Enzyme Stability Prediction: Advancing with Ensemble Machine Learning and Explainable Artificial Intelligence

Swati Sampada Parida

National Institute of Technology, Tiruchirappalli - 620015, Tamil Nadu, India
swatiparida1504@gmail.com

Abstract. Accurate prediction of enzyme thermostability is crucial for bioengineering applications. This paper proposes a novel ensemble learning framework for predicting protein thermostability. The proposed ensemble learning framework combines XGBoost, a potent gradient boosting technique, with a Bidirectional Long Short-Term Memory (BiLSTM) network, which captures complex sequence-based features. The proposed framework attained the RMSE, MAE, R2 score, and Spearman Correlation coefficient of 0.37, 0.68, 0.72, and 0.76 respectively. Its performance is also evaluated against other machine-learning models and performs noticeably better than all of them. Furthermore, we leveraged Explainable Machine Learning (XML) techniques like SHAP (SHapley Additive Values), LIME (Local Interpretable Model Explainer), ELI5 and QLattice to enhance model interpretability.

Keywords: Artificial intelligence · Ensemble modelling · Protein stability prediction · Thermostability.

1 Introduction

Stability is a critical property of molecules, and when it comes to proteins, determining their stability has been accomplished through various experimental techniques such as calorimetry, denaturation studies, and optical spectroscopy. Although the number of known protein sequences is increasing, characterizing their stability lags significantly behind. Protein stability is a critical property [1] that influences various biological and industrial processes [2]. Among these, thermal stability stands out due to its significance in maintaining protein functionality under varying environmental conditions, especially in biotechnological and food science applications [3]. Thermodynamic stability, often assessed through parameters such as the free energy of stabilization and the melting temperature (T_m), which indicates the temperature at which 50% of the protein unfolds, plays a pivotal role in these considerations [4]. Protein thermal stability is based on a variety of concepts, including amino acid sequences [5], physicochemical features [6], protein chain lengths [7], temperature-dependent statistical potentials [8], microorganisms' natural temperature and salt bridges [9], and numerous other

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

features. More advanced methods have incorporated decision trees and neural networks (NN) [6] and adaptive network-fuzzy inference systems (ANFIS) [5]. While significant progress has been made in understanding protein stability, there remains a gap in effectively predicting it. The prediction methods are hindered by limited experimental data, leading to tools based on small datasets, which negatively impacts their performance due to the complexity of stability, which is influenced by various features.

Protein stability predictors are classified into two types: those that predict the stability of entire proteins and those that forecast the influence of sequence changes on protein stability [10]. In this context, we focus on predictors targeting entire protein stabilities. A variety of prediction techniques have been developed, such as length of sequences, sequence composition [5] surface and physiochemical features [11] the temperature at which organism lives and salt bridges [9], various statistical and sequence potentials [12], and different protein properties [13]. A thorough explanation of the different features and characteristics can be seen used in model training [13]. While computational methods have become increasingly important in addressing the challenge of limited experimental data, they have faced hurdles related to dataset size and bias [13]. Recent efforts have shifted towards larger training datasets; however, the over-representation of dominant species in these datasets may limit their applicability across diverse protein characteristics and properties. Furthermore, available algorithms for thermal stability prediction differ in their approaches. Some, like DeepTP [14] and BertThermo [15], focus on classification problems, distinguishing between thermostable and thermolabile proteins, without predicting T_m values directly. While the predictors that are based on classification have shown impressive performance, they ease the process by categorizing proteins into discrete stability classes, whereas regression-based models account for the continuous nature of T_m values.

AI based frameworks have been employed earlier in different bioinformatics related works [16]. In this paper, we propose a novel hybrid approach combining BiLSTM [17] and XGBoost [18] for predicting protein thermostability, and further compare it with other machine learning models. Our goal is to contribute to the growing body of knowledge in this field, with potential applications in protein engineering, drug discovery, and beyond. The pipelines implemented include BiLSTM network, Extreme Gradient Boosting (XGBoost), Random Forests [19], Support Vector Regressor (SVR) [20] and other baseline linear regression techniques including Ridge, Lasso, and standard Linear regression. To gain deeper insights into the proposed ensemble model's predictions and enhance interpretability for protein stability analysis, we employed Explainable Machine Learning (XML) techniques. This approach is crucial for domain experts who require a comprehensive understanding of the rationale behind individual predictions. In this study, we utilized SHAP, LIME, ELI5 and QLattice, to provide a more interpretable view of the model's decision-making process. The main contributions of this paper are as follows:

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

1. Proposed a BiLSTM-XGBoost ensemble model to capture complex protein sequence features and leverage the power of ensemble learning.
2. Conducted a comparative analysis with established machine learning models like Random Forest, SVR, linear regression techniques (Ridge and Lasso), and standard Linear regression.
3. Incorporated XML techniques like SHAP, LIME, ELI5 and QLattice to enhance model interpretability.

