Using Protein Language Models to Predict the Effects of Variants on Thermal Stability

Arnav Gupta & Franco Viscarra

Received June 01, 2024
Accepted November 14, 2024
Electronic access November 30, 2024

The melting point of a protein characterizes its thermal stability, a critical factor in determining its suitability for various applications. For example, *Ideonella sakaiensis* produces a plastic-degrading enzyme, polyethylene terephthalate hydrolase, capable of breaking down polyethylene terephthalate (PET). However, the enzyme's low thermal stability limits its potential for industrial use. To address this, we fine-tuned a pre-trained protein language model to predict thermal stability based on protein sequence and identified mutations to improve stability. Two mutant proteins were designed and evaluated using molecular dynamics simulations. While the simulations revealed that the mutated proteins were less stable than the wild-type, this outcome highlights the complexity of accurately predicting mutation effects on protein stability using sequence-based models. Our findings underscore the need for further refinement of computational tools and the need for more comprehensive datasets to enhance protein engineering for industrial applications.

Keywords: Protein Language Model, Protein Engineering, *Ideonella sakaiensis* PETase, Thermal stability

Introduction

Polyethylene terephthalate (PET) is a polymer made from the polymerization reaction of ethylene glycol and dimethyl terephthalate, and it is one of the most common single-use plastics due to its low cost and strength[1]. Additionally, because PET is not biodegradable, taking years to decompose[2], its buildup in landfills and as litter poses a threat to the environment[3]. This is due to the fact that as PET accumulates, it leads to the production of greenhouse gases like CO2 and other organic compounds that accelerate global warming, most commonly methane[4]. Although recycling of PET is possible through heat reprocessing, this alters the desirable mechanical properties of PET. Furthermore, chemical recycling is not viable because of its high cost[5].

Ideonella Sakaiensis, which was discovered in 2016 by Yoshida and collaborators[6], is a bacterium that can use PET as a carbon source. The enzyme responsible for PET degradation, PETase (*Is*PETase), has been found to be more efficient and stable than other enzymes that can degrade PET[6]. However, the PETase enzyme does not have a very high thermal stability, with a melting temperature (T_m) of 45°C, which makes it lose most of its activity within 24h at 37°C. This poses a problem for it to be effectively used in PET degradation at an industrial scale7. Because of this, many scientists have researched rational protein engineering techniques to increase thermal stability[7].

Current research focuses on laboratory methods for either improving the efficiency of the enzyme or its stability. Most forms of research for making thermostable *Is*PETase involve protein engineering techniques with replacing amino acid residues to increase the number of contacts within the protein, via hydrogen bonding, in order to increase the T_m[8] or post-translational modifications like glycosylation to improve the thermal stability[9].

In this research paper, we aim to predict point mutations that can improve the thermal stability (TS) for *I. Sakaiensis* PETase using a pre-trained machine learning (ML) model and molecular dynamic (MD) to assess the effects of these variations in silico. The advantage of doing it computationally is that we can use MD to control the environmental conditions without incurring the costs and extensive work required for setting up mutagenesis and thermal stability analysis in the laboratory. Although researchers have used computational methods to predict the TS of the protein, due to the high structural complexity of enzymes like *Is*PETase, these methods struggle to predict melting points accurately. For example, Yang et al proposed ProTstab2[10], a ML model based on substantial calculations to collect thousands of protein features that will be used to train the algorithm, this increases the complexity of the model and makes it prone to overfitting. Additionally, other methods have been tried to predict protein TS, including analyzing the protein length and hydrogen bonding[11,12], but they perform worse when trying to predict the effect of single amino acid substitutions. This occurs due to a lack of experimental data, since many factors that may influence protein stability, no matter how subtle, will be difficult to detect unless many samples have been analyzed. Other approaches include the use of the state-of-the-art structure-prediction-model AlphaFold to predict protein stability, with little success, mainly because AlphaFold is primarily designed for predicting protein structures rather than accurately modelling the subtle effects of mutations on stability and since it was trained on a database of experimentally resolved structures suffers from a lack of diversity on its training dataset [13]. To overcome these challenges, we propose the fine-tuning of a transformer protein language model developed by Facebook research [14,15] that has been pre-trained on hundreds of millions of unlabelled protein sequences in an unsupervised manner, so it can learn the fundamental properties of proteins by itself. Afterwards, by running MD simulations on wild-type and mutant proteins, we can test our predictions on data from the ML model by seeing if the mutant protein shows an increase in the compared stability.

