AbPROP: Language and Graph Deep Learning for Antibody Property Prediction

Talal Widatalla [1] **Zachary Rollins** [1] **Ming-Tang Chen** [2] **Andrew Waight** [3] **Alan C. Cheng** [1]

Abstract

Despite the rising importance of monoclonal antibodies as clinical therapeutics, sizes of experimental datasets for antibody developability remain in the 100's, creating a challenge for predicting developability properties during the discovery phase, and hindering success in bioprocess development. Here we present AbPROP, which combines various three-dimensional graph and language models to leverage structurally aware learning and pre-training on millions of unlabeled sequences towards predicting antibody binding and developability properties. Our key findings: (1) supervised fine-tuning on experimental data greatly improves performance of language models compared to using supervised few-shot prediction with embeddings, (2) the addition of a structurally aware prediction head for language model fine-tuning increases downstream performance up to a distinct threshold beyond which integrating structural features into language models does not yield further improvement, and (3) unlabeled sequence pre-training increases performance on downstream tasks while structural pre-training has minimal to no effect.

1. Introduction

Monoclonal antibodies (mAbs) are one of the fastest growing segments of therapeutics, due to several factors including favorable specificity, potency, pharmacokinetic half-lives and wide therapeutic versatility (Goulet & Atkins, 2020; Bailly et al., 2020). However, most candidates fail in the drug development process, a primary source of the large cost to antibody drug discovery. Poor antibody developability profiles can further hamper the pipeline by requiring investment in additional downstream processes for formulation and storage, preventing preferred methods of administration (needing to resort to intravenous injection), and increasing overall cost of goods – which, together, limits accessibility of new therapeutics (Bailly et al., 2020; Whaley & Zeitlin, 2022). Developability properties include but are not limited to thermostability, aggregation propensity, poly-specificity (affinity to off-target antigens) and hydrophobicity (Bailly et al., 2020; Waight et al., 2023). While throughput capabilities are increasing, dataset sizes for these properties (with high fidelity scalar measurements) remain in the 100's, and combining similar assays is rarely possible due to differences in experimental parameters. Thus, we considered use of a Large Language Model (LLM) to leverage pre-training on millions of unlabeled, naturally occurring antibody sequences to predict these properties despite small dataset sizes. In addition to developabilty properties, we also train and predict on two large ($>10^5$ sequences) datasets from yeast-display followed by FACS-based (Fluorescence-Activated Cell Sorting) binding assays.

1.1. Large Language Models for Protein Design

Transformer based LLMs trained on millions of unlabeled protein sequences have been successful at a wide variety of tasks (Devlin et al., 2019) including classifying sequences as human or non-human, (Prihoda et al., 2022) generating antibody CDR loop residues based on a given sequence scaffold (Shuai et al., 2021) and serving as foundation for state-of-the-art protein structure prediction algorithms (Jumper et al., 2021). The success of the transformer architecture in these tasks can be partially attributed to its attention mechanism – calculation of a scaled dot product between key and query vectors, both representing the amino acids in a sequence, through which the attention between residues is derived (Vaswani et al., 2017). This allows for learning of long-range and higher order relationships to make predictions based on the structure, and therefore function, of proteins. The function of proteins are heavily reliant on interactions between multiple residues far apart in sequence due to the often globular 3D fold of protein structures via hydrophobic, van der Waals, and non-bonded electrostatic interactions

[1]Modeling and Informatics, Discovery Chemistry, Merck & Co., Inc., South San Francisco, CA, USA [2]Discovery Biologics, Merck & Co., Inc., Boston, MA, USA [3]Discovery Biologics, Merck & Co., Inc., South San Francisco, CA, USA. Correspondence to: Talal Widatalla <talalw@stanford.edu>, Alan C. Cheng <alan.cheng@merck.com>.

The 2023 ICML Workshop on Computational Biology. Honolulu, Hawaii, USA, 2023.

(Gromiha & Selvaraj, 2004; Vaswani et al., 2017). For proteins, LLM training is typically masked language modeling (MLM), in which the identities of residues hidden from the model are predicted (Nijkamp et al., 2022). In this work we employ AbLang, a RoBERTa based Protein Language Model (PLM) trained on the OAS – a database of hundreds of millions of antibody V_H and V_L sequences, compiled by Olsen et. al (2021) (Liu et al., 2019). AbLang has 85 million parameters between its V_H and V_L pretrained models and was chosen due to it being open source and having better MLM performance on antibodies than a PLM baseline trained on general protein sequences (Olsen et al., 2022).

1.2. Graph Neural Networks for Protein Design

Protein structures can be represented as graphs in which residues are nodes and information about their connections are encoded in edges. In recent years protein GNNs have been successful at the task of inverse-folding – predicting the proteins sequence given its backbone structure (Yang et al., 2022b; Ingraham et al., 2019; Jing et al., 2020). All three GNNs in this study employ algorithms to extract structurally aware embeddings from graphs. The Graph Attention Network (GAT) is a general purpose algorithm for predicting properties from graphs, with applications from modeling networks of journal citations to protein-protein interactions (Velickovic et al., 2017). It takes graphs with featured nodes and featureless edges as input, a stark contrast to Geometric Vector Perceptrons (GVP) (Jing et al., 2020) and the Structured Transformer (SGN) (Ingraham et al., 2019). The two latter models employ geometric distances and torsion angles calculated from the protein structure and employ them as node and edge encodings. Additionally, GVP and SGN were pretrained with PDB structures on the inverse-folding task. AbPROP models using these heads were initialized with these weights, made available by Yang et. al. (2022b).

1.3. AbPROP Model Architecture

AbPROP combines sequential learning of AbLang with structurally aware learning of GNNs. In theory, passing sequence representations through a GNN head provides an advantage in that dependencies between residues do not have to be learned through pretraining, as would be the case with a sequence only model, but are explicitly defined by graph edges (Wang et al., 2022; Gligorijević et al., 2021). Shown in **Figure 1** we pass information from AbLang to GAT for AbPROP-GAT. We additionally evaluated a sequence-only option we refer to as AbPROP-Seq in which sequence embeddings are passed directly to the pooling function for property prediction. SGN and GVP, unlike GAT, are pretrained on the inverse-folding task (Olsen et al., 2022) and employ angular and geometric features in addition to the basic graph depicted in **Figure 1**. With AbPROP-SGN and AbPROP-GVP, we add amino-acid logits from AbLang to structural node and edge features from the graph featurization step (Appendix B.8,9) instead of the sequence embedding. Graph convolutions then use sequence and structural data to produce graph embeddings, which are pooled into a representation (Appendix B.10) for property prediction.