2 Materials and methods

This section explores various techniques and algorithms employed for protein thermostability as depicted in Fig. 1. We conducted a comparative analysis among eight regression models including BiLSTM-XGBoost, BiLSTM, XGBoost, Random Forest, SVR, Ridge and Lasso regressions, and Linear regression. The BiLSTM-XGBoost ensemble model proved to be the best algorithm among all the other methods.

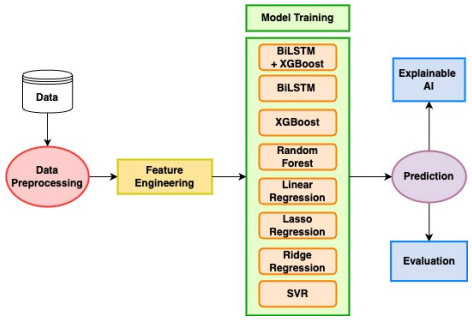


Fig. 1: The overall process flow for protein thermostability prediction

2.1 Dataset description

This research utilised the dataset from the Novozymes enzyme challenge [21]. Encompassing 31,390 protein sequences drawn from published studies and diverse sources, it offers a powerful resource for our investigation. The dataset is provided in CSV format and includes following key elements for predicting thermostability:

- **Protein sequences**: The primary structural information of proteins, essential for determining their properties and behavior.
- **pH value**: Environmental conditions under which the protein operates, influencing its stability.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

- **Melting temperature (T_m)**: The target variable indicating thermostability, where higher values reflect greater thermal stability (see Fig. 2 for distribution).
- **Rich feature set**: Beyond sequences, we employed feature engineering techniques to enrich the dataset with additional features relevant to protein stability. This includes amino acid composition (excluding uncommon residues): Alanine (Ala, A), Arginine (Arg, R), Asparagine (Asn, N), Aspartic acid (Asp, D), Cysteine (Cys, C), Glutamic acid (Glu, E), Glutamine (Gln, Q), Glycine (Gly, G), Histidine (His, H), Isoleucine (Ile, I), Leucine (Leu, L), Lysine (Lys, K), Methionine (Met, M), Phenylalanine (Phe, F), Proline (Pro, P), Serine (Ser, S), Threonine (Thr, T), Tryptophan (Trp, W), Tyrosine (Tyr, Y), Valine (Val, V); and a comprehensive array of physiochemical properties such as aromaticity, hydrophobicity, isoelectric point, instability index, molecular weight, and net charge.

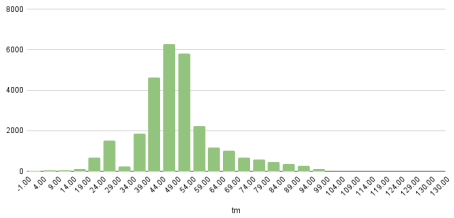


Fig. 2: Range of temperature (T_m) in the data

2.2 Preprocessing

To ensure data quality and relevance, we meticulously removed duplicates, imputed missing pH values using the mean, and calculated amino acid frequencies. Protein sequences were filtered based on length, retaining only those with a maximum length of $<= 221$, which covers the majority of the dataset and ensures model consistency. Sequences were encoded using one-hot encoding. Additionally, outlier checks were performed on the pH values, and any entries with pH > 14 were removed, as they fall outside the valid biological range. This resulted in a total of 30,965 protein sequences with a balanced training set (80%), validation set (10%), and test set (10%), each containing 24,772, 3096, and 3097 diverse protein sequences respectively. This curated dataset forms the basis for predicting protein thermostability.

2.3 Model development

In this section, we detail the implementation of our proposed model for protein thermostability. We also show the development of other models based on

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

the state-of-the-art bagging and boosting techniques that were considered while building our model.