Methods

MD Simulations

Molecular Dynamics (MD) simulations were used to assess the TS of our enzyme. The system was parameterized using the Amber99SB-ILDN [16] force field for proteins, solvated with the TIP3P water model [17] and neutralised in 0.15 M of NaCl. In order to compare the stability of the protein two simulations, systems were run at the temperatures of 300K and 400K for 100ns. All simulations were carried out with GROMACS 2021 [18]. These simulations were used to calculate the Q-value, which gives information on how much of the protein structure is retained after the simulation time. This was calculated using the soft cut function described by Best et al [19] and implemented in MDAnalysis [20]:

$$Q(r, r_0) = \frac{1}{1 + e^{\beta(r - \lambda r_0)}} \tag{1}$$

where r and r_0 are the contact distances at time t and time $t = 0$, respectively. β is the softness of the switching function, and λ is the reference distance tolerance, set to 5 Å^{-1} and 1.8 Å, respectively.

The Q-values were calculated for all inter- and intra-combinations of three groups:

- **Hydrophilic:** D, E, Q, N, R, K, H
- **Hydrophobic:** F, Y, W, L, V, I, M, C, P
- **Small:** G, A, S, T

Evidence shows that the average Q-value over MD simulations correlates well with protein T_m when run at a higher temperature, specifically at 400K. Furthermore, this correlation is best when the Q-value is calculated for the hydrophilic versus all residues [21].

Protein language model fine-tuning

We used a pre-trained transformer protein language model (PLM) ESM2 of 35 million parameters. The model consists of a modified BERT-like encoder transformer architecture with 12 layers with 20 attention heads each [15]. We fine-tuned the model using the Meltome atlas database of protein melting temperatures [22]. The finetuning was run using the Kaggle notebook platform with GPU acceleration (https://www.kaggle.com). Due to memory limitations, the sequences longer than 512 amino acids were omitted from the database leaving a dataset of 22971 proteins. This dataset was split into train, validation, and test sets (80%, 10%, and 10% of the total database respectively). The model was trained for 10 epochs, and the best model was selected at the end of training using the validation loss as a metric. Additionally, mean absolute error (MAE), mean square error (MSE), and R^2 metrics were calculated to assess the accuracy of the regression and for comparison with similar methods. MSE was calculated by adding the squared errors and dividing by the number of data points then taking the square root (Eq. 2). Likewise, the process for MAE was done without the squaring and square rooting and just taking the absolute value of the average of the errors (Eq. 3). The R^2 error was calculated using the standard formula for finding the coefficient of determination (Eq. 4). The 95% confidence interval (CI) was calculated by resampling the test set using bootstrapping with 5000 iterations.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2 \tag{2}$$

$$\text{MAE} = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \tag{3}$$

$$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}} \tag{4}$$

Results & Discussion

Metric	Fine-tuned PLM (95% CI)	ProTstab2
MSE	47.59 (43.99-53.94)	82.752
MAE	5.08 (4.82-5.22)	6.934
R^2	0.63 (0.59-0.675)	0.580

Table 1 Comparison of metrics for the fine-tuned PLM and ProTstab2

We fine-tuned the best model for 10 epochs and after the third epoch the model's accuracy did not improve any further, at the end of training the best model was chosen based on the MAE.