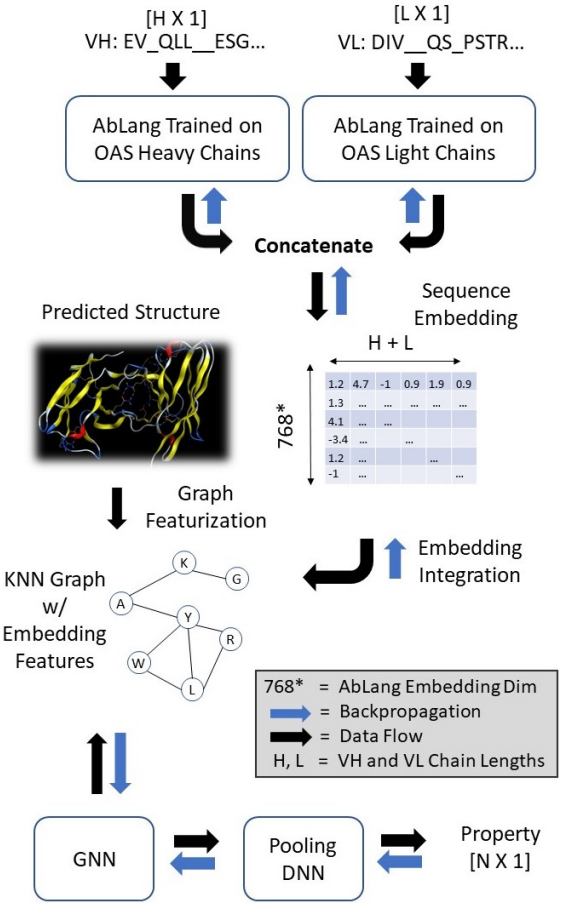

Figure 1. AbPROP. The F_V(s) are passed into AbLang models trained on the corresponding set of OAS chains. Using the predicted structure of the F_V(s), a k-nearest-neighbors graph is generated and featurized with representations as node features. From this, GAT/GVP/SGN graph convolutions produce graph embeddings for each node that are finally passed to a pooling neural net which predicts the property. The number of layers loss is backpropagated into AbLang is a hyperparameter.

2. Evaluation and Results

In **Table 1** we show performance of all four AbPROP models. Baselines are random forests with one-hot sequence encoding and AbLang embeddings as features (Pedregosa et al., 2011). For AbPROP models we show performance after hyperparameter tuning with five-fold cross validation then testing on unseen, and sufficiently unique holdout data. The MRGX and MERS data are large 4-class and 5-class binding affinity classification datasets against the

MrgX1 GPCR and MERS-CoV Spike Protein respectively, derived from proprietary single-domain V_{HH} yeast display and FACS experiments against these targets. The HIC-RRT (relative hydrophobicity) and T-Agg (Temperature of Aggregation) scalar datasets are also proprietary. The PSR (Poly-Specific Reactivity), HIC-RT (hydrophobicity) and T-Mid (Temperature of protein denaturation) scalar datasets were retrieved from Shehata et. al (2019). Lastly, the Amyloid Light Chain Database (AL-Base) is a binary classification task for antibody light chains as amyloidic (aggregation prone) or non-amyloidic from Bodi et. al (2021). For this dataset, instead of one-hot encodings and embeddings as baseline comparators, we show performance of a novel decision-tree based machine learning model VLAmY-Pred (Embedding Column) as well the best performing physiochemical aggregation predictor (One-Hot Column), both from Rawat et. al. (2021). See Appendices A and B for details on datasets and methods.

Table 1. Performance of AbPROP Models and Baselines. Baselines have a 95% confidence interval from n = 10 trials with varying random states and number of trees. For scalar valued datasets we provide the Spearman rank correlation coefficients [0-1] and percent accuracies for classification datasets [0-100].

	BASELINES		ABPROP MODELS			
DATA SET	ONE-HOT	EMBEDDING	SEQ	GAT	GVP	SGN
MERS	66.9± 0.2	65.0± 0.1	68.5	**68.7**	68.1	68.1
MRGX	57.7± 0.4	57.3± 0.2	58.4	**59.4**	59.0	58.7
PSR	-0.03± 0.01	0.08± 0.01	**0.27**	0.25	0.23	0.24
HIC-RT	0.36± 0.01	0.32± 0.02	0.46	**0.51**	0.43	0.48
T-MID	0.49± 0.01	0.38± 0.02	0.61	**0.62**	0.54	0.60
T-AGG	0.41± 0.02	0.47± 0.03	0.44	**0.55**	0.44	0.43
HIC-RRT	0.67± 0.01	0.29± 0.02	0.72	0.73	**0.77**	0.55
AL BASE	48.4	71.0	85.3	**86.6**	N/A	N/A

2.1. Dataset Titration

The assays in this work are time and resource intensive (Shehata et al., 2019; Bailly et al., 2020), hence finding dataset size requirements for ML models is imperative. We began to understand comparative performance between one-hot and AbPROP-Seq models by doing a titration of our HIC-RRT dataset along with 'default' hyperparameters on downsampled data, shown in dark blue in **Figure 2**. We expect hyperparameter-optimized models to perform better than shown (blue line), and indicate an 'over-fitted' scenario with a dashed light-blue line, which indicates hyperparameter optimization using the hold-out set. These two lines indicate the likely bounds of a properly hyperparameter-optimized model, which, at the low-n end, is challenging due to dataset sizes. Our full dataset accuracy reported in **Table 1** is shown

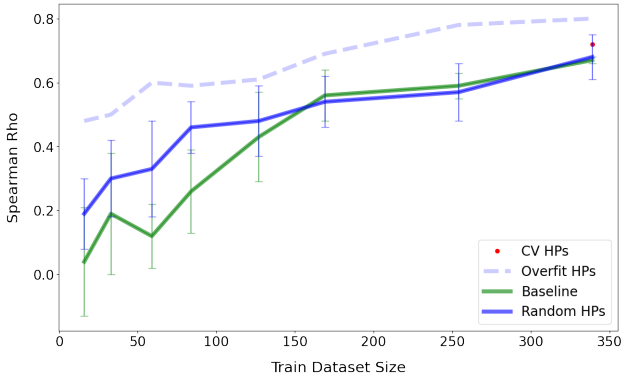

Figure 2. HIC-RRT Titration. Baseline is random forest with one-hot-encoding. Dark blue is AbPROP-Seq with default hyperparamters. Light blue is the best score from 100 models with hyperparameters fitted on the held-out set, providing an upper bound. The red point indicates the proper performance for the full dataset. Standard error derived from 95% confidence intervals from 10 trials with varying data splits and hyperparameters.

in **Figure 2** in red, and lies between these bounds. Thus in the worst case, AbPROP will outperform one-hot encoding for small data and perform similarly for larger data (>150); in the best case, where there is no validation overfitting, it will outperform at all sizes.

2.2. Effect of Sequence and Structure Pretraining

The hypothesis behind AbPROP is pretraining on large datasets of unlabeled proteins can be leveraged for increased performance on small, labeled datasets. To test this, we conducted an experiment (**Figure 3**) comparing model performance of AbPROP-Seq with MLM pretraining versus randomly initialized weights. We analogously performed the same experiment with the GNN portion of AbPROP-SGN and AbPROP-GVP, holding the AbLang transformer pretrained but either using randomly initialized or weights from inverse-folding pretraining for the GNN heads.

3. Discussion

3.1. Finetuning

It is clear from comparison of the embedding baseline with AbPROP-Seq in **Table 1** that PLM performance is increased with supervised fine-tuning as demonstrated by significant gains in six of seven datasets. We hypothesize this is true because language models learn the likelihood of a sequence based on its training data of observed sequences (Nijkamp et al., 2022). Thus, because the training data of sequences are functionally selected, the sequence probability distribution learned by AbLang should resemble the overall fitness distribution of natural sequences. However, here we predict

developability properties, and being a favorable sequence for pharmaceutical development does not necessarily correlate to being a favorable sequence in natural selection. Thus, it follows that fine-tuning of the learned representation which predicts sequence likelihood is needed to increase predictivity for fitness-adjacent developability endpoints.