Bidirectional LSTM The proposed model integrates a BiLSTM network to effectively capture contextual dependencies within protein sequences while incorporating additional non-sequential features for improved predictive performance. Fig. 3 illustrates the architecture of the model. The specific hyperparameters used in this work are detailed in Table 1. The architecture comprises three distinct input pathways: (i) a sequence input processed through an embedding layer followed by two stacked BiLSTM layers to encode the sequential dependencies, (ii) a feature-based input processed via dense layers to extract high-dimensional representations of supplementary attributes, and (iii) an amino acid count input passed through dense layers to capture specific feature interactions. Each branch applies intermediate layers of batch normalization to improve training stability and convergence. The outputs from the three branches are concatenated to form a unified representation, which is subsequently refined through additional dense layers. The final dense layer produces a scalar output, making the model well-suited for regression tasks such as predicting protein-related properties. This multi-input, multi-branch design enables the model to enhance its predictive performance.

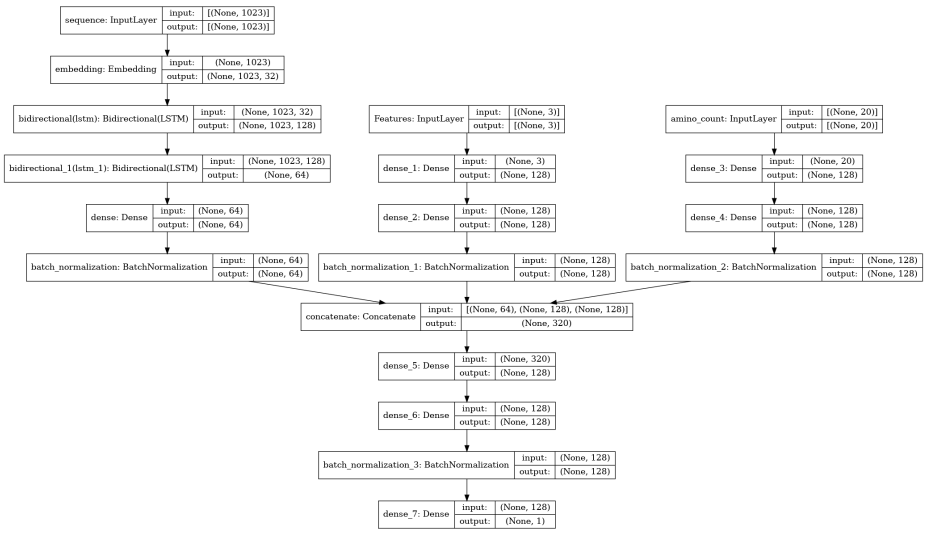


Fig. 3: The architecture of Bidirectional LSTM Model

XGBoost For T_m prediction, we developed a specific model based on the XGBoost algorithm. We created five distinct XGBoost models by varying the ran-

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

dom seed values used in dataset creation, resulting in unique subsets for training and validation. This approach allowed us to explore the robustness of our XGBoost model across various dataset instances. For the training of the XGBoost models, we configured specific hyperparameters, including a learning rate of 0.1, a maximum depth of 20, and 250 estimators, as detailed in Table 1. After training each XGBoost model, we evaluated their accuracy through validation and testing on independent test data. Fig. 4 illustrates the architecture of our XGBoost ensemble model.

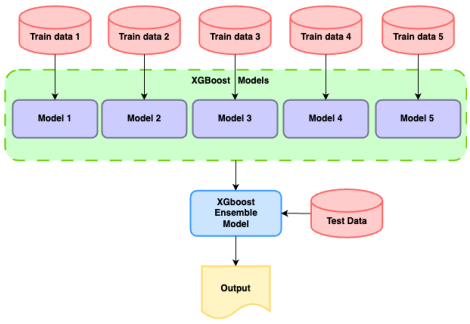


Fig. 4: XGBoost Model

Table 1: Hyperparameters of the models

Model	Hyperparameter
BiLSTM	'loss': mse, 'n_epochs': 200, 'batch_size': 256, 'LSTM_units: 64,32, 'regularisation': L1(1e-5), L2(1e-4), 'optimizer': adam, 'activation_funtion': selu, tanh, 'metrics': RMSE
XGBoost	'learning_rate': 0.1, 'max_depth': 20, 'n_estimators': 250, 'tree_method': 'gpu_hist'
Random Forest	'n_estimators': 1024, 'max_depth': 256, 'min_samples_split': 20, 'min_samples_leaf': 1, 'max_features': sqrt
SVR	'C': 1.0, 'kernel_function': 'RBF'
Lasso Regression	'alpha': 1.0
Ridge Regression	'alpha': 1.0

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

Ensemble Model (BiLSTM & XGBoost) We constructed an ensemble model combining BiLSTM and XGBoost to leverage the strengths of both sequential and gradient boosting approaches, as shown in Fig. 5a. BiLSTM captures long-range dependencies in protein sequences, essential for thermal stability, while XGBoost, a powerful gradient boosting algorithm, provides robustness. We used an ensemble of five XGBoost models to generate diverse training and validation sets, improving performance across varying data distributions. The final prediction was obtained by computing a weighted average of BiLSTM and XGBoost outputs, each assigned a weight of 0.5. This integration balances both models, improving accuracy and generalizability in protein T_m prediction.

