Classification of Contacts in Protein Structures with Machine Learning

Enrico Bazzacco Valentina Signor Filip Trajkovski

Abstract

In this study, we present a supervised machine learning approach for the automatic classification of residue-residue interactions (RRIs) in protein structures. Using a dataset with RING annotations as ground truth, we analyze over 2.9 million RRIs extracted from 3,914 protein structures, encompassing several distinct interaction types. To enhance the predictive capacity of our models, we enrich the input data with additional features that capture both spatial and biochemical properties of residues. These descriptors are carefully selected to balance predictive performance with computational efficiency. We assess the impact of these new features by comparing model performance on both the original and the extended datasets.

Given the significant class imbalance in the data, particularly the dominance of interaction types such as hydrogen bonds and Van der Waals interactions, we employ several strategies to mitigate this issue. These include oversampling minority classes using the Synthetic Minority Oversampling Technique (SMOTE) and incorporating class weights during training. We evaluate multiple models, including XGBoost and Neural Networks, focusing on their ability to classify individual interaction types accurately. The study also addresses key challenges such as missing values, the potential presence of multiple interaction labels for the same residue pair, and the preprocessing steps necessary to improve model performance.

Our results demonstrate that the proposed methods are effective in capturing the complexity of protein residue interactions and highlight the potential of machine learning to predict interaction types directly from structural and biochemical data.

1. Introduction

Interactions between residues within protein structures play a fundamental role in conformational stability, facilitating biological functions, and mediating molecular recognition events. Accurately identifying and classifying these interactions is essential for understanding the structural logic of macromolecules and for supporting predictive tools in bioinformatics and structural biology.

Residue Interaction Networks (RINs) offer a powerful way to represent non-covalent interactions between amino acid residues. Derived from protein three-dimensional structures, these networks capture interaction patterns based on geometrical proximity and physicochemical properties. One of the most widely used tools for analyzing RINs is the RING [1] software, which processes PDB files to detect and classify residue-residue contacts. RING assigns interaction types to each contact based on a set of predefined geometric and biochemical data. These interaction types include Hydrogen Bonds (*HBOND*), Van der Waals interactions (*VDW*), Disulfide Bridges (*SSBOND*), Salt Bridges (*IONIC*), π-π Stacking (*PIPISTACK*), π-Cation Interactions (*PICATION*), π-Hydrogen Bonds (*PIHBOND*), and a generic Unclassified category.

Despite its usefulness, the RING's rule-based approach can be limiting in the presence of atypical structures, disordered regions, or borderline cases that do not fall within the predefined criteria. In this study, we propose a Machine Learning framework to learn and replicate the RING classification of residue-residue contacts directly from structural and biochemical features. Ultimately, this work lays the foundation for a flexible prediction pipeline that can operate independently of geometry-based tools like RING, while achieving comparable performance.

2. Dataset

The dataset used in this study consists of 3,914 protein structures, each represented by an individual .tsv file.

Each file contains a list of RRIs identified within a single protein structure. Every row corresponds to a pair of contacting residues, along with a set of features describing their properties. These files were concatenated into a single DataFrame for analysis.

Across all structures, the dataset comprises approximately 2.96 million interactions, each labeled with one of several interaction types defined by the RING software. Interactions that do not meet the criteria for any known category are labeled as *Missing* (i.e., unclassified). Each interaction is described by a comprehensive set of features that can be grouped into the following categories:

- **Identification and Structural Metadata**: PDB ID (unique identifier for each protein structure), chain ID, residue index, insertion code, and residue name for the source and target residues;
- **Evolutionary Features (Atchley Factors)**: Five numerical descriptors for each residue that capture conserved biochemical properties of the amino acid sequence;
- **Structural Features**: Secondary structure (8-state DSSP classification), 3Di structural states, relative solvent accessibility, and backbone torsion angles (φ and ψ) for source and target residues;
- **Interaction**: The interaction type assigned by RING. This is the target variable for classification.

The dataset presents several challenges that must be addressed to ensure reliable model training and evaluation: the **presence of incomplete rows with missing features**, strong **class imbalance**, and the potential **multi-label nature** of the problem itself. In *Figure 1* in the *Supplementary Material*, the imbalanced nature of the problem is clearly presented.

3. Feature Extraction

To enhance model performance and incorporate more biologically relevant signals, we extended the original feature set with additional descriptors reflecting spatial and biochemical properties of RRIs. These engineered features aim to improve the model's ability to distinguish between different interaction types by encoding domain-relevant information.