We performed a deep mutational scan predicting the T_m of every possible single-point mutant, excluding the residues in the binding site (Y87, W159, S160, M161, W185, C203, D206, I208, H237, S238, C239, N241, $R^2$80) (10). and found that

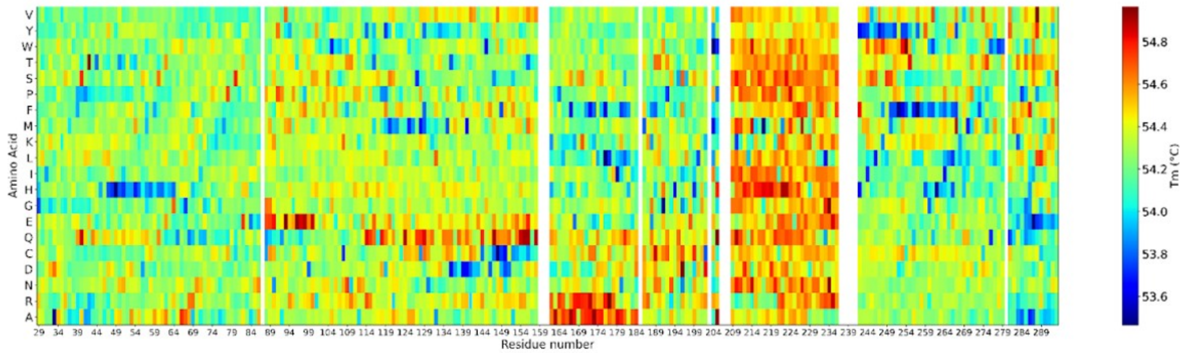

Fig. 1 Graph showing the melting temperatures of the various mutants along the protein sequence.

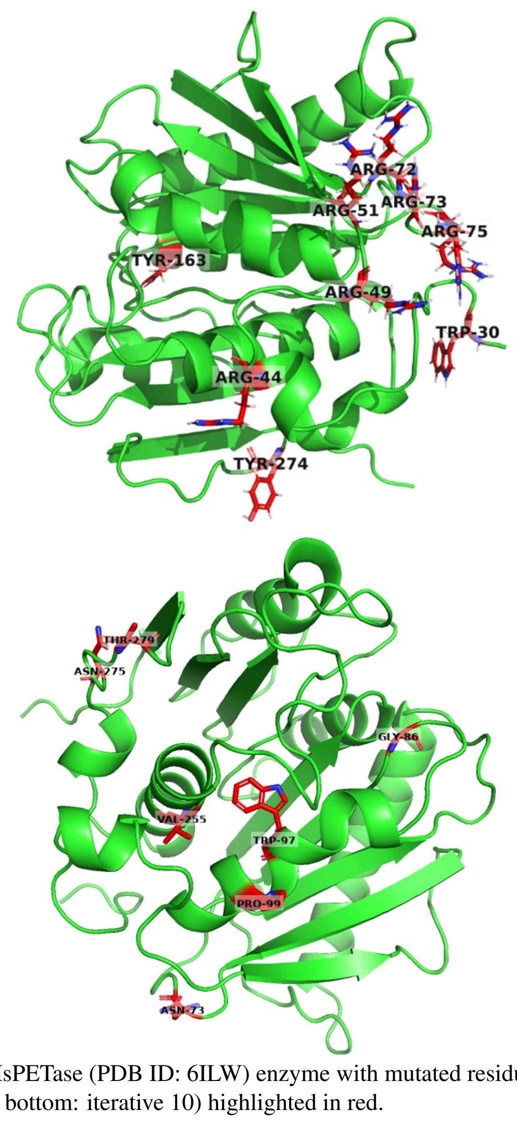

Fig. 2 IsPETase (PDB ID: 6ILW) enzyme with mutated residues (top: 10 best, bottom: iterative 10) highlighted in red.