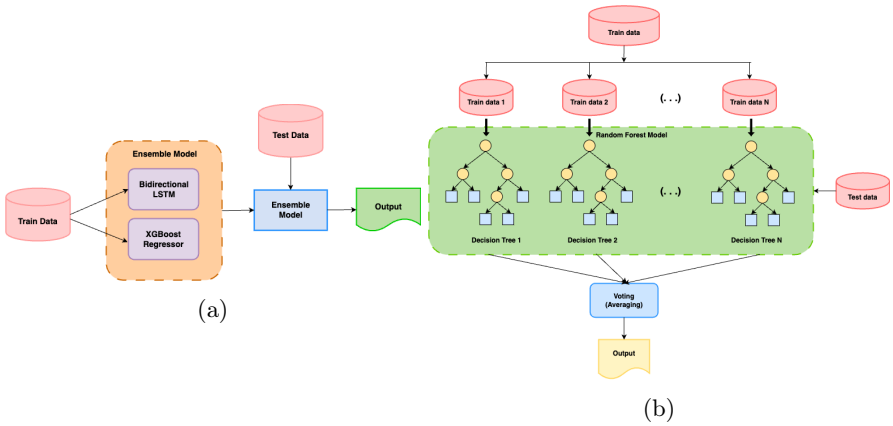


Fig. 5: (a) BiLSTM-XGBoost Ensemble Model (b) Random Forest Model

Random forest In addition to gradient boosting models, we explored the application of a Random Forest model within our enzyme stability prediction framework. Random Forest is a robust ensemble learning technique that constructs a collection of decision trees, each with a random subset of the features and data used for training. The hyperparameters of the model are shown in Table 1. Fig. 5b illustrates the architecture of our Random Forest model.

Linear Regression We additionally investigated the use of linear regression for protein thermostability prediction. This method establishes a linear relationship between protein features and T_m. While linear regression offers simplicity and interpretability, it might not capture the complex non-linear relationships often present in protein sequence-structure-stability relationships. The results from the linear regression model served as a baseline for comparison with the more complex models.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

Lasso Regression We further explored LASSO (Least Absolute Shrinkage and Selector Operator) regression for protein T_m prediction. By adding a penalty term, LASSO shrinks some feature coefficients to zero, enabling feature selection and improving interpretability by highlighting key contributors to thermostability. Hyperparameters are listed in Table 1.

Ridge Regression While similar to linear regression, Ridge regression applies an L2 penalty during training. This penalty shrinks the coefficients of all features, but to a lesser extent than the L1 penalty in Lasso regression. This can help address the issue of multicollinearity, where features are highly correlated, by preventing overly large coefficients and potentially improving model stability. The hyperparameters of the model are shown in Table 1.

Support Vector Regressor We also evaluated a Support Vector Regressor (SVR) for protein thermostability prediction. SVRs are effective at modeling non-linear relationships using high-dimensional feature spaces. We used an radial basis function (RBF) kernel to capture non-linear patterns between protein features and Tm, and optimized the model via grid search. Hyperparameters are listed in Table 1.

3 Results

3.1 Experimental setup

All the algorithms were implemented in Python (v.3.10.8). The model was trained and tested on a compute node with 2.2 GHz Intel-Xeon 2 vCPUs with 13GB RAM and NVIDIA Tesla K80 with 12GB RAM. To visualize data, Plotly and the Plotly.NET library (v.3.0.0) [22] were used. We have used XGBoost (v.1.7.4), LightGBM (v.3.3.5) and CatBoost (v.1.2) in our framework.

3.2 Performance metrics

We evaluated our model using performance metrics, including root mean squared error (RMSE), mean absolute error (MAE), R-squared (R2) score, and Spearman correlation.

