Pretraining transformers for protein interaction

Teslim Olayiwola

Cain Department of Chemical Engineering, Louisiana State University, Baton Rouge, Louisiana 70803, United States.

Introduction

Proteins, the fundamental building blocks of life, present a unique challenge in understanding their functions despite the exponential growth of protein sequence data [1]. In this context, the Transformer architecture, originally designed for natural language processing and exemplified by models like BERT, has emerged as a powerful tool. These deep neural networks, pretrained on extensive datasets, have revolutionized automated text analysis across various domains.

Transfer learning, a core concept in this paradigm, involves pretraining models on one task and fine-tuning them for other related tasks [2]. The ability of transformer to be applied in downstream tasks such as classification where labels are mapped to specific languages has played a significant role in enabling models to learn from massive datasets [3]. While these techniques have yielded substantial performance gains in various fields, adapting them to the unique properties of proteins, such as their lack of clear-cut linguistic structures and variable lengths, remains a challenge. In this project, the impact of pretraining strategies on BERT model was studied.

Methodology

To train a language model effectively, a substantial dataset is essential. In this study, we leveraged the provided dataset comprising 130,471 examples for the pretraining phase. The model used for this purpose is the TCRModel, as defined in the model.py file, which follows the standard BERT transformer architecture (as depicted in Fig. X). In preparing the inputs for the transformer, the protein sequence was converted into numerical inputs namely input token ids and attention mask. Here, a python class *AntigenTCRDataset* in dataPreprocessing.py was created. Each training sample is a pair of antigen and tcr data from a file. The combined words are not consecutive and a [CLS] token prepended to the first sentence (i.e. antigen) and a [SEP] token appended to each sentence (as a separator). Then, the two sentences will be concatenated as a sequence of tokens to

form a [CLS] antigen [SEP] tcr [SEP] to become a training sample. The combined sequence is padded to the maximum length and the final sequence is tokenized based on a dictionary like the WordPiece [4] tokenizer algorithm and the encoding is as follows:

encodings = {"[PAD]": 0, "[CLS]": 1, "[SEP]": 2, "[MASK]": 3, "[UNK]": 4, "G": 5, "T": 6, "C": 7, "Q": 8, "N": 9, "W": 10, "E": 11, "I": 12, "Y": 13, "A": 14, "R": 15, "L": 16, "S": 17, "M": 18, "D": 19, "F": 20, "H": 21, "K": 22, "V": 23, "P": 24}

In the pretraining stage, two distinct approaches were explored. Firstly, we excluded the Next Sentence Prediction (NSP) stage typically found in BERT models. This decision stemmed from the nature of protein structure tasks, where predicting the next word in a sentence holds limited relevance. Instead, we trained the transformer model to treat each sequence as an independent document, obviating the need for NSP. A small percentage of the tokens in the training sample are masked with a special token [MASK] as contained in the masked language modeling protocol.

Secondly, we adopted a method inspired by the text in-filling approach of the BART model, incorporating token deletion and text in-filling from BART [5]. The encoder was provided with a corrupted version of the tokens, while the decoder received the original tokens with masking applied to conceal future words, akin to a standard transformer decoder. The pretraining tasks for the encoder encompassed a combination of transformations, including masking random tokens (similar to BERT), deleting random tokens, masking a span of k tokens using a single mask token (with a span of 0 tokens representing the insertion of a mask token), permuting sentences, and rotating the document to initiate at a specific token.

In the subsequent fine-tuning stage, whether starting from a randomly initialized model or a pretrained model, we thoroughly examined the dataset for potential class imbalance, ensuring that no specific label class dominated the dataset. Additionally, to mitigate overfitting, we employed a 3 k-fold cross-validation strategy to maintain model generalization during training. To evaluate the training schemes, in the pretraining stage, the loss of the model was computed. For the downstream classification task, the performance of the model was evaluated using the loss, accuracy score and area under the curve (AUC).

Result and Discussion

As shown in Fig. 1, The analysis of the dataset showed one with 130,471 samples, with a substantial skew towards the "no interaction" class, occurring at least five times out of every six samples. To rectify this imbalance issue, I resampled the dataset, resulting in a balanced dataset comprising 65,046 samples with even distribution as shown in Fig 1. This strategic resampling was pivotal to ensure unbiased model training. Subsequently, our model underwent rigorous evaluation and performance evaluation using metrics such loss, accuracy, and AUC.