even the best mutants had a very slight increase compared to the wild types. For the best mutant, our model predicted a T_m no more than 2°C higher than the T_m predicted for the wild-type protein (54.49°C vs 52.8°C respectively). We calculated the Q-values of two multiple mutants with ten modifications using molecular-dynamic simulations. One of the mutants was generated by taking the 10 best substitutions predicted by the model (N30W, E44R, P49R, T51R, T72R, N73R, G75R, G76R, G163Y, E274Y) and the other by iteratively modifying the protein 10 times (T73R, P86W, W97D, G99I, G255I, E275Y, S279C), the ML model was used to predict the most favorable single modification at each step. After applying each modification, the mutated protein was fed into the model to predict the next optimal modification. As shown in Table 2, the wild-type proteins showed a higher Q-value than the mutant protein between each combination of amino acid groups. The most contact preservation in both mutant proteins at 400 K, was between the small amino acids, which had a Q value of around 0.78 and 0.89 for the 10-best and iterative-10 mutants respectively. However, the Q value in the original wild-type protein was 0.89 on average, implying that no improvements were seen in the preservations in native contacts for the mutant protein over the wild-type protein.

Pre-trained protein language models have been successfully applied to several tasks, such as structure prediction, effect of variants on protein function, and prediction of binding residues, either with or without fine-tuning[23–28], and they seem able to generalise better for downstream, complex tasks if some extra data is provided for fine-tuning.

Compared to a previously published thermal stability machine learning model, which was trained on the same dataset (ProTstab2) (11), our model had a lower MSE and MAE, as well as a higher R^2 value, indicating greater accuracy and better correlation than the existing model on the test dataset.

The predicted mutants obtained from the PLM did not show any improvement in the TS over the wild-type protein. There are several potential reasons why the model failed to predict an improved protein. Namely, models trained using wild-type protein

	Wild-type	Mutant 10-best	Mutant iterative-10
hydrophobic-hydrophobic	0.86 ± 0.08	0.75 ± 0.13	0.85 ± 0.07
hydrophobic-small	0.87 ± 0.06	0.77 ± 0.10	0.86 ± 0.05
hydrophilic-hydrophobic	0.79 ± 0.07	0.68 ± 0.11	0.75 ± 0.06
hydrophilic-hydrophilic	0.82 ± 0.06	0.70 ± 0.11	0.79 ± 0.06
hydrophilic-small	0.80 ± 0.06	0.69 ± 0.11	0.76 ± 0.06
small-small	0.89 ± 0.06	0.78 ± 0.11	0.89 ± 0.06
all-hydrophobic	0.85 ± 0.06	0.76 ± 0.10	0.84 ± 0.05
all-hydrophilic	0.82 ± 0.05	0.73 ± 0.09	0.80 ± 0.05
all-small	0.85 ± 0.06	0.76 ± 0.09	0.84 ± 0.05
all-all	0.86 ± 0.07	0.74 ± 0.11	0.84 ± 0.07

Table 2 Q values for each type of contact between different amino acids in both the wild-type and mutants, plus-minus the standard deviation of the mean across all the simulations.

libraries, such as the one we used in this work, are designed to predict certain properties of a natural sequence, which does not align with the task of predicting the properties of an engineered protein. One example of this is a recent work using AlphaFold, a model trained to predict the structure of proteins, to predict the effect of point mutations on protein stability which resulted to be quite inaccurate (14). Recently, Chu et al[29] used a similar approach by finetuning the ESM2 protein language model but fine-tuned it to predict general stability (folding ΔΔG) for small protein domains, which generalized well for small proteins not in the dataset, being not as accurate with the larger ones like the enzyme used in this study.

Additionally, even if a model can predict the effect of a single-point mutation, this does not mean that the predicted effect of several mutations will be additive. For example, if mutation X and Y each improve the T_m of a protein, it does not mean that mutations X and Y used in combination will improve the T_m of a protein even more. This is because these amino acids can interact in non-trivial ways that are not captured by the algorithm31.

Additionally, when fine-tuning the model for 10 epochs we observed that after the third epoch, we did not see any improvement in the validation loss, but the training loss kept improving, which tells us that after that the model's goal changed from generalising the problem of predicting the sequence T_m to aligning the results as much as it can with the given data. This behaviour of the model overfitting after a few epochs shows that either the complexity of the model is too high, the dataset is too small, or more likely a combination of the two since more complex machine learning models need more data to learn properly[30].