1. **Root Mean Squared Error (RMSE)**: The square root of the average of the squared differences between predicted and actual values. A lower RMSE indicates better predictive accuracy.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y_i})^2}{n}}$$

2. **Mean Absolute Error (MAE)**: It computes the average absolute differences between predicted and actual values. A lower MAE signifies better model performance.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

$$\text{MAE} = \frac{\sum_{i=1}^{n} \|y_i - \hat{y_i}\|}{n}$$

3. **R-squared ($\mathbf{R^2}$) Score**: The proportion of the variance in the dependent variable is explained by the independent variables in a regression model. It ranges between 0 and 1, where higher values indicate better performance.

$$\mathbf{R^2} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y_i})^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}$$

4. **Spearman Correlation Coefficient**: It evaluates the strength and direction of a monotonic relationship between two variables. It captures non-linear relationships and is suitable for assessing associations between non-normally distributed variables. $\rho = \frac{\sum_{i=1}^{n}(s_i - \overline{s_i})(t_i - \overline{t_i})}{\sqrt{\sum_{i=1}^{n}(s_i - \overline{s_i})^2 \sum_{i=1}^{n}(t_i - \overline{t_i})^2}}$
where; n is the number of samples, y_i is the actual value of the i-th sample, $\hat{y_i}$ is the predicted value of the i-th sample, $\overline{y}$ is the mean of the actual values, s_i is the rank of the i-th actual value, t_i is the rank of the i-th predicted value, $\overline{s_i}$ is the mean of the ranks of the actual values, and $\overline{t_i}$ is the mean of the ranks of the predicted values.

Lower RMSE and MAE indicate greater accuracy, while higher R2 and Spearman coefficients signify a stronger correlation between predicted and actual values.

3.3 Comparative analysis

To thoroughly assess our approach, we conducted a detailed comparative analysis with all the models using multiple evaluation metrics including RMSE, MAE, R2 score and Spearman Correlation coefficient. There is no previous work that has worked on the same dataset. Therefore, a direct comparison with previous works is not possible, given the different inputs and goals. The performance of

Table 2: Performance of all models on the test set

Model	RMSE	MAE	R2	Spearman
Ensemble Model (BiLSTM-XGBoost)	**0.37**	**0.68**	**0.72**	**0.76**
Bidirectional LSTM	0.46	0.81	0.69	0.73
XGBoost	1.02	1.25	0.64	0.69
Random Forest	3.75	4.31	0.41	0.44
Linear Regression	5.12	5.33	0.36	0.39
Support Vector Regressor	5.27	5.46	0.36	0.38
Ridge Regression	5.33	5.81	0.34	0.35
Lasso Regression	5.96	6.13	0.33	0.35

all models on the test set is summarized in Table 2. BiLSTM-XGBoost ensemble model achieved the best overall performance with an RMSE of 0.37 and MAE

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

of 0.68. Fig. 6a indicates the loss curve, and Fig. 6b indicates the actual vs. predicted values.

In terms of RMSE, BiLSTM-XGBoost ensemble model achieved the best performance, with a RMSE of 0.37, followed by BiLSTM with an RMSE of 0.46. The Random Forest model had a RMSE of 3.75, significantly higher than the previous models. The performance of the SVR, Ridge Regression, Lasso Regression, and Linear Regression were all similar, with RMSEs ranging from 5.12 to 5.96.

Focusing on the MAE scores, we observe a similar trend to the RMSE analysis. BiLSTM-XGBoost achieved the best MAE of 0.68. BiLSTM followed closely with an MAE of 0.81, proving its ability to capture relevant sequence information. XGBoost has 1.25 MAE followed by Random Forest. The linear regression models (MAE: 5.33 - 6.13) exhibited significantly higher MAE values.

Examining the R2 scores, BiLSTM-XGBoost model achieved the highest R2 score (0.72). BiLSTM also showed a strong performance (0.69), while XGBoost achieved (0.64) R2 score. Linear regression models (0.36 - 0.33) exhibited lower R2 values, capturing a smaller proportion of the variance in melting temperatures.

The Spearman Correlation coefficients in Table 2 further reinforce the performance trends. BiLSTM-XGBoost ensemble model achieved the highest value (0.76), closely followed by BiLSTM and XGBoost. The other models like Random Forest (0.44) and linear regression approaches (0.39 - 0.35) displayed lower Spearman Correlation values, implying a weaker ability to capture the monotonic trends in protein thermostability.