3.2. Sequence and Structure Pretraining

Previous literature has shown an advantage in sequence pretraining for property prediction on proteins (Yang et al., 2022b), but none have directly compared this advantage with structural pretraining. As seen in **Figure 3**, significant performance gain using pretrained weights compared to random weights was observed for the language model but not the GNN head. Despite the pretraining MLM task being starkly different than property prediction, this result indicates there is predictive value in sequence pretraining as a starting point for property prediction. As for the GNN results, there are several possible explanations for the lack of improvement. Most obvious is inaccurate input structures. The pretrained GNNs (GVP and SGN) employ alpha-carbon distance vectors and dihedral angles as structural features from which the pretrained graph convolutions extract meaning from; however, our input structures were predicted using IgFold which cannot accurately model the most variable region of the antibody, CDR H3, (Ruffolo & Gray, 2022) which some argue must be modeled by an ensemble of structures due to the intrinsic flexibility of the loop in solution (Fernández-Quintero et al., 2019). Therefore, these distances, orientations and angles computed from predicted, static structures may just be noisy features from which the pretrained GNN models can use to overfit to the target variables, which is less likely for GAT, our best performing model, as it does not employ these features. Additionally, the models may need to be trained on more structures than the roughly 20k in CATH (of which only 3% are antibodies). Finally, an approach more sophisticated than simply concatenating language model logits to node structural encodings is needed to better integrate the language and structural features for property prediction.

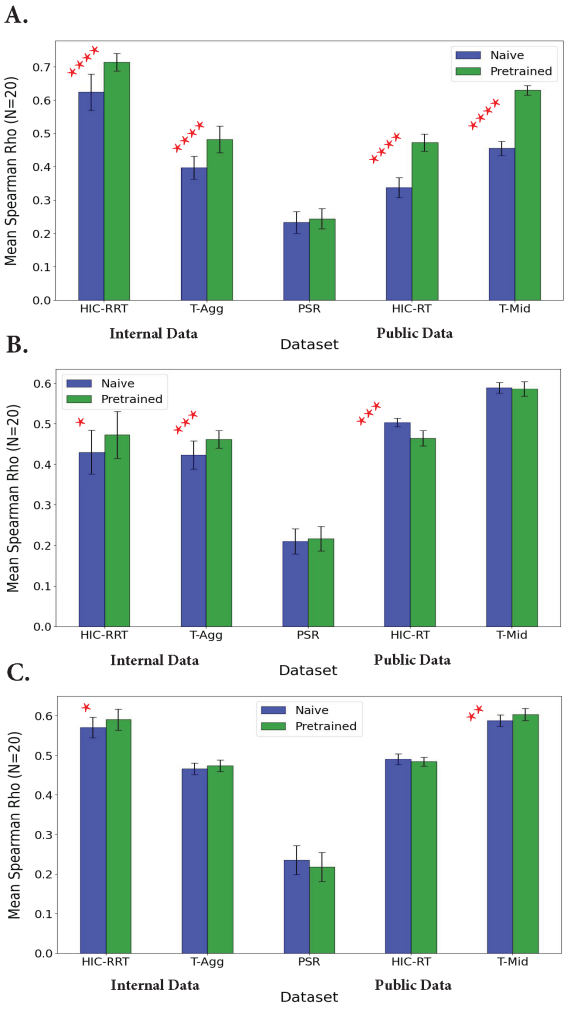

Figure 3. Effect of MLM Pretraining for A. AbPROP-Seq and Structural Pretraining for B. AbPROP-GVP and C. AbPROP-SGN on performace. Error bars show 95% confidence interval for mean Spearman coefficient from 20 trials, with hyper parameters varying between trials. *, **, ***, **** correspond to 80%, 90%, 95% and 99% certainty from a difference of means t-test.

3.3. Structural Feature Integration

In Wang et al. (2022), addition of the GNN head improved accuracy for residue level properties, but a conclusion for whole-protein property prediction was not clear and their study was limited to only two datasets. To provide a clearer answer we trained on eight protein fitness datasets with language models connected to three types of GNNs, and found a nuanced relationship between additional structural information and performance. In line with the finding that sequence pre-training is more critical than structural pre-training for property prediction, we see in **Table 1** that the structurally pre-trained (GVP and SGN) AbPROP models do not perform significantly better than sequence-only models. However, we also see that the best model across the board is AbPROP-GAT, consistently performing as well or better than the AbPROP-Seq in all eight, suggesting that structural information boosts performance when it only consists of edges connecting the nearest neighbors of each residue. Given that (1) structural pre-training and incorporation of additional structural features does not improve performance but (2) the GAT model outperforms all other models, we argue the addition of geometric and structural features from imperfect, static, predicted structures in the SGN and GVP models introduces noise which decreases performance, whereas the simple nearest neighbor edges employed by GAT is the ”sweet spot” for structural

incorporation, and is robust to the inaccuracies of predicted structures while still gleaning salient structural information from them to boost performance.

4. Conclusions

In this work we aimed to address the challenge of predicting antibody properties given antibody sequences, including for small developability datasets and binding datasets. We did this by leveraging pre-training on millions of unlabeled sequences through structurally aware fine-tuning on downstream experimental data. The AbPROP models and comparisons with associated baselines we have shown (1) sequence pre-training increases downstream performance, (2) fine-tuning sequence representations on downstream data increases performance compared to naive sequence embeddings, (3) the addition of a graph neural network head onto language models increases performance, however, in our hands, (4) structural pre-training in said graph neural net does not significantly improve performance compared to sequence pre-training nor does increasing the amount of structural features in graph heads beyond a k-nearest-neighbors graph. We suspect our latter finding (4) could be explored further. We are also exploring more sophisticated unsupervised structural pre-training techniques, training on larger sets of experimental and high fidelity predicted structures, and incorporating ensembles of structures instead of a single static structure for prediction.

Acknowledgements

The authors thank the Merck & Co., Inc. Modeling & Informatics and Discovery Biologics departments,including Marc Bailly, Kevin Metcalf, Xiao Xiao, Kevin Teng, Yara Seif, BoRam Lee, Katherine Delevaux, Essam Metwally and Jingzhou Wang, for helpful discussion and insights.

References

Bailly, M., Mieczkowski, C., Juan, V., Metwally, E., Tomazela, D., Baker, J., Uchida, M., Kofman, E., Raoufi, F., Motlagh, S., Yu, Y., Park, J., Raghava, S., Welsh, J., Rauscher, M., Raghunathan, G., Hsieh, M., Chen, Y.-L., Nguyen, H. T., Nguyen, N., Cipriano, D., and Fayadat-Dilman, L. Predicting antibody developability profiles through early stage discovery screening. *mAbs*, 12(1):1743053, 2020. ISSN 1942-0862. doi: 10.1080/19420862.2020.1743053.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

Estep, P., Caffry, I., Yu, Y., Sun, T., Cao, Y., Lynaugh, H., Jain, T., Vásquez, M., Tessier, P. M., and Xu, Y. An alternative assay to hydrophobic interaction chromatography for high-throughput characterization of monoclonal antibodies. *mAbs*, 7(3):553–561, 2015. ISSN 1942-0862. doi: 10.1080/19420862.2015.1016694.

Fernández-Quintero, M. L., Kraml, J., Georges, G., and Liedl, K. R. Cdr-h3 loop ensemble in solution – conformational selection upon antibody binding. *mAbs*, 11 (6):1077–1088, 2019. ISSN 1942-0862. doi: 10.1080/19420862.2019.1618676.

Gligorijević, V., Renfrew, P. D., Kosciolek, T., Leman, J. K., Berenberg, D., Vatanen, T., Chandler, C., Taylor, B. C., Fisk, I. M., Vlamakis, H., Xavier, R. J., Knight, R., Cho, K., and Bonneau, R. Structure-based protein function prediction using graph convolutional networks. *Nat Commun*, 12(1):3168, 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-23303-9.