Conclusion

Currently, the machine learning model used in our case had better accuracy than older models, however, we were unable to predict an improved mutant regarding the Q values. Furthermore, the melting temperature of the best-predicted mutants increased by only a few degrees C. This shows that, even though state-of-the-art protein language models can be used for several tasks on native proteins, they still lack the ability to successfully predict the effect of mutations, due to the limitations of their algorithm and the data they were trained on. Thus highlighting the importance of diverse datasets that accurately represent the properties of both wild-type and mutant proteins to achieve better predictions in protein engineering tasks.

References

1 K. Gopalakrishna and N. Reddy, *Regulations on Recycling PET Bottles*.

2 R. Muller, I. Kleeberg and W. Deckwer, *Biodegradation of polyesters containing aromatic constituents*.

3 S. Sharma and S. Chatterjee, *Microplastic pollution, a threat to marine ecosystem and human health: a short review*.

4 J. Diao, Y. Hu, Y. Tian, R. Carr and T. Moon, *Upcycling of poly(ethylene terephthalate) to produce high-value bio-products*.

5 K. Hiraga, I. Taniguchi, S. Yoshida, Y. Kimura and K. Oda, *Biodegradation of waste PET*.

6 S. Yoshida, K. Hiraga, T. Takehana, I. Taniguchi, H. Yamaji and Y. Maeda, *A bacterium that degrades and assimilates poly(ethylene terephthalate*.

7 H. Nguyen, P. Qi, M. Rostagno, A. Feteha and S. Miller, *The quest for high glass transition temperature bioplastics*.

8 H. Son, I. Cho, S. Joo, H. Seo, H. Sagong and S. Choi, *Rational Protein Engineering of Thermo-Stable PETase from Ideonella sakaiensis for Highly Efficient PET Degradation*.

9 X. Han, W. Liu, J. Huang, J. Ma, Y. Zheng and T. Ko, *Structural insight into catalytic mechanism of PET hydrolase*.

10 Y. Yang, *ProTstab2 for Prediction of Protein Thermal Stabilities*.

11 K. Ghosh and K. Dil, *Computing protein stabilities from their chain lengths*.

12 M. Prevost, Proceedings of the National Academy of Sciences of the United States of America.

13 M. Pak, *Using AlphaFold to predict the impact of single mutations on protein stability and function*.

14 A. Rives, Proceedings of the National Academy of Sciences of the United States of America.

15 Z. Lin, *Evolutionary-scale prediction of atomic-level protein structure with a language model*.

16 K. Lindorff-Larsen, *Improved side-chain torsion potentials for the Amber ff99SB protein force field*.

17 W. Jorgensen, J. Chandrasekhar, J. Madura and R. Impey, *Comparison of simple potential functions for simulating liquid water*.

18 M. Abraham, *GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers*.

19 R. Best, Proceedings of the National Academy of Sciences of the United States of America.

20 R. Gowers, *MDAnalysis: A Python Package for the Rapid Analysis of Molecular Dynamics Simulations*, http://conference.scipy.org.s3-website-us-east-1.amazonaws.com/proceedings/scipy2016/oliver_beckstein.hT_ml.

21 D. Kuroda, *Computer-aided antibody design*.

22 A. Jarzab, *Meltome atlas-thermal proteome stability across the tree of life*.

23 C. Hsu, R. Verkuil, J. Liu, Z. Lin, B. Hie and T. Sercu, *Learning inverse folding from millions of predicted structures. bioRxiv*.

24 K. Weissenow, *Protein language-model embeddings for fast, accurate, and alignment-free protein structure prediction*.

25 M. Heinzinger, *Contrastive learning on protein embeddings enlightens midnight zone*.

26 C. Marquet, *Embeddings from protein language models predict conservation and variant effects*.

27 M. Lit$T_m$ann, *Protein embeddings and deep learning predict binding residues for various ligand classes*.

28 H. Stärk, *Light attention predicts protein location from the language of life*.

29 S. Chu, *Protein stability prediction by fine-tuning a protein language model on a mega-scale dataset*.

30 N. Liu, S. Li, Y. Du, J. Tenenbaum and A. Torralba, *Learning to Compose Visual Relations*, arXiv.