3.1. Same-Chain Indicator

The original dataset encodes the chain identity of each residue using the categorical variables *s_ch* and *t_ch*. However, since these values are arbitrarily assigned per structure and can take up to 60 different values, they provide no consistent structural information to learn from. To address this, we introduced a new binary variable, *same_chain*, which indicates whether the interacting residues belong to the same chain (1) or to different chains (0). This simplified representation replaces the two original chain ID variables.

Biologically, this feature is informative: **intra-chain** contacts are typically associated with short-range interactions like hydrogen bonds or Van der Waals forces, while **inter-chain** contacts more frequently involve long-range interactions such as ionic bonds, disulfide bonds, or π-π stacking [2]. Given that only 4.23% of the interactions are inter-chain, this transformation leads to minimal information loss while improving model interpretability and reducing dimensionality.

3.2. ΔRSA - Solvent Accessibility Difference

The Δ*RSA* is the absolute difference in relative solvent accessibility (RSA) between the source and target residues. **Small** *ΔRSA* values often indicate either buried-buried or exposed-exposed contacts, whereas **large** *ΔRSA* values may suggest asymmetric or rare interaction patterns.

3.3. ΔAtchley factors

Atchley factors are five numerical descriptors derived from amino acid properties such as hydrophobicity, polarity, charge, and molecular volume [4]. For each interaction, we calculated the absolute difference for all five factors. These differences capture the biochemical similarity or dissimilarity between the residues. **Smaller** differences suggest compatible physicochemical properties, while **larger** differences may reflect interactions between contrasting residues, such as polar-nonpolar or charged-neutral pairs.

3.4. Cα Distance

The Euclidean distance between the Cα atoms of the two interacting residues serves as a geometric indicator of physical proximity. This feature is essential for distinguishing physically plausible contacts: weak interactions like *HBOND* and *VDW* require **proximity**, while specific structural motifs (e.g., disulfide bonds) occur only within **defined spatial ranges**. Of the 3,914 PDB entries initially considered, 31 were excluded due to either semantic irregularities in their IDs or because they couldn’t be downloaded via Biopython and were thus omitted when calculating the Cα distance.

3.5. 3Di Centroid Coordinates

To preserve spatial information encoded in the 3Di clustering space without resorting to one-hot encoding, we mapped each residue’s *3Di state* value to its corresponding 2D centroid coordinates (x, y) using the *states.txt* file. This results in two continuous features per residue, reducing the model's dimensionality and memory footprint while retaining useful information.

Although these engineered features are biologically and structurally meaningful, their contribution to model performance is not guaranteed. In some cases, such as *ΔRSA* or *ΔAtchley*, they may provide redundant or less informative signals compared to the original features. To rigorously assess their impact, we conducted comparative experiments between models trained on the original dataset (augmented with *same_chain*) and those trained on the enriched dataset containing all engineered features.

4. Pre-processing

Following feature extraction, we performed a pre-processing phase to clean, reformat, and prepare the data for machine learning models. This section outlines the key steps and challenges encountered during the process.

4.1. Handling a large amount of data

Given the large number of interactions, we investigated whether some data could be discarded to reduce dimensionality and remove noise. To this end, we removed:

- **Uninformative or redundant features** (PDB ID, chain ID, insertion code, 3Di letter). In the enriched dataset, we also removed the 3Di state, since its numerical coordinates were already included;
- **Unclassified interactions**;
- **Incomplete observations**, i.e., rows containing NaN values in any feature. We opted for complete case analysis, as no consistent relationship was observed between missing values and interaction types.

4.2. Multi-Label Problem Definition

Protein residue pairs can participate in multiple types of interactions simultaneously - approximately **400K** out of 1.4 million unique residue pairs in our dataset are associated with more than one interaction label. We chose to treat the task as a *multi-label classification* problem. To support this formulation, we removed duplicate rows while allowing each unique residue pair to retain all associated interaction types. We then applied **multi-label binarization**, converting the interaction labels into binary vectors where each dimension represents the presence (1) or absence (0) of a specific interaction type.

4.3. Splitting Strategy

To evaluate model performance reliably and prevent data leakage, we implemented a two-step stratified splitting procedure:

1. An initial **80/20 split** of the full dataset into a training and test set;
2. A second **80/20 split** of the training set to create a validation set.