Goulet, D. R. and Atkins, W. M. Considerations for the design of antibody-based therapeutics. *Journal of Pharmaceutical Sciences*, 109(1):74–103, 2020. ISSN 0022-3549. doi: 10.1016/j.xphs.2019.05.031.

Gromiha, M. M. and Selvaraj, S. Inter-residue interactions in protein folding and stability. *Progress in Biophysics and Molecular Biology*, 86(2):235–277, 2004. ISSN 0079-6107. doi: https://doi.org/10.1016/j.pbiomolbio.2003.09.003.

He, F., Woods, C. E., Becker, G. W., Narhi, L. O., and Razinkov, V. I. High-throughput assessment of thermal and colloidal stability parameters for monoclonal antibody formulations. *J Pharm Sci*, 100(12):5126–41, 2011. ISSN 0022-3549. doi: 10.1002/jps.22712.

Ingraham, J., Garg, V. K., Barzilay, R., and Jaakkola, T. Generative models for graph-based protein design. In *DGS@ICLR*, 2019.

Jain, T., Sun, T., Durand, S., Hall, A., Houston, N. R., Nett, J. H., Sharkey, B., Bobrowicz, B., Caffry, I., Yu, Y., Cao, Y., Lynaugh, H., Brown, M., Baruah, H., Gray, L. T., Krauland, E. M., Xu, Y., Vásquez, M., and Wittrup, K. D. Biophysical properties of the clinical-stage antibody landscape. *Proceedings of the National Academy of Sciences*, 114(5):944–949, 2017. ISSN 0027-8424. doi: 10.1073/pnas.1616408114.

Jing, B., Eismann, S., Suriana, P., Townshend, R. J. L., and Dror, R. O. Learning from protein structure with geometric vector perceptrons. *ArXiv*, abs/2009.01411, 2020.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S.

A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P., and Hassabis, D. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021. ISSN 0028-0836. doi: 10.1038/s41586-021-03819-2.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.

Madeira, F., Pearce, M., Tivey, A. R. N., Basutkar, P., Lee, J., Edbali, O., Madhusoodanan, N., Kolesnikov, A., and Lopez, R. Search and sequence analysis tools services from embl-ebi in 2022. *Nucleic acids research*, pp. gkac240, 2022. ISSN 0305-1048. doi: 10.1093/nar/gkac240.

Navarin, N., Tran, D. V., and Sperduti, A. Universal readout for graph convolutional neural networks. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, 2019. ISBN 2161-4407. doi: 10.1109/IJCNN.2019.8852103.

Nijkamp, E., Ruffolo, J. A., Weinstein, E. N., Naik, N. V., and Madani, A. Progen2: Exploring the boundaries of protein language models. *ArXiv*, abs/2206.13517, 2022.

Olsen, T. H., Boyles, F., and Deane, C. M. Observed antibody space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences. *Protein Science*, 31:141 – 146, 2021.

Olsen, T. H., Moal, I. H., and Deane, C. M. Ablang: An antibody language model for completing antibody sequences. *bioRxiv*, pp. 2022.01.20.477061, 2022. doi: 10.1101/2022.01.20.477061.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12(null):2825–2830, 2011. ISSN 1532-4435.

Prihoda, D., Maamary, J., Waight, A., Juan, V., Fayadat-Dilman, L., Svozil, D., and Bitton, D. A. Biophi: A platform for antibody design, humanization, and humanness evaluation based on natural antibody repertoires and deep learning. *MAbs*, 14(1):2020203, 2022. ISSN 1942-0862 (Print) 1942-0862. doi: 10.1080/19420862.2021.2020203.

Rawat, P., Prabakaran, R., Kumar, S., and Gromiha, M. M. Exploring the sequence features determining amyloidosis in human antibody light chains. *Scientific Reports*, 11(1), 2021. ISSN 2045-2322. doi: 10.1038/s41598-021-93019-9.

Ruffolo, J. A. and Gray, J. J. Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies. *Biophysical Journal*, 121(3):155a–156a, 2022. ISSN 0006-3495. doi: 10.1016/j.bpj.2021.11.1942.

Sevy, A. M., Chen, M.-T., Castor, M., Sylvia, T., Krishnamurthy, H., Ishchenko, A., and Hsieh, C.-M. Structure- and sequence-based design of synthetic single-domain antibody libraries. *Protein Engineering, Design and Selection*, 33:gzaa028, 2020. ISSN 1741-0126. doi: 10.1093/protein/gzaa028.

Shehata, L., Maurer, D. P., Wec, A. Z., Lilov, A., Champney, E., Sun, T., Archambault, K., Burnina, I., Lynaugh, H., Zhi, X., Xu, Y., and Walker, L. M. Affinity maturation enhances antibody specificity but compromises conformational stability. *Cell Reports*, 28(13):3300–3308.e4, 2019. ISSN 2211-1247. doi: 10.1016/j.celrep.2019.08.056.

Shuai, R. W., Ruffolo, J. A., and Gray, J. J. Generative language modeling for antibody design. *bioRxiv*, pp. 2021.12.13.472419, 2021. doi: 10.1101/2021.12.13.472419.

Vaswani, A., Shazeer, N. M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *ArXiv*, abs/1706.03762, 2017.

Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio', P., and Bengio, Y. Graph attention networks. *ArXiv*, abs/1710.10903, 2017.

Waight, A., Prihoda, D., Shrestha, R., Metcalf, K., Bailly, M., Ancona, M., Widatalla, T., Rollins, Z., Cheng, A. C., Bitton, D., and Fayadat-Dilman, L. Machine learning feature importance of in silico descriptors and prediction of IgG1 monoclonal antibody developability properties. [Submitted for Publication], 2023.

Wang, Z., Combs, S. A., Brand, R., Calvo, M. R., Xu, P., Price, G., Golovach, N., Salawu, E. O., Wise, C. J., Ponnapalli, S. P., and Clark, P. M. Lm-gvp: an extensible sequence and structure informed deep learning framework for protein property prediction. *Scientific Reports*, 12(1), 2022. ISSN 2045-2322. doi: 10.1038/s41598-022-10775-y.

Whaley, K. J. and Zeitlin, L. Emerging antibody-based products for infectious diseases: Planning for metric ton manufacturing. *Human Vaccines 'I&' Immunotherapeutics*, 18(2):1–11, 2022. ISSN 2164-5515. doi: 10.1080/21645515.2021.1930847.

Xu, Y., Roach, W., Sun, T., Jain, T., Prinz, B., Yu, T. Y., Torrey, J., Thomas, J., Bobrowicz, P., Vásquez, M., Wittrup, K. D., and Krauland, E. Addressing polyspecificity of antibodies selected from an in vitro yeast presentation system: a facs-based, high-throughput selection and analytical tool. *Protein Eng Des Sel*, 26(10):663–70, 2013. ISSN 1741-0126. doi: 10.1093/protein/gzt047.

Yang, K. K., Lu, A. X., and Fusi, N. Convolutions are competitive with transformers for protein sequence pretraining. *bioRxiv*, 2022a.

Yang, K. K., Zanichelli, N., and Yeh, H. Masked inverse folding with sequence transfer for protein representation learning. *bioRxiv*, pp. 2022.05.25.493516, 2022b. doi: 10.1101/2022.05.25.493516.