(a) (b)

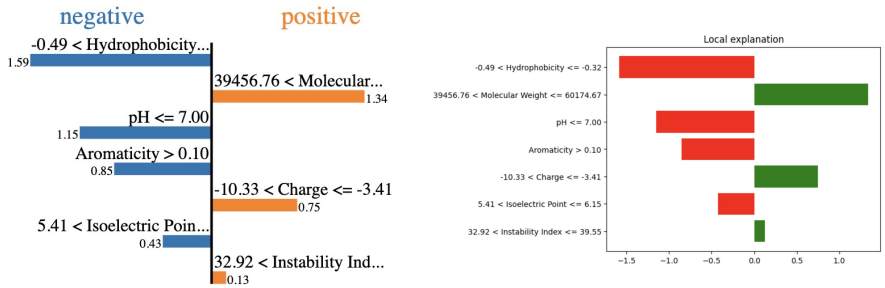
Fig. 6: (a) Loss Curve for BiLSTM-XGBoost Ensemble Model. (b) Scatter plot of predicted vs. actual protein temperatures for BiLSTM-XGBoost Ensemble Model.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

3.4 Explainable artificial intelligence

We incorporated XAI techniques to enhance the interpretability of our model's decision-making process. This allows us to gain deeper insights into the factors influencing the model's predictions and fosters trust in its reliability. We have deployed tools like LIME, SHAP, ELI5, and Qlattice on our best pipeline: ensemble of BiLSTM & XGBoost.

Local interpretable model-agnostic explanations LIME provides interpretable explanations by locally approximating a complex model's behavior using a simple surrogate. We applied LIME to the XGBoost component of our ensemble, leveraging its suitability for tabular features. A linear regression model was used as the surrogate, and local fidelity was validated by ensuring close alignment between its predictions and those of XGBoost on perturbed samples. Orange and blue colours denote positive and negative feature impacts, respectively. Fig. 7a illustrates a local prediction for protein T_m. The plot 7b, depicts feature contributions to the prediction. Bar sizes represent influence magnitude, with green indicating positive and red negative contributions.



(a) (b)

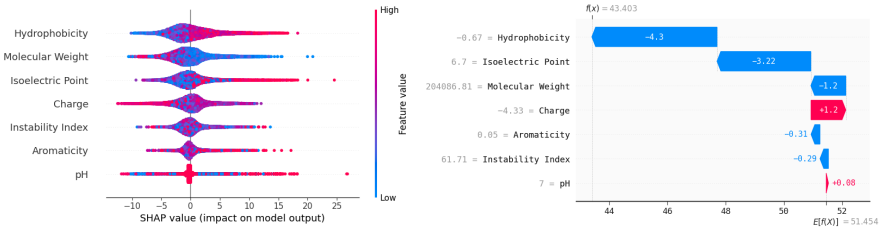
Fig. 7: (a) Prediction probability chart (b) Individual parameter contribution plot of a sample protein sequence

SHapley additive explanations (SHAP) SHAP values offer a game-theoretic method to explain individual predictions and feature importance in the model. The visualizations provide insights into feature contributions for specific data points and global feature importance across the dataset.

– **SHAP Beeswarm Plot**: It provides an overview of each feature's impact on the target variable, with features ranked by average SHAP value. Each dot represents a protein sequence, colored red for high and blue for low feature values. Fig. 8a shows that Hydrophobicity, Molecular Weight, and Isoelectric Point have the greatest influence. Higher Hydrophobicity and Isoelectric

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

Point increase SHAP values, while higher Molecular Weight decreases them.



(a) (b)

Fig. 8: (a) Shap Beeswarm Plot (b) Shap Waterfall Plot

– **SHAP Waterfall Plot**: This plot explains the predicted T_m for a specific protein by showing how each feature shifts the baseline prediction. Red bars indicate positive, and blue negative contributions. The Y-axis lists features with parameter values in grey. Fig. 8b shows Hydrophobicity and Isoelectric Point as major factors, with Charge and pH contributing slightly but positively to thermostability.

– **SHAP Force Plot**: This plot provides a localized explanation of feature contributions for a specific protein. Red features increase, and blue features decrease the predicted Tm (50.50), with bar size indicating impact. Fig. 9 highlights Hydrophobicity and Instability Index as major contributors, with red features contributing to a higher predicted melting temperature.