This yielded a final distribution of approximately **64% training**, **16% validation**, and **20% test** data. To maintain class balance and avoid overfitting to specific protein structures, we enforced two key constraints. All interactions belonging to the same PDB structure were kept within the same split to avoid structural leakage, and stratification was based on the **dominant interaction type** per structure to preserve the distribution of interaction classes across subsets.

4.4. Data Augmentation with SMOTE-NC

As shown in *Supplementary Material (Figure 1)*, the dataset suffers from significant class imbalance, with some interaction types (e.g., *PIHBOND*) being heavily underrepresented compared to others (e.g., *HBOND*). This imbalance can bias models toward the majority classes.

To address this, we applied **SMOTE-NC** [3], a variant of SMOTE tailored for datasets with both numerical and categorical variables. SMOTE-NC was applied prior to one-hot encoding (OHE) because it requires categorical variables in their original format. Our sampling strategy increases the number of training samples for underrepresented labels by applying a **5x** augmentation for labels with intermediate frequency (*PIPISTACK* and

IONIC) and a **10x** augmentation for rare labels (*PICATION*, *SSBOND*, and *PIHBOND*).

4.5. One-Hot Encoding

To prepare the data for model training, categorical features lacking intrinsic order, such as residue names, secondary structure labels, and 3Di states, were encoded using One-Hot Encoding (OHE). It was applied consistently across all splits (training, validation, and test) to maintain feature alignment and ensure model compatibility. Depending on the dataset version (original or enriched), the set of encoded features varied but followed the same procedure.

5. Models

In this study, we implemented two machine learning approaches: **XGBoost** and a feedforward **Neural Network**. All experiments were conducted on both the original and enriched datasets, with and without oversampling via SMOTE-NC. Additionally, we introduced instance-level class weighting to mitigate class imbalance while training. Class weights represent the inverse frequency of each class and ensure the model pays more attention to underrepresented classes.

To handle the multi-label nature of the task, we adopted a **One-vs-Rest (OvR)** classification strategy for both models. In this setup, a separate binary classifier is trained for each interaction type, allowing us to address the uneven distribution of labels more effectively. This resulted in seven different classifiers, each optimized to detect a specific interaction class.

5.1. XGBoost

We employed **XGBoost**, a gradient boosting framework based on decision trees that iteratively combines weak learners to build a strong classifier [5]. For each interaction type, we trained a binary XGBoost model under the OvR strategy. Hyperparameter tuning was performed using a grid search over the following parameter space:

- **Max depth** (6, 8) controls the complexity of individual trees and helps prevent overfitting;
- **Learning rate** (0.05, 0.1) regulates the contribution of each tree to the final model, balancing convergence speed and generalization;
- **The number of estimators** (100, 200) specifies how many trees are used in the ensemble.

5.2. Neural Network

The **Neural Network** approach posed additional challenges due to its sensitivity to class imbalance. We initially explored a unified multi-label architecture but observed limited performance. Consequently, we transitioned to the **One-vs-Rest formulation**, training seven separate binary classifiers on the enriched dataset without SMOTE-NC, but with class weighting. This choice was also motivated by the encouraging results already observed with XGBoost under similar conditions.

Each classifier is a fully connected feedforward neural network, consisting of an input layer, three hidden layers with Leaky ReLU activations, and an output layer with a sigmoid activation to produce a probability for the positive class. Dropout and Batch Normalization were applied in order to improve the training and prevent overfitting. Training was conducted using the Adam optimizer and binary cross-entropy loss, with mini-batch gradient descent to efficiently process large volumes of data.

6. Results and Evaluation

The models were evaluated on a test set using a comprehensive suite of metrics, including *accuracy, balanced accuracy, micro and macro precision, recall, F1 score, Matthews Correlation Coefficient (MCC)*, and the *Area Under the ROC Curve (ROC AUC)*.

6.1. XGBoost Results

At the global level, the balanced accuracy of **0.8626** and macro-averaged ROC AUC of **0.9470** in the case of the enriched dataset without performing SMOTE_NC confirm the model's ability to distinguish between classes, even in the presence of class imbalance. The MCC of **0.7414** indicates a solid overall correlation between predicted and true labels. Evaluating each contact type individually, we observe distinct performance profiles. The classifiers for *PIPISTACK* and *SSBOND* contacts performed exceptionally well, with balanced accuracies of **0.9942** and **0.9896**, respectively, and F1 scores of **0.9289** and **0.9302**. On the other hand, *PIHBOND* was the most challenging contact type to classify, with balanced accuracy of **0.6219** and F1 score of **0.5329**, indicating limited generalization, especially in identifying the minority class correctly.

