly initialized with the same parameters as the base 3D CNN.

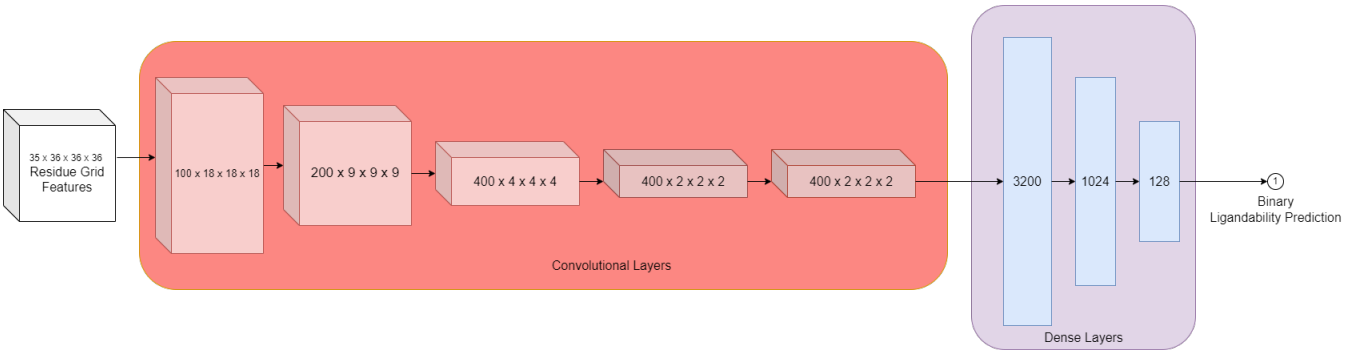


Figure 4: Deep CNN architecture for serine where $n = 35$

3.6 Model Discussion

Liu et al. (2023) [6] implemented both a 3D CNN and XGboost models on the amino acid cysteine, where the models performed similarly. While CNNs exhibit great performance across a wide range of classification tasks, due to the lack of substantial amounts of data, there is a severe risk of overfitting if the model is too large. On the other side, tree-based models like XGBoost perform better on cysteine-based active sites, and we aim to investigate whether this will hold for serine-based active sites.

4 Results

The results here present findings from two distinct perspectives: serine-specific models, which are exclusively trained and tested on serine residues, and residue-agnostic models, encompassing training on all amino acids. All results below are computed using k-fold cross-validation with $k = 5$, ensuring a comprehensive evaluation of model performance.

4.1 Serine Results

This section details the results of models that were both trained and tested on serine. Table 1 contains the results we found for XGBoost, Logistic Regression, and Random Forest models. All results below were averaged over 100 different seeds to ensure robustness. Due to flexibility features being computationally expensive as mentioned in Section 2.1, we also ran the analysis on the models with the 2 flexibility features removed.

Features	Model	Accuracy	Precision	Recall	F1-Score	AUC
All	XGBoost	94.21	96.61	**95.70**	96.14	92.64
	Logistic Regression	85.76	90.36	90.84	90.58	80.51
	Random Forest	**94.28**	**97.12**	95.27	**96.17**	**93.32**
Flexibility Features Removed	XGBoost	93.59	96.17	**95.31**	95.73	91.82
	Logistic Regression	85.97	90.53	90.95	90.72	90.82
	Random Forest	**93.78**	**96.70**	95.02	**95.84**	**92.52**