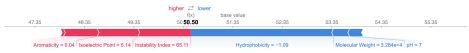
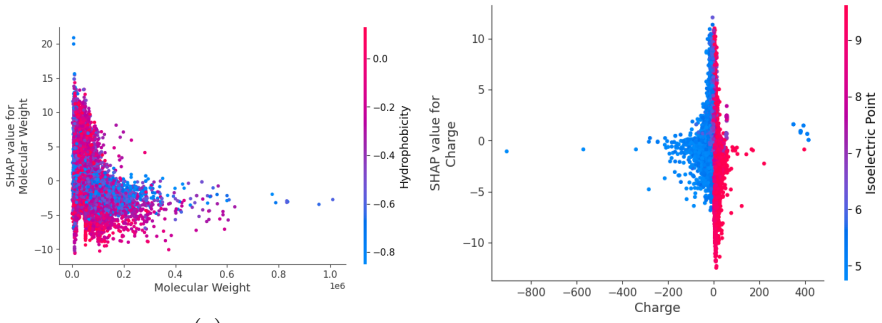


Fig. 9: SHAP Force Plot displaying how features push the Tm prediction higher or lower for a single sequence

– **SHAP Dependence Plot**: This plot shows the relationship between two features, with the X-axis for feature values, the Y-axis for SHAP values, and colors representing a second feature. Fig. 10a indicates the relation between Molecular Weight and Hydrophobicity. Higher Hydrophobicity values corresponds to lower Molecular Weight values. Fig. 10b indicates the relation between Charge and Isoelectric Point, showing that proteins with higher Isoelectric Points have a positive charge, while those with lower points have a negative charge.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22



(a) (b)

Fig. 10: SHAP Dependence plots: (a) Indicates the relation between the Molecular Weight and Hydrophobicity (b) Indicates the relation between Charge and Isoelectric Point.

ELI5 ELI5 is a Python library that helps interpret complex machine learning models by providing global and local explanations. Fig. 11a shows a visualization of feature importance, with pH having the strongest influence. Fig. 11b illustrates a single protein T_m prediction, highlighting Hydrophobicity and Aromaticity as the most impactful features, while Isoelectric Point has the least effect in this case.

Weight	Feature
0.2410	pH
0.1925	Charge
0.1621	Isoelectric Point
0.1335	Hydrophobicity
0.1134	Molecular Weight
0.0987	Instability Index
0.0589	Aromaticity

(a)

y (score **24.338**) top features

Contribution[?]	Feature	Value
+9.660	Hydrophobicity	0.150
+5.552	Aromaticity	0.060
+5.414	Molecular Weight	36320.720
+4.829	Charge	-1.870
-0.000	<BIAS>	1.000
-0.001	pH	7.000
-0.327	Instability Index	28.390
-0.789	Isoelectric Point	6.110

(b)

Fig. 11: (a) Depicts the feature weights (b) Indicates individual protein-wise prediction

Qlattice QLattice is a Python-based library used for exploring feature spaces and selecting optimal models through supervised learning, with support from the Feyn library. It handles both categorical and numerical data and generates visualizations that reveal mathematical relationships behind model predictions.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

This visual method, known as QGraphs, clarifies the features and operations used during model development. Fig. 12 shows that Hydrophobicity, Molecular Weight, and Instability Index are treated as independent inputs, with the model leveraging their interactions to predict Tm.

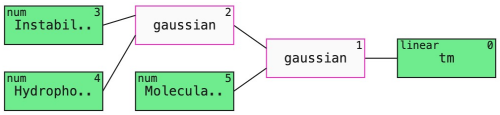


Fig. 12: QGraph visualizing selected features and their relationships

4 Conclusion

This study evaluated various machine learning models for protein thermostability prediction, with the BiLSTM-XGBoost ensemble showing superior performance across multiple metrics. Our results highlight the importance of amino acid composition, physiochemical properties, and engineered features in estimating protein melting temperatures. However, the model's reliance on sequence-based features limits its applicability to well-annotated proteins and excludes structural insights like 3D conformations and spatial interactions—key determinants of stability. The proposed framework can be practically useful in bioengineering workflows. For example, it can help prioritize thermostable enzyme candidates during optimization, reducing the need for extensive lab testing. It may also assist in selecting enzymes that perform reliably at high temperatures for industrial applications like biocatalysis.

Future work should explore advanced deep learning architectures like transformers and graph neural networks to capture spatial and contextual features. Incorporating 3D structures, post-translational modifications, and evolutionary conservation, along with mutation-aware models and transfer learning, could greatly enhance accuracy and broaden applicability in protein design.