Augmenting the feature space produced an immediate lift in nearly every metric. *HBOND*, for example, rose from 0.72 to **0.84** Balanced Accuracy and MCC from 0.41 to **0.64**, showing that the additional cues helped the booster exploit subtle patterns previously missed. *VDW*, historically difficult because of its ubiquity, improved from 0.61 to **0.69** Balanced Accuracy and almost **doubled** its MCC, indicating better discrimination of true weak contacts from noise. The full results can be appreciated in *Tables 1-4* in the *Supplementary Material.*

Beyond predictive performance, XGBoost also offers insight into model interpretability through **feature importance** scores. These scores help identify which features (both original and engineered) contribute most significantly to the decision-making process of the model. We evaluated feature importance using three standard criteria provided by XGBoost:

- **Gain**: Measures the improvement in the model's loss function when a feature is used in a split, indicating its direct impact on predictive accuracy;
- **Weight**: Counts the number of times a feature is used in splits across all trees, reflecting its overall usage;
- **Cover**: Quantifies the number of samples affected by splits involving a given feature, capturing its reach within the data.

Analysis of the feature importance plot based on the weight criterion reveals that several of the engineered features, such as Cα distance, 3Di centroid coordinates, and ΔAtchley factors, rank among the most influential variables. Notably, these new features often surpass their original counterparts in importance, underscoring the value of domain-informed feature augmentation. In *Figures 2* and *3* in the *Supplementary Material* we can observe the difference in the feature importance in the XGBoost model between the original and enriched dataset for the *HBOND* classifier.

6.2. Neural Network Results

The NN almost matched the XGBoost's performance in all classes and slightly improved the Balanced Accuracy for rare classes such as PICATION and *PIHBOND*. However, these gains came at the cost of lower precision, yielding modest F1 and MCC values. In essence, the network became more forgiving, capturing additional positives but introducing **more false alarms** (FP). The full results from this approach can be seen in *Table 5*.

7. Experiments

To support the validity of our results, we present the following experiments, in which we compare the predictions of our **XGBoost** model against the interaction labels provided by RING.

7.1. Correct prediction

PDB_ID: 1i27
Residue pair index: (495, 499)

RING:
Interaction label: **HBOND**
Our model:
Prediction: **['HBOND']**
Probability score: **[0.9929]**

7.2. Wrong prediction

PDB_ID: 1i27
Residue pair index: (457, 460)

RING:
Interaction label: **Unclassified**
Our model:
Prediction: **['VDW']**
Probability score: **[0.6567]**

8. Conclusion

In summary, the XGBoost-based classifiers demonstrated strong predictive capabilities, especially for well-represented and biophysically distinct contact types. Nevertheless, performance varied significantly across classes, largely due to class imbalance and subtle structural features.

These results highlight both the promise and limitations of machine learning in the structural bioinformatics domain, emphasizing the need for targeted strategies such as feature enrichment to further enhance classification performance.

SMOTE offers limited incremental value once informative features are present. The best overall trade-off between recall and precision was obtained with XGBoost on the enriched set without SMOTE, yielding the highest F1 while preserving high MCC.

8.1. Unclassified Interactions Analysis

The majority of the interactions provided by RING (**1,089,547**) lack a classification. Although these missing interactions fall outside the scope of this project, we still attempted to investigate the possible reasons behind their occurrence. Most unclassified interactions do not contain missing feature values, indicating that data absence alone does not explain RING's failure to classify them. However, certain structural features - such as *ss8* and *rsa* - are strongly associated with unclassified cases and may partially contribute. Additional factors, including unresolved regions, structural disorder, or limitations in RING's classification criteria, likely play a role. No single factor clearly accounts for all unclassified interactions.

References

[1] https://ring.biocomputingup.it.

[2] Slides and notes of Structural Bioinformatics Course - Chapters: Chemical bonds, Amino acids, Distance matrix & Contact map - A.A 2024-2025.

[3] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321-357.

[4] Atchley, W. R., Zhao, J., Fernandes, A. D., & Drüke, T. (2005). Solving the protein sequence metric problem. *Proceedings of the National Academy of Sciences*, *102*(18), 6395-6400.