Name	Property	Chain	Source	Output	Size	# Holdout	Split
T-Mid	Thermostability	V_L+ V_H	Public	Scalar	483	73	Random
PSR	Polyspecificity	V_L+ V_H	Public	Scalar	535	65	Random
HIC-RT	Hydrophobicity	V_L+ V_H	Public	Scalar	485	86	Random
AL-Base	Aggregation	V_L	Public	Binary	1828	305	Random
T-Agg	Aggregation	V_L+ V_H	Internal	Scalar	520	81	Clustered
HIC-RRT	Hydrophobicity	V_L+ V_H	Internal	Scalar	514	103	Clustered
MERS	Affinity	V_HH	Internal	4-Class	48k	2396	Clustered
MRGX	Affinity	V_HH	Internal	5-Class	14k	1114	Clustered

Figure 4. Datasets. In order to test the robustness of our models to predict for several different use cases we procured a group of datasets diverse in sequence type, size, assay, and output shape to train on from public and internal sources.

A. Data

A.1. Internal Datasets

A.1.1. MERS and MRGX1 Binding Affinity Yeast Display

In this paper we demonstrate our performance on binders identified from two in-vitro phage-yeast display campaigns for single-domain V_{HH}'s (nanobody) on two separate antigen targets. Naturally produced by camelids, nanobodies retain the high affinity and specificity of traditional immunoglobulins while only having a heavy chain (Sevy et al., 2020). We have developed a diverse nanobody library with characteristics that closely mimic the natural repertoire and isolated V_{HH} sequences against MERS spike protein and MRGX1 GPCR, in which we were able to isolate diverse binders. Mutations were limited to the CDR H3 and positions in the H2 and H1 which were predicted to affect binding, in order to minimize mutations to the humanized germline framework. CDR H1 and H2 lengths remained fixed while CDR H3 length varied. Residues which impact binding were determined via Rosetta DDG analysis of binding interface residues in 208 nanobody-antigen complexes in the PDB. After the library was designed and sequences were expressed in yeast cells, magnetic cell sorting (MACS) was employed to filter out cells with sequences that bound off-target reagents. Then yeast cells containing antigen-specific sequences were sorted using fluorescence-activated cell sorting (FACS) into bins of antigen concentration. Finally, with next generation sequencing of the nanobodies contained in the sorted cells, binding sequences were correspondingly sorted into four or more antigen concentration bins–we used (100, 30, 10, 3) and (300, 30, 10, 3) nM concentrations for the MERS and MRGX1 antigens, respectively. Details of library generation and binding assays can be found in Sevy et. al (2020). In addition to being against different targets and having a different number of affinity bins, the size of the MERS and MRGX datasets are 48,134 and 14,064 VHH sequences respectively.

A.1.2. Hydrophobic Interaction Chromatography Relative Retention Time (HIC-RRT)

The HIC-RRT dataset used in the present study contains 514 IgG1 datapoints from 24 different clustered mAb projects, with each project consisting of sequences designed for a different antigen target. As described in Waight et al. (2023), the hydrophobicity of a given mAb were determined by recording the relative elution time (compared to a control mAb) through

a hydrophobic interaction chromatography column and measuring the absorbance spectra at 280nm, A280nm. The resulting value is relative retention time (RRT), which was done to increase the accuracy of HIC measurements due to gradual drift to longer retention times during the lifetime of the column. Taking the relative retention time eliminates the effect of this confounding variable. Briefly, in the assay, 50 µg of sample at 0.5-1.0 mg/mL were mixed 1/1 (v/v) with a buffer solution (100 mM sodium phosphate, 2 M Ammonium Sulfate pH 7.0), filtered through a 0.22 µm PVDF membrane, and 60 µL loaded on a Dionex Pro Pac HIC-10 column. The column was equilibrated in 100 mM sodium phosphate, 1 M Ammonium Sulfate pH 7.0 (mobile phase A) and samples were eluted using an inverted gradient to 100 mM sodium phosphate pH 7.0 (mobile phase B).

A.1.3. Aggregation Temperature by nano-DSF (T-Agg)

The T-Agg dataset used in the present study contains 525 IgG1 datapoints from 25 different internal clustered mAb projects. Nano-DSF (Nano–Differential Scanning Fluorimetry) studies were performed as described in Waight et al. (2023) using the Nanotemper Prometheus NT.48 instrument to measure protein stability. Briefly, samples (10 µL at 0.5-1 mg/mL) were loaded into capillaries and the temperature ramped at 1°C/min from 20°C to 94.8°C. The melting point ($T_{m,onset}$ and T_m) (°C) temperatures indicate the structural stability of the samples, and the unfolding curves (or thermogram) were generated by plotting the ratio of the fluorescence intensities (F_{350nm}/ (F_{330nm}) as a function of temperature, with each intensity tracking the level of folded or unfolded protein. The melting point temperatures were defined by the onset ($T_{m,onset}$) and inflection point (T-Mid) of the thermogram. The colloidal stability of the sample can be simultaneously determined by measuring the attenuation of back reflected light intensity passing through the sample and the aggregation temperature (T-Agg) was defined as the point at which light scattering increases (or back reflected light intensity decreases) due to unfolding.

A.2. External Datasets

The three datasets for PSR, HIC-RT and T-Mid were all derived from two studies using the same assays released by Adimab (Shehata et al., 2019; Jain et al., 2017). The difference between the two studies is the size and origin of mAbs tested, with the Jain et. al. study testing 137 clinical mAbs and the Shehata et. al. study testing 400-450 (depending on assay) mAbs from primary naïve B cells, IgM and IgG memory B cells, and long-lived plasma cells (LLPCs) (2019).

A.2.1. Poly-specificity reagent assay (PSR)

The PSR assay is a measure of the relative non-specific binding of a given mAb compared to reference antibodies. This data was collected as previously described (Shehata et al., 2019; Jain et al., 2017; Xu et al., 2013). In short, soluble membrane proteins were prepared from CHO cells, biotinylated, mixed with IgG-presenting yeast, secondary labelled (Extravidin-R-PE, anti-human LC-FITC, and propidium iodide) and analyzed by FACS. The median fluorescence intensity (MFI) in the R-PE channel is used to assess nonspecific binding and normalized to reference antibodies with low, medium, and high MFI values (in the R-PE channel).

A.2.2. Hydrophobic interaction chromatography retention time (HIC-RT)

The hydrophobicity of a given mAb in these datasets were an analog to internally collected data, however, under modified experimental conditions (Estep et al., 2015; Shehata et al., 2019; Jain et al., 2017). In brief, 5 µg of sample at 1 mg/mL were mixed into a buffer solution (1.8 M ammonium sulfate and 0.1 M sodium phosphate pH 6.5) to reach 1 M ammonium sulfate and loaded into a Sepax Proteomix HIC butyl-NP5 column. The column is equilibrated in 1.8 M ammonium sulfate, 0.1 M sodium phosphate pH 6.5 (mobile phase A) and samples are eluted (1 mL/min over 20 min) using an inverted gradient to 0.1 M sodium phosphate pH 6.5 (mobile phase B).

A.2.3. Melting Temperature by DSF (T-Mid)

The mAb structural stability, T-Mid (also referred to as T_m), collected in these datasets (Shehata et al., 2019; Jain et al., 2017; He et al., 2011) were an analog to internally collected data, however, under modified experimental conditions. Briefly, melting temperature (T-Mid) was determined by loading 10 µl of sample at 1 mg/mL mixed with 10 µL of 20x SPYRO orange onto a plate that is scanned from 40 °C to 95 °C at a ramp rate of 0.5 °C/2 min. The T-Mid is defined as the inflection point of the thermogram and determined using BioRad analysis software.