References

1. S. Frokjaer and D. E. Otzen, "Protein drug stability: a formulation challenge," *Nature reviews drug discovery*, vol. 4, no. 4, pp. 298–306, 2005.
2. P. Almeida, *Proteins: concepts in biochemistry*. Garland Science, 2016.
3. P. Leuenberger, S. Ganscha, A. Kahraman, V. Cappelletti, P. J. Boersema, C. von Mering, M. Claassen, and P. Picotti, "Cell-wide analysis of protein thermal unfolding reveals determinants of thermostability," *Science*, vol. 355, no. 6327, p. eaai7825, 2017.
4. C. Vieille and G. J. Zeikus, "Hyperthermophilic enzymes: sources, uses, and molecular mechanisms for thermostability," *Microbiology and molecular biology reviews*, vol. 65, no. 1, pp. 1–43, 2001.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22

5. M. Gorania, H. Seker, and P. I. Haris, "Predicting a protein's melting temperature from its amino acid sequence," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. IEEE, 2010, pp. 1820–1823.
6. M. Ebrahimi, A. Lakizadeh, P. Agha-Golzadeh, E. Ebrahimie, and M. Ebrahimi, "Prediction of thermostability from amino acid attributes by combination of clustering with attribute weighting: a new vista in engineering enzymes," *PloS one*, vol. 6, no. 8, p. e23146, 2011.
7. A. D. Robertson and K. P. Murphy, "Protein structure and the energetics of protein stability," *Chemical reviews*, vol. 97, no. 5, pp. 1251–1268, 1997.
8. F. Pucci, M. Dhanani, Y. Dehouck, and M. Rooman, "Protein thermostability prediction within homologous families using temperature-dependent statistical potentials," *PloS one*, vol. 9, no. 3, p. e91659, 2014.
9. Y. Dehouck, B. Folch, and M. Rooman, "Revisiting the correlation between proteins' thermoresistance and organisms' thermophilicity," *Protein engineering, design & selection*, vol. 21, no. 4, pp. 275–278, 2008.
10. J. Horne and D. Shukla, "Recent advances in machine learning variant effect prediction tools for protein engineering," *Industrial & engineering chemistry research*, vol. 61, no. 19, pp. 6235–6245, 2022.
11. P. Braiuca, A. Buthe, C. Ebert, P. Linda, and L. Gardossi, "Volsurf computational method applied to the prediction of stability of thermostable enzymes," *Biotechnology Journal: Healthcare Nutrition Technology*, vol. 2, no. 2, pp. 214–220, 2007.
12. F. Pucci, J. M. Kwasigroch, and M. Rooman, "Scoop: an accurate and fast predictor of protein stability curves as a function of temperature," *Bioinformatics*, vol. 33, no. 21, pp. 3415–3422, 2017.
13. Y. Yang, X. Ding, G. Zhu, A. Niroula, Q. Lv, and M. Vihinen, "Protstab–predictor for cellular protein stability," *BMC genomics*, vol. 20, no. 1, pp. 1–9, 2019.
14. J. Zhao, W. Yan, and Y. Yang, "Deeptp: A deep learning model for thermophilic protein prediction," *International Journal of Molecular Sciences*, vol. 24, no. 3, p. 2217, 2023.
15. H. Pei, J. Li, S. Ma, J. Jiang, M. Li, Q. Zou, and Z. Lv, "Identification of thermophilic proteins based on sequence-based bidirectional representations from transformer-embedding features," *Applied Sciences*, vol. 13, no. 5, p. 2858, 2023.
16. W. Hu, Y. Liu, X. Chen, W. Chai, H. Chen, H. Wang, and G. Wang, "Deep learning methods for small molecule drug discovery: A survey," *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 2, pp. 459–479, 2024.
17. M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
18. T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou *et al.*, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, vol. 1, no. 4, pp. 1–4, 2015.
19. Y. Qi, "Random forest for bioinformatics," *Ensemble machine learning: Methods and applications*, pp. 307–323, 2012.
20. R. G. Brereton and G. R. Lloyd, "Support vector machines for classification and regression," *Analyst*, vol. 135, no. 2, pp. 230–267, 2010.
21. D. Pultz, E. Friis, J. Salomon, P. F. Hallin, S. B. Jørgensen, W. Reade, and M. Demkin, "Novozymes enzyme stability prediction," https://kaggle.com/competitions/novozymes-enzyme-stability-prediction, 2022, kaggle.
22. K. Schneider, B. Venn, and T. Mühlhaus, "Plotly. net: A fully featured charting library for. net programming languages," *F1000Research*, vol. 11, p. 1094, 2022.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97554-7_22