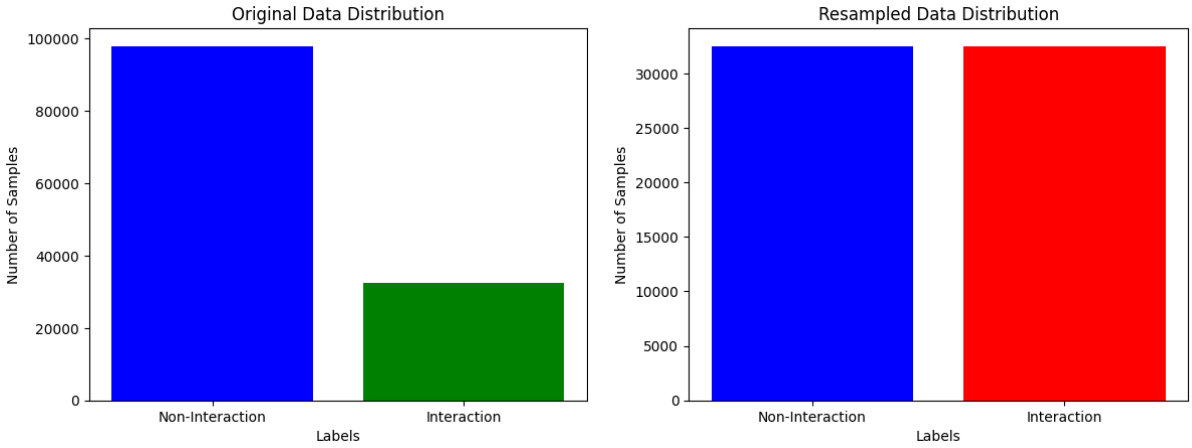

Fig. 1. Distribution of available dataset

In creating the classifier with a randomly initialized transformer, a classification model was created with an instance of the TCRModel. The model was trained with a k-fold cross-validation strategy with k set to 3 effectively prevented overfitting and bolstered model generalization. The result showed a predictive accuracy of 50% and AUC score of 0.5 for both the train and validation data as shown in Fig. 2. After completing the training, the trained model was tested with an unseen dataset and the model performance is same as the training model. Our study underscores the potential of domain-specific pretraining and fine-tuning in bioinformatics and the promise of deep learning models in elucidating complex biological systems.

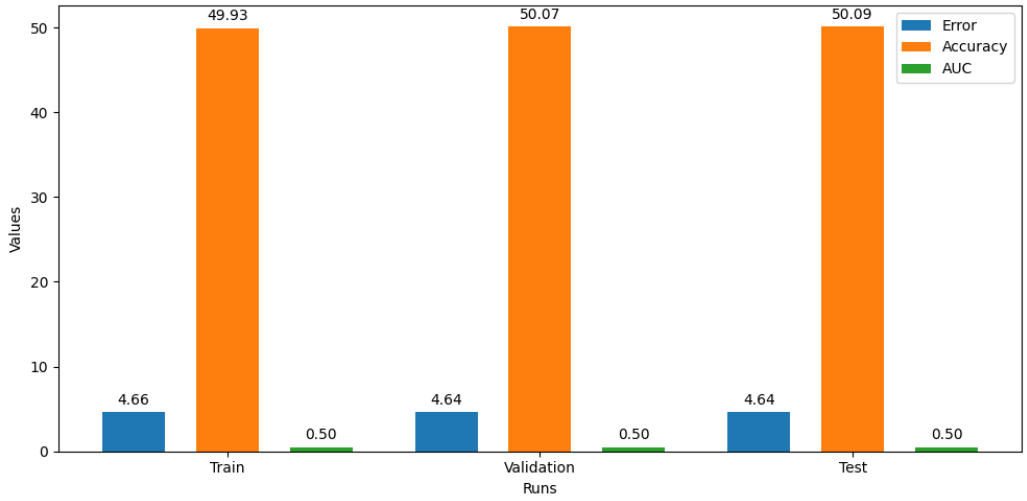

Fig. 2. Performance of base model

To illustrate the influence of pretraining on the classifier model's performance, we have encapsulated our findings in Fig. 3. Here, we distinguish between two distinct pretraining approaches: Approach 1, denoted as BERT, which involves traditional masked language modeling, and Approach 2, referred to as BART, which incorporates token deletion and text in-filling techniques. However, the results obtained from these two approaches did not exhibit a substantial enhancement in the model's performance. This outcome is somewhat disappointing, and we conjecture that the lack of significant improvement may be attributed to several factors. Firstly, it's possible that the complexity and uniqueness of protein sequences pose challenges that generic pretraining strategies may not adequately address. Unlike natural language, proteins lack explicit linguistic structures, making it challenging for models to capture meaningful representations effectively. Moreover, the limited size of the dataset might have constrained the potential benefits of pretraining. Additionally, the choice of pretraining tasks, despite being inspired by successful natural language models, may not have fully aligned with the intricacies of protein sequence analysis. Further investigations and refinements in pretraining strategies specifically tailored to protein sequences may be necessary to unlock their full potential.