A.2.4. Light Chain Amyloid Dataset and Baselines (AL-Base)

This dataset consists of kappa and lambda antibody light chains, which are either amyloid sequences derived from patients with AL amyloidosis or chains derived from patients with multiple myeloma or healthy controls, i.e., non-amyloid chains. A model which can predict a which class a given light chain falls under could also be extended to engineering of aggregation resistant antibody therapeutics (Rawat et al., 2021). Thus, to investigate whether our model could accomplish this better than other approaches, we trained on the same smaller subset (1828 out of 4364) of sequences from Rawat et. al., who reported the accuracy of several physio-chemical and structure-based baseline models as well as their own novel machine learning approach "VLAmY-Pred". This model was a decision-tree algorithm which used physicochemical amino-acid features from AA-Index averaged across the whole chain. Some important differences in the k-fold cross validation and test sets between AbPROP and VLAmY-Pred is (1) AbPROP was tested on 5 folds of 83% of the data instead of "VLAmY-Pred" which was tested on 10 folds of presumably 90% of the data (100% if they included their test set) (2) the AbPROP test set was a randomly selected 17% of the sequences while AL-Test was a randomly selected 10% of the sequences (Al-Test was not released so we were unable to test on this exact set), and (3) the TANGO, WALTZ and RFAmyloid models were completely blind to the AL-Base data while AbPROP and VLAmY-Pred trained on the data.

B. Methods

B.1. Structure and Graph Generation

For training of AbPROP models, predicted structures were generated for all sequences using IgFold with PyRosetta minimization due to its speed and state-of-the-art accuracy for antibody structure prediction (Ruffolo & Gray, 2022). For all sequences only the F_V's were modeled, whether that be the V_L, V_H or both. Following structure generation, structures are transformed into graphs, in which nodes are amino acids, and edges are placed between the closest k amino acids to each respective node as in Ingraham et. al (2019). For the GAT model, k is a hyperparameter, but for GVP and SGNN, k was fixed at 20 so the pretrained weights could be used (pre-training was done by Yang et. al. (2022b) with k-nearest-neighbors graphs generated with k = 20). Depending on the GNN, the edges either contain or do not encode the geometry between its respective amino acids. Code for graph transformation and processing was adapted from the models and respective GitHub repositories in Ingraham et. al. (SGN) (2019), Yang et. al. (MIF-ST) (2022b) and Wang et. al (LM-GVP) (2022).

B.2. MSA Generation

For all sequence datasets, with the exceptions of MERS and MRGX1 due to their larger size, multiple sequence alignments (MSA) were created using the MOE software. Split into FASTA files with all VH and VL chains, the CDRs were first annotated using IMGT numbering (Lefranc, Pommié et al. 2003), and aligned using MOE's default settings. Due to the smaller size of these datasets the gaps produced were minimal relative to the sequence size and took a trivial amount of compute time. Due to the larger size of the MRGX1 dataset, the Clustal Omega software, which utilizes seeded guide trees and Hidden Markov Model profile techniques (Madeira et al., 2022), was used for alignment as it can be paralleled across several CPUs. Due to the even larger size and diversity of the MERS datasets, additional manual data trimming was employed to minimize the number of gaps, given the compute and memory required by transformers scale quadratically with input sequence length (Yang et al., 2022a). Through reducing our dataset size from 50406 to 48131, we reduced our MSA length from 340 to 131; this 2.6 fold reduction in length results in an estimated 6.7 fold reduction in compute needs from AbLang.

B.3. Data Splitting

All datasets were split into train and holdout. Five-fold cross validation was employed in hyperparameter tuning, in which the validation set was a randomly selected but predefined fifth of the training data. Models with the highest average accuracy across all folds were selected for holdout prediction, which were never seen by any of the models during validation, ensuring we are measuring the model's ability to generalize to unseen data. Specifically, the five versions of the model trained on each validation split were ensembled for prediction on the holdout set. Depending on the dataset and its diversity, we either employed random or clustered splitting for holdout selection, as described below.

B.4. Random Splitting

For the AL Base dataset, we concluded a simple random split was appropriate given we were comparing performance with Rawat et. al (2021) which had results derived from presumably random k-fold-cross-validation and holdout testing. For the datasets from Jain et. al. (2017) and Shehata et. al. (2019) (PSR, T-Mid and HIC-RT), approximately 400 human mAbs were measured in Shehata et. al. by deriving sequences from primary naïve B cells, IgM and IgG memory B cells, and long-lived plasma cells (LLPCs) from the germline cells of four healthy donors (i.e., as opposed to single/double mutation yeast display libraries). Additionally, the clinical mAbs included in the Jain et. al (2017) are inherently diverse due to their targeting of different antigens. From these two facts, our intuition was the combined Jain and Shehata dataset would be diverse enough for random splitting. This was confirmed by looking at the distribution of sequence similarity across all pairs, where we found 99.9% of sequence pairs had under 92.9% similarity, 98.8% of sequence pairs had under 87.1% similarity.

B.5. Sequence Similarity Clustering

Unlike the datasets from Jain et. al. (2017) and Shehata et. al. (2019), our internal T-Agg and HIC-RRT datasets had sequences with high similarity due to the natural grouping of antibodies from therapeutic pipeline projects. To effectively assess the generalizability of the models we built, the holdout set was designed to contain low sequence identity (<95%) from the train/validation stage to provide a worst-case scenario model fit. In brief, the sequence analysis workflow as described in Waight et al.(2023), begins with generation of a pairwise mutation matrix, and clustering of sequences based on the matrix. The sequences are split using a stratified group method that ensures (1) <95% sequence identity for all sequences in the holdout compared to train/validation, (2) the holdout contains approximately 20% of the dataset, and (3) there is a representative distribution of the biophysical property in both the train/validation and holdout. A slightly different approach was taken for the larger MERS and MRGX-1 yeast display datasets. We clustered the dataset into 600 clusters with k-means clustering and sequence similarity as the distance metric. To ensure we had large clusters in our holdout set, we sampled 3 random clusters from those which were 50-100 sequences, then proceeded to randomly add 197 smaller sized clusters to the holdout set. As such, the holdout set has a total of 200 clusters that are unseen during training. For the remainder of the dataset, we stratified split the sequences between train and holdout, so that the holdout set summed up to 2,500 sequences each, which was eventually reduced to 2,396 sequences after data trimming and MSA generation.

B.6. AbLang

AbLang is a PLM trained on all sequences in the OAS (Observable Antibody Space) for the purpose of restoring missing residues in antibody sequences (Olsen et al., 2021). It inherits from the Bidirectional Encoder Representations from Transformers (BERT) base architecture of 12 transformer layers with 12 attention heads and was trained on the MLM task of predicting masked residues. Specifically, 1 to 25% of residues were chosen per sequence in the OAS, and, of this subset, 80% were masked, 10% were randomly changed and 10% were left unchanged. AbLang was shown to restore the missing residues of antibody sequences better than using IMGT germlines or by using a PLM trained on general protein sequences, suggesting for antibody related downstream tasks, using a language model trained on antibody sequences offers better learned representations for antibody related tasks. For this reason as well as it being open source, we selected AbLang to serve as the transformer encoder portion of our model.