[5] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

Supplementary Material

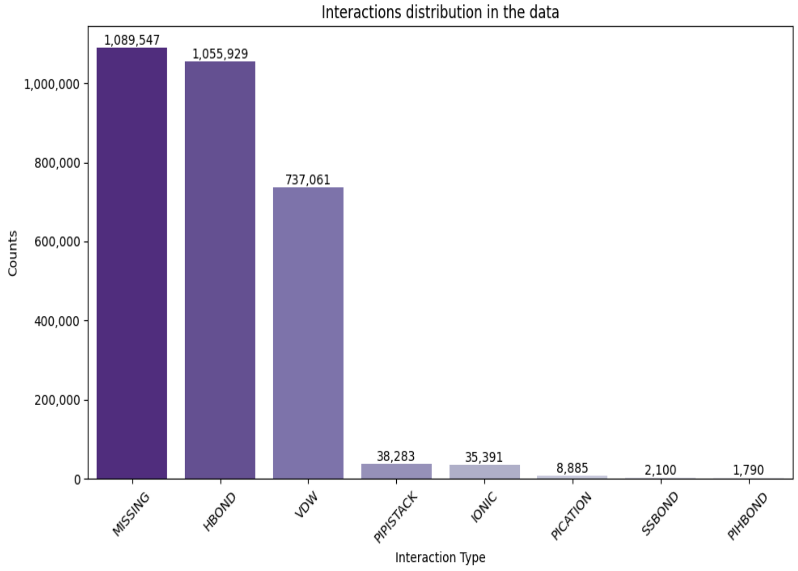

Figure 1: *Distribution of interaction types in the dataset.*

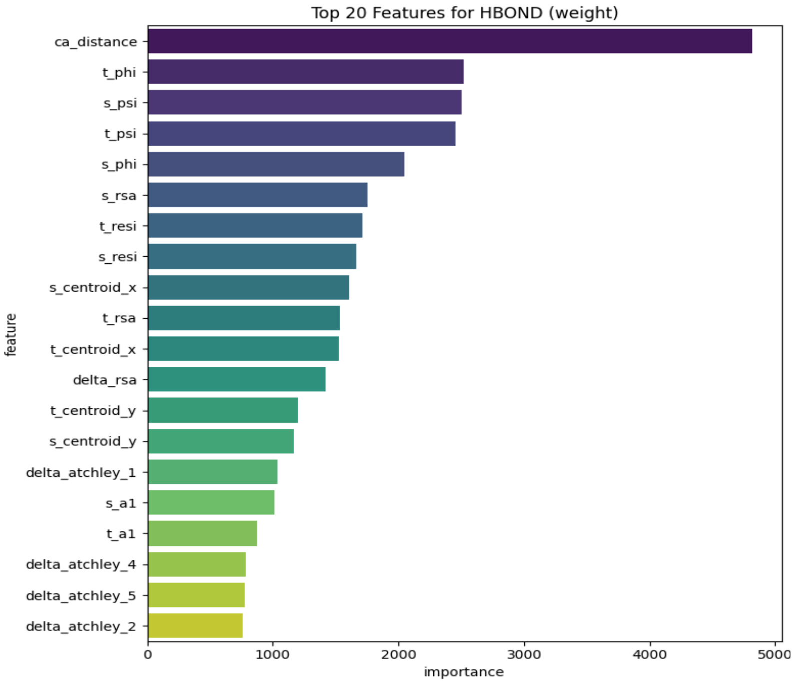

Figure 2: *Feature importance (by **weight**) given by **XGBoost** model on **enriched** dataset for **HBOND**.*

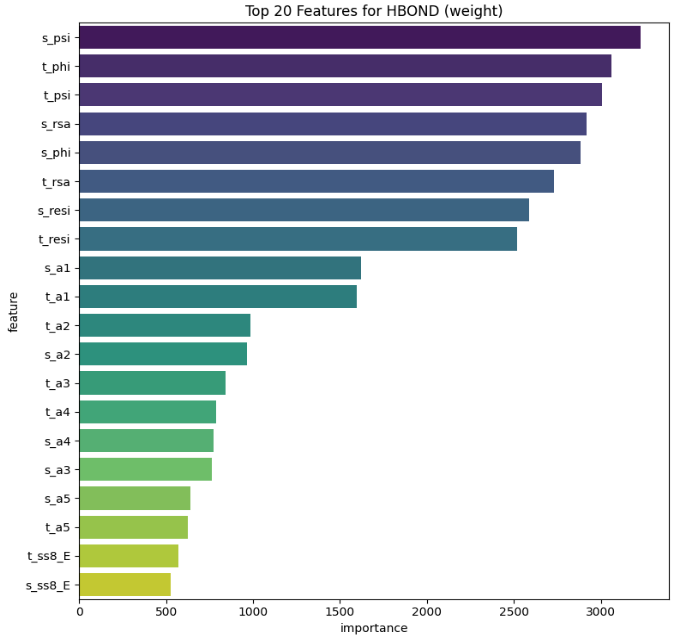

Figure 3: *Feature importance (by **weight**) given by **XGBoost** model on **original** dataset for **HBOND**.*