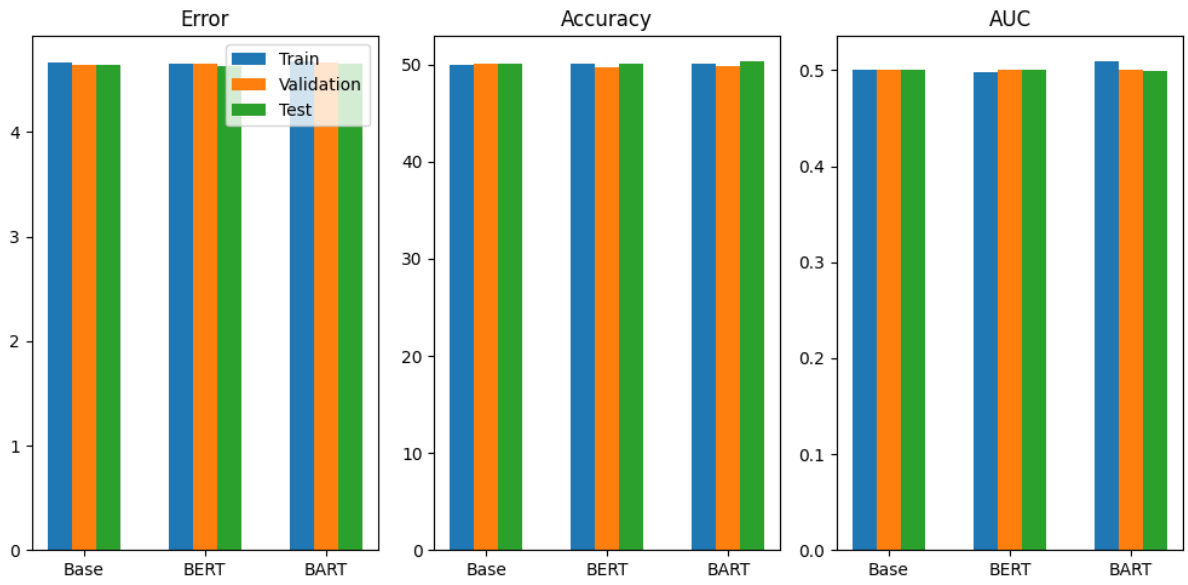

Fig. 3. Comparative performance of pretraining strategies on classification

Conclusion

In this project, we explored the adaptation of the Transformer architecture, originally designed for natural language processing, to the unique challenges posed by protein sequence data. While we investigated different pretraining strategies, our results revealed application of language modeling to uncover the importance of protein sequences. This study highlights the potential for further advancements in deep learning models to unlock new insights into the functions of proteins, paving the way for impactful applications in bioinformatics and healthcare.

References

[1] G. Caetano-Anollés, M. Wang, D. Caetano-Anollés, and J. E. Mittenthal, "The origin, evolution and structure of the protein world," *Biochem. J.*, vol. 417, no. 3, pp. 621–637, Jan. 2009, doi: 10.1042/BJ20082063.

[2] C. V. Theodoris *et al.*, "Transfer learning enables predictions in network biology," *Nature*, vol. 618, no. 7965, Art. no. 7965, Jun. 2023, doi: 10.1038/s41586-023-06139-9.

[3] W.-C. Chang, H.-F. Yu, K. Zhong, Y. Yang, and I. S. Dhillon, "Taming Pretrained Transformers for Extreme Multi-label Text Classification," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, in KDD '20. New York, NY, USA: Association for Computing Machinery, Aug. 2020, pp. 3163–3171. doi: 10.1145/3394486.3403368.

[4] X. Song, A. Salcianu, Y. Song, D. Dopson, and D. Zhou, "Fast WordPiece Tokenization." arXiv, Oct. 05, 2021. doi: 10.48550/arXiv.2012.15524.

[5] M. Lewis *et al.*, “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension.” arXiv, Oct. 29, 2019. doi: 10.48550/arXiv.1910.13461.