Table 1: Serine residue results for serine-specific tabular models

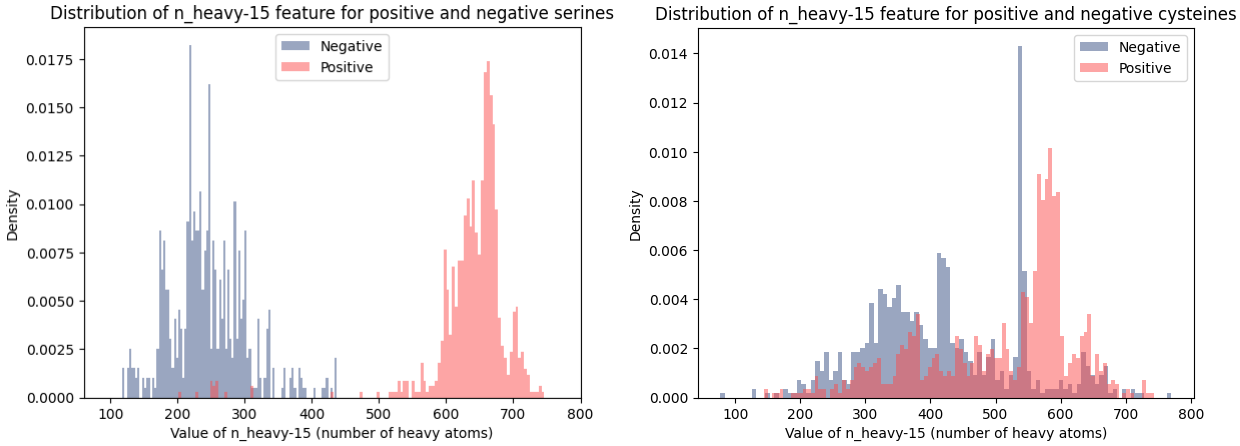
As described in Section 2.2, both CNN models developed took in the 4D array as an input, which encoded the 20 selected features. The CNN hyperparameters employed included a learning rate of 0.0001, Adam optimizer, and binary cross-entropy loss function. The training was conducted for the 3D CNN over 50 and 125 epochs, each with a batch size of 32. Table 2 illustrates

the CNN results over 5-fold cross-validation, with average training runtimes per cross-validation trial and peak VRAM requirements reported. The models were trained on a 32 GB RAM remote Linux workstation utilizing a Nvidia GTX 1080 GPU with 8 GB of VRAM.

Model	Epochs	Accuracy	Precision	Recall	F1-Score	AUC	Runtime	VRAM
3D CNN	50	86.38	93.61	86.74	89.94	N/A	1hr33m	2.1GB
3D CNN	125	89.25	93.74	90.98	92.29	87.93	2hr20m	2.3GB
Deep CNN	125	88.96	92.97	90.33	91.60	88.18	5hr12m	3.9GB

Table 2: Serine residue results for serine-specific CNN Models

During feature analysis, where the mutual information (MI) of all features was computed to gauge their informativeness, it was observed that certain features exhibited exceptionally high MI for serine residue classification. These features displayed a distinctive bimodal behaviour, wherein positive and negative reactivity classifications demonstrated significant distinctions. This phenomenon is illustrated in Figure 5a for the n_heavy-15 feature on serine.



(a) Number of heavy atoms on serine reactivity *(b) Number of heavy atoms on cysteine reactivity*

Figure 5: A comparison between the number of heavy atoms of serine vs cysteine

4.2 Residue Agnostic Results

The residue agnostic models in this section were trained with all amino acid residues to test the generalizability of methodologies in Section 4.1. Table 3 allows for performance comparison between serine and cysteine amino acids concerning models trained on all residues.

Table 4 illustrates the performance when classifying all residues for each model that was developed. The CNN models were trained over 125 epochs. We conducted two types of residue agnostic training with one including all amino acids and one excluding only serine from our pipelines, this is done because we discovered there exists unusual distributions in our serine features that could have induced biases in our dataset, as seen in Section 4.1. This discovery and its implications will be expanded below in Section 5.1.

Features	Model	Accuracy	Precision	Recall	F1-Score	AUC
Serine residue results						
All	XGBoost	97.88	97.13	**98.72**	97.89	98.08
	Logistic Regression	97.42	97.16	98.37	97.75	97.25
	Random Forest	**98.40**	**98.93**	98.37	**98.63**	**98.65**
Flexibility Features Removed	XGBoost	98.39	97.82	**98.97**	98.38	98.44
	Logistic Regression	97.91	97.68	98.57	98.11	97.97
	Random Forest	**98.84**	**99.58**	98.50	**99.03**	**99.06**
Cysteine residue results						
All	**XGBoost**	**89.69**	91.97	**88.68**	**90.10**	**89.91**
	Logistic Regression	84.24	86.09	83.56	84.55	84.72
	Random Forest	88.59	**93.25**	84.97	88.70	89.26
Flexibility Features Removed	XGBoost	87.44	91.10	85.85	88.12	87.80
	Logistic Regression	84.79	86.67	84.42	85.25	85.32
	Random Forest	**88.90**	**92.86**	**85.87**	**89.00**	**89.45**

Table 3: Serine and cysteine residue results for residue agnostic tabular models

Residues	Model	AUC	F1-Score	Recall	Precision	Accuracy
All	XGBoost	92.75	94.05	91.85	92.94	92.78
	Logistic Regression	89.57	90.45	89.37	89.91	89.58
	Random Forest	91.45	92.91	90.44	91.66	91.49
	3D CNN	**95.25**	**95.80**	**94.59**	95.18	**95.22**
	Deep CNN	95.19	95.17	93.59	**96.86**	95.17
All except serine	XGBoost	90.79	83.68	83.97	82.73	83.35
	Logistic Regression	83.67	87.04	87.95	85.43	86.67
	Random Forest	87.02	83.67	87.02	87.95	85.43
	3D CNN	92.21	92.03	90.64	**93.51**	92.19
	Deep CNN	**93.21**	**93.15**	**92.80**	**93.51**	**93.25**

Table 4: All residue results for all residue agnostic models

5 Discussion

Considering our preference for higher precision over recall, the overall metric we are interested in is the AUC as it is a combination of the two and gives us a high-level picture of the performance of the models.

5.1 Serine-specific Discussion

The serine-specific results provide valuable insights into our model performance, prioritizing higher precision over recall, and emphasizing the AUC as the primary metric. In Table 1, the removal of two flexibility features had minimal impact on model performance, suggesting their potential omission without significant consequences. As anticipated and consistent with previous findings [6], both Random Forest and XGBoost outperformed Logistic Regression, aligning with logistic regression's known limitations in handling complex, non-linearly separable data. Surprisingly, in contrast to [6], Random Forest outperformed XGBoost with an AUC of 93.32 compared to 92.64,

highlighting a nuanced performance difference that could also be addressed by more hyperparameter tuning for XGBoost.

Turning to the CNN results in Table 2, both 3D CNN and Deep CNN demonstrated lower AUC values (88.18 and 87.93, respectively) compared to Random Forest, suggesting a relative under-performance in the serine-specific context. This disparity may stem from the intricate spatial relationships within protein structures, favourably captured by tree models, especially Random Forest, while CNNs might face challenges effectively representing these relationships. The observed performance variations emphasize the need for tailored model selection in the context of serine-specific reactivity prediction.

The observed bimodal behaviour in Figure 5, where positive and negative reactivity classifications show significant distinctions, holds crucial implications for our model's understanding of serine reactivity. This behaviour suggests that certain features, such as the number of heavy atoms (n_heavy feature), could potentially introduce biases in the dataset, leading to distinctive patterns in classification. Specifically, the bimodal pattern implies that the model may be relying heavily on the number of heavy atoms to classify serine reactivity. This reliance could be attributed to the methodology of generating negative labels, where non-reactive sites are determined as those furthest away from positive reactive sites. However, this approach may introduce inaccuracies, as it assumes that distance is a robust indicator of covalent reactivity. In reality, covalent reactivity is a complex interplay of various factors, and solely relying on spatial distance might oversimplify the classification process, contributing to the observed bimodal behaviour. This highlights the importance of a nuanced understanding of features and their implications in accurate covalent reactivity prediction.

5.2 Residue Agnostic Discussion

The performance of our residue-agnostic models, as depicted in Tables 3 and 4, reveals valuable insights into the adaptability and generalizability of our models across diverse amino acids.

Specifically, when evaluating serine on the residue-agnostic model (Table 3), we observe a minimal drop in performance compared to serine-specific models. Random Forest again emerges as the top-performing model for serine, achieving an AUC of 99.06, surpassing both logistic regression (97.97) and XGBoost (98.44). Furthermore, the removal of flexibility features showcases their dispensability for enhanced generalizability.

Given the identified issues with the n_heavy feature for serine, we extended our evaluation to include cysteine in the residue-agnostic model. Remarkably, with minimal hyperparameter tuning, our model nearly matches the performance of previous works dedicated to cysteine-specific models. XGBoost and Random Forest once again shine in this context, yielding AUCs of 89.45 and 87.80, outperforming Logistic Regression with an AUC of 85.32.

Expanding our scope to encompass all amino acids displayed in Table 4, the 3D CNN stands out as the top-performing model with an AUC of 95.22, demonstrating its robustness and effectiveness across diverse amino acids. When excluding serine from the evaluation and therefore removing a potentially biased subset of data from our pipeline, the Deep CNN slightly outperforms base 3D CNN and all other methods with an AUC of 93.21, reinforcing the importance of the feature rich Deep CNN in capturing complex spatial relationships within protein structures for accurate covalent reactivity prediction.

6 Implementation

As of now, the client has not yet implemented our models, but the envisioned implementation process involves a preprocessing step to prepare the data, followed by feeding it into either the CNN or XGBoost models. The preprocessing step is crucial for ensuring that the input data aligns with the models' requirements. However, a foreseeable barrier to successful implementation is the potential presence of unreliable data, which could introduce noise and impact the models' predictive accuracy. Careful consideration and validation of the input data quality will be essential to address this challenge during the implementation phase.

7 Conclusion and Future Directions

In conclusion, our study on predicting covalent reactivity of amino acid active sites, with a specific focus on serine, revealed insightful findings. The serine-specific models demonstrated the effectiveness of tree-based models, particularly Random Forest, in capturing the nuanced relationships within protein structures. However, 3D CNNs showcased relative underperformance in this context, suggesting challenges in representing intricate spatial relationships. The residue-agnostic models extended our understanding, highlighting the generalizability of Random Forest and XGBoost across various amino acids, with the 3D CNN emerging as the top performer overall. The results underscore the importance of considering diverse models for different amino acids, emphasizing the need for tailored approaches in drug discovery applications.

Moving forward, several avenues offer possibilities for refinement and expansion of our models. First, exploring diverse datasets with positive and negative labels could enhance model robustness and generalization. Ensembling techniques, such as combining predictions from multiple models, may further improve performance by leveraging the strengths of individual models. Ensembling is particularly advantageous when dealing with complex and diverse data, providing a more comprehensive and accurate prediction. Additionally, incorporating Graph Neural Networks (GNNs) to represent proteins as graphs holds promise. GNNs can capture intricate relationships between amino acids more dynamically, potentially enhancing the predictive capabilities of our models. Continual exploration of novel features and model architectures, as well as collaboration with domain experts, will be pivotal for advancing the accuracy and applicability of our models in real-world drug discovery scenarios.

References

[1] M. Gehringer and S. A. Laufer, "Emerging and re-emerging warheads for targeted covalent inhibitors: Applications in medicinal chemistry and chemical biology," *J. Med. Chem.*, vol. 62, pp. 5673–5724, 2019.

[2] T. A. Baillie, "Targeted covalent inhibitors for drug design," *Angewandte Chemie International Edition*, vol. 55, no. 43, pp. 13 408–13 421, 2016.

[3] W. Zhang, J. Pei, and L. Lai, "Statistical analysis and prediction of covalent ligand targeted cysteine residues," *J. Chem. Inf. Model.*, vol. 57, pp. 1453–1460, 2017.

[4] H. Du, D. Jiang, J. Gao, X. Zhang, L. Jiang, Y. Zeng, Z. Wu, C. Shen, L. Xu, D. Cao, T. Hou, and P. Pan, "Proteome-wide profiling of the covalent-druggable cysteines with a structure-based deep graph learning network," *Research*, vol. 2022, p. 9873564, 2022.

[5] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Z˘´ıdek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, pp. 583–589, 2021.

[6] R. Liu, J. Clayton, M. Shen, and J. Shen, "Machine learning models to interrogate proteome-wide cysteine ligandabilities," *bioRxiv*, 2023. [Online]. Available: https://www.biorxiv.org/content/early/2023/09/11/2023.08.17.553742

[7] RCSB, https://www.rcsb.org/, 2023.

[8] J. C. Blackstock, "Chapter 4 - amino acids and proteins," in *Guide to Biochemistry*, J. C. Blackstock, Ed. Butterworth-Heinemann, 1989, pp. 32–52. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780723611516500108

[9] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, pp. 2577–2637, 1983, pMID: 6667333; UI: 84128824.

[10] A. G. de Brevern, A. Bornot, P. Craveur, C. Etchebest, and J.-C. Gelly, "Predyflexy: Flexibility and local structure prediction from sequence," *Nucleic Acids Res*, vol. 40, pp. W317–W322, Jul 2012. [Online]. Available: https://doi.org/10.1093/nar/gks482

[11] V. L. Guilloux, P. Schmidtke, and P. Tuffery, "Fpocket: An open source platform for ligand pocket detection," *BMC Bioinformatics*, vol. 10, June 2009, open Access. [Online]. Available: https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-10-168

[12] P. J. A. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M. J. L. de Hoon, "Biopython: freely available python tools for computational molecular biology and bioinformatics," *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, 2009. [Online]. Available: https://biopython.org/

[13] N. M. O'Boyle and G. R. Hutchison, "Pybel: A Python wrapper for the open Babel cheminformatics toolkit," 2008, version 2.0.0. [Online]. Available: https://open-babel.readthedocs.io/en/latest/UseTheLibrary/Python_Pybel.html

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "scikit-learn: Randomforestclassifier documentation," *scikit-learn: Machine Learning in Python*, 2011, version 0.24.2. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[15] T. Chen and C. Guestrin, "XGBoost: A scalable and accurate implementation of gradient boosting," 2016. [Online]. Available: https://github.com/dmlc/xgboost

[16] M. M. Stepniewska-Dziubinska, P. Zielenkiewicz, and P. Siedlecki, "Development and evaluation of a deep learning model for protein-ligand binding affinity prediction," *Bioinformatics*, vol. 34, no. 21, pp. 3666–3674, Nov 2018.

8 Appendices

8.1 Closest and Amount type Features

Feature Type	Atom Category Considered
Closest	Backbone nitrogens
	Backbone oxygens
	Polar atoms
	Nonpolar atoms
	Sidechain oxygens
	Sidechain nitrogens
	Nitrogens of positively charged sidechains
	Oxygens of positively charged sidechains
Amount	Heavy atoms
	Alpha carbons

Table 5: Atom Categories Considered for Different Feature Types

8.2 CNN Features

1. One-hot encoding of atom types: C, N, O, S, and others.
2. One integer representing atom hybridization: sp1, sp2, sp3.
3. One integer indicating the number of bonded heavy atoms: C, N, O, S.
4. One integer representing the number of bonded heteroatoms excluding C and H.
5. One-hot encoding of atom patterns: hydrophobic, aromatic, acceptor, donor, ring.
6. One float representing partial charge electronegativity.
7. One-hot encoding for residue types: Negative charge, Positive charge, His, Ser, polar, and others.

These features collectively provide a comprehensive and nuanced representation of the protein's structural and chemical characteristics, facilitating accurate covalent reactivity classification.

9 Attribution Table

Group member contributions.

Name	Contribution
Jackson Kaunismaa	Contributed to tabular feature extraction for tabular models for all amino acids.
Ryan Chen	Contributed to voxel feature extraction for CNNs for all amino acids.
Julia L. Wang	Contributed to feature testing, creating presentations, and all writing of the final report.
Vishnu Akundi	Contributed to exploring and evaluating tabular models: logistic regression, XGBoost, and Random Forest.
Tony Fu	Contributed to implementing, training, and evaluating Pafnucy 3D CNN and Deep 3D CNN models.

Table 6: Attribution Table