Performance Tables

Interaction Type/Metric	Balanced Accuracy	Precision	F1 Score	MCC	ROC AUC
HBOND	0.7187	0.6910	0.6988	0.4087	0.7943
IONIC	0.9676	0.6710	0.7421	0.5655	0.9870
PICATION	0.9573	0.6158	0.6809	0.4602	0.9927
PIHBOND	0.6300	0.5128	0.5208	0.0815	0.9412
PIPISTACK	0.9946	0.8735	0.9250	0.8596	0.9971
SSBOND	0.9986	0.8836	0.9336	0.8747	0.9999
VDW	0.6118	0.6119	0.6114	0.2237	0.6633

***Table 1**: **XGBoost** on the **original** dataset **without** SMOTE.*

Interaction Type/Metric	Balanced Accuracy	Precision	F1 Score	MCC	ROC AUC
HBOND	0.6562	0.6884	0.6668	0.3431	0.7559
IONIC	0.9597	0.6779	0.7492	0.5719	0.9862
PICATION	0.9561	0.6177	0.6835	0.4635	0.9928
PIHBOND	0.5957	0.5166	0.5272	0.0798	0.9410
PIPISTACK	0.9951	0.8726	0.9245	0.8590	0.9964

SSBOND	0.9925	0.8861	0.9329	0.8722	0.9999
VDW	0.5665	0.5878	0.5411	0.1528	0.6385

Table 2*:* ***XGBoost*** *on the* ***original*** *dataset* ***with*** *SMOTE.*

Interaction Type/Metric	Balanced Accuracy	Precision	F1 Score	MCC	ROC AUC
HBOND	0.8364	0.8021	0.8151	0.6375	0.9228
IONIC	0.9636	0.6847	0.7574	0.5853	0.9897
PICATION	0.9414	0.6374	0.7068	0.4925	0.9939
PIHBOND	0.6219	0.5199	0.5329	0.0984	0.9503
PIPISTACK	0.9942	0.8797	0.9289	0.8663	0.9982
SSBOND	0.9896	0.8837	0.9302	0.8668	0.9999
VDW	0.6909	0.6919	0.6899	0.3828	0.7738

Table 3*:* ***XGBoost*** *on* ***enriched*** *dataset* ***without*** *SMOTE.*

Interaction Type/Metric	Balanced Accuracy	Precision	F1 Score	MCC	ROC AUC
HBOND	0.8057	0.8129	0.8092	0.6186	0.9073
IONIC	0.9563	0.6965	0.7693	0.5989	0.9897
PICATION	0.9314	0.6404	0.7094	0.4923	0.9939
PIHBOND	0.6002	0.5236	0.5376	0.0974	0.9512
PIPISTACK	0.9938	0.8779	0.9276	0.8640	0.9981
SSBOND	0.9896	0.8827	0.9296	0.8657	0.9999
VDW	0.6651	0.6743	0.6620	0.3393	0.7584

Table 4*:* ***XGBoost*** *on* ***enriched*** *dataset* ***with*** *SMOTE.*

Interaction Type/Metric	Balanced Accuracy	Precision	F1 Score	MCC	ROC AUC
HBOND	0.8271	0.7882	0.8061	0.6140	0.9147
IONIC	0.9729	0.6615	0.7311	0.5527	0.9889
PICATION	0.9841	0.6003	0.6605	0.4407	0.9841
PIHBOND	0.8857	0.5065	0.4912	0.0998	0.9565
PIPISTACK	0.9949	0.8731	0.9248	0.8595	0.9976
SSBOND	0.9996	0.8027	0.8769	0.7778	0.9998
VDW	0.6827	0.6846	0.6810	0.3673	0.7589

Table 5*:* ***Neural Network*** *on* ***enriched*** *dataset* ***without*** *SMOTE.*