B.7. Graph Attention Network

The graph attention network (GAT) aims to harness the success of self-attention in sequence-based tasks for graph input, the foundation of which is its graph attention layer. Each GAT layer iterates the updates of the embeddings or hidden state of each node based on "messages" relayed from nearby residues (Velickovic et al., 2017). Unlike the transformer however, GAT takes advantage of the relational knowledge contained in the edges of graphs. A significant advantage to LLMs, which must infer protein contacts, GAT and GNNs largely are implicitly given prior knowledge of the inter-dependencies of residues by way of an input structure (Wang et al., 2022). The advantage is the edges allow use of masked attention, in which attention coefficient calculations are restricted to residue pairs which are neighbors, unlike the original transformer approach which calculates attention between all N^2 residue pairs. The attention coefficient for a given edge between two nodes is calculated by multiplying both nodes' representations by a weight matrix, then multiplying an attention weight vector to the concatenation of these transformed representations. This operation is done along several attention heads which have different weight matrices. Then for a given node, the attention weight of all its edges, aggregated across all attention heads, is then used to update its hidden state. Through each iteration each node's representation becomes contextually aware

of its surrounding nodes. GAT achieved state of the art performance on protein-protein interaction and citation network datasets (Velickovic et al., 2017).

B.8. Structured Transformer

The structured transformer, or as we refer to it here, Structured Graph Network (SGN), is one of the first inverse-folding deep learning model for proteins. It functions very similar to a transformer, predicting masked residues autoregressively, with a few key differences relating to its prediction being pre-conditioned on the structure of the protein. First, like the approach of GAT, its calculation of self-attention is restricted to residue pairs which are neighbors, defined for proteins using a k-nearest-neighbors graph derived from the Euclidean distance between residues. Unlike GAT, SGN calculates attention coefficients like the original transformer, by taking the scaled-dot product of the two nodes but nonetheless updates node-embeddings using these coefficients in a similar manner as GAT (Ingraham et al., 2019). Another key difference is that the only structural information used by GAT is knowledge of which nodes are neighbors, while SGN utilizes node and edge features. It includes the distances between residues, and orientation and direction of residues as a structural encoding for each edge. It then calculates a positional encoding like the original transformer (having to do with the distance between residues in sequence), and finally concatenating the positional and structural encoding to become the starting features of each edge. For node features they take the sine and cosine of the three dihedral backbone angles as the starting features for each residue. The additional structural information aims to help the model make predictions of missing residues by informing it of the geometry of its neighbors (Ingraham et al., 2019).

B.9. Geometric Vector Perceptron

Unlike the GAT and SGN, the Geometric Vector Perceptrons (GVP) does not have an attention mechanism, and instead employs a message passing algorithm, using messages from neighboring nodes and edges to update node representations at each step. Details are provided in Jing et al. (2020). Of all three structural graph algorithms, it encodes the most structural information in its node representations, with each node having scalar and vector features. The node embedding includes a one-hot representation of the amino acid when available, the sine and cosines of the three dihedral angles (same as SGN), and six unit vectors providing orientations between inter- and intra-residue C_α and C_β atoms. The edge embeddings contain a unit vector of the direction between the two residues' C_α atoms, an encoding of the distance between the residues, and a positional encoding of the sequence position distance of the two residues (same as SGN and the original transformer).

B.10. Pooling Function

We employed two different pooling functions to predict properties from the output of GNN or Transformer layers. The output of the GNN (or of AbLang if using sequence-only model) layers will be in $\mathbb{R}^{LXH}$, with L being the sequence length of the MSA and H being the embedding size of the GNN or AbLang encoder. The first pooling function is a simple averaging across the embedding dimension, creating a vector of size L, followed by dense neural layers which predicts the output from the L-length vector. Our other pooling function was an implementation of the "Universal Readout Function" for graphs from Navarin et al.(2019), which is a learnable pooling function consisting of two sets of dense layers, Phi (ϕ) and Rho ρ (Rho). ρ, like the averaging function, first operates on the embedding dimension of length H, transforming the output from $\mathbb{R}^{LXH}$ to $\mathbb{R}^{L}$. ρ is followed by ϕ, which operates on the residue dimension, transforming the ρ output from $\mathbb{R}^{L}$ to $\mathbb{R}^{O}$, where O is the size of the target variable. Essentially, ρ learns to extract information from the relevant positions in the embedding dimension, while ϕ learns to extract information from relevant the positions in the MSA. Universal pooling was found to be superior to averaging in Navarin et al. (2019), while Wang et al. (2022a) found averaging to be the best pooling function, thus we included the choice of universal versus average pooling as a hyperparameter which varies between models.

B.11. Model Training and Hyperparameter Tuning

All deep learning models were implemented in PyTorch and trained on NVIDA P100 GPUs. Models were hyperparameter tuned using the HyperOpt python package, with Tree-structured Parzen Estimators selected as the optimization algorithm. The optimization algorithm employed for training was the PyTorch implementation of AdamW. A wide range of scalar hyperparameters were explored: learning rate, batch size, drop rate, number of encoder frozen layers, weight decay, betas of the AdamW optimization algorithm, hidden dimension size between dense layers, as well as binary options: use of universal vs. average pooling, normalization of the target variable, and use of a dynamic learning rate. We evaluated anywhere between 100-1000 sets of hyperparameters for each model with 5-fold cross-validation (testing each selection of

hyperparameters on all five folds and averaging the accuracies), with large models trained on larger datasets residing in the lower part of that range due to time constraints. At the end of hyperparameter tuning we took the hyperparameter set with the highest average score across all five folds as the best set. We then ensembled the predictions from each of the five models with this hyperparameter set (each model was trained on one of the five folds) on an unseen holdout set to derive a final accuracy. These values are reported in **Table 1**.

B.12. One-Hot Encoding and Embedding Baselines

The two sequence-based baselines evaluated in this study were random forest classifiers or regressors using either one-hot encoding or the language embedding of the sequences as features. Several studies presenting performance of language models on downstream tasks neglect to show a one-hot encoding baseline, which can outperform language embeddings in practice despite it being faster to compute $O(N)$ vs $O(N^2)$ (where N is the length of the MSA) for each sequence and requiring a few lines of simple code (Yang et al., 2022a). The one-hot encoding is derived by encoding each amino-acid as a 21-length-vector where the *i'th* position corresponds to each of the 20 canonical amino acids, with the final position in each vector corresponding to a gap in the MSA. Letting *L* be the length of the MSA, these vectors were then concatenated to form a single *L x 21* length feature input to represent each sequence. Embeddings were derived from the final hidden layer in the encoder before the amino-acid prediction head of AbLang. Letting *H* be the hidden/embedding dimension size, this hidden layer has shape *L x H*. The average value was taken across the hidden dimension to derive a single sequence vector of shape *L* as recommended in Olsen et. al (2022) . Averaging and max-pooling across both dimensions, *L* and *H*, were attempted but this method seemed to yield the best accuracy for our datasets. These inputs were then used as features for a random Forest regressors (for HIC-RRT, HIC-RT, T-Agg, PSR, and T-Mid) and classifiers (for MERS, MRGX and Al-Base). These models were trained on the same training data and tested on the same holdout data as the deep learning models. The accuracies we report have error bars reflecting 95% confidence intervals for the mean performance derived from 10 trials for each dataset with different random forest number of trees and random states.

C. Supplemental Figures and Tables

Table 2. Effect of MLM Pretraining on Downstream Performance. As with **Figure 3A**, error bars show 95% confidence interval for mean Spearman coefficient or classification accuracy from 20 trials, with hyper parameters varying between trials. Average scores which are significantly greater by at least 80% certainty are in bold, and same as **Figure 3A**, *, **, ***, **** correspond to 80%, 90%, 95% and 99% certainty from a difference of means t-test.

DATA SET	NAIVE	PRETAINED	P-VALUE
MERS	61.7± 5.8	**67.6± 0.4**	0.040746***
AL BASE	88.0± 0.9	**89.6± 0.5**	0.012714***
HIC-RRT	0.59± 0.15	**0.74± 0.14**	0.122316*
PSR	0.16± 0.07	0.18± 0.08	0.370891
HIC-RT	0.14± 0.13	**0.35± 0.18**	0.071298**
T-MID	0.25± 0.08	**0.42± 0.05**	0.007776****
T-AGG	0.21± 0.12	**0.47± 0.17**	0.019808***
MRGX	59.1± 1.3	**61.3± 0.5**	0.011503***

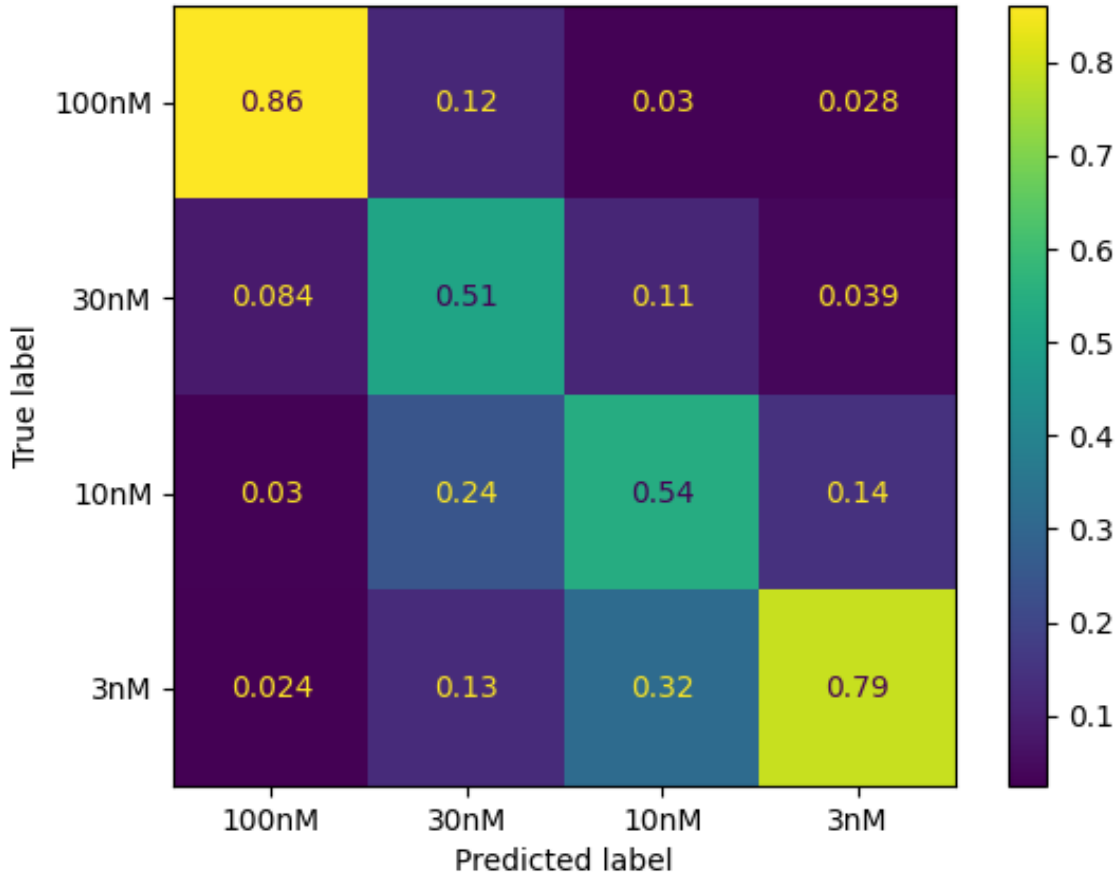

Figure 5. Confusion Matrix for Best Model's Performance on MERS Binding Dataset

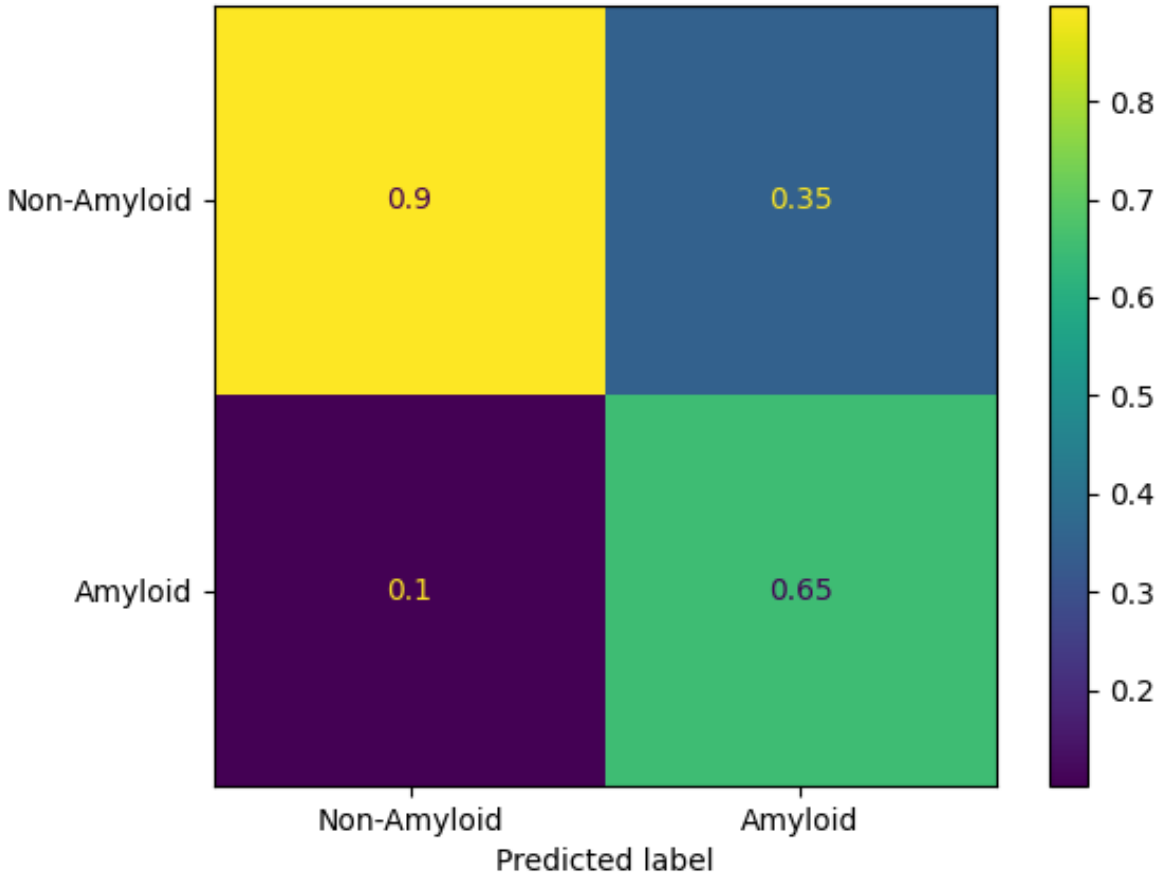

Figure 6. Confusion Matrix for Best Model's Performance on AL-Base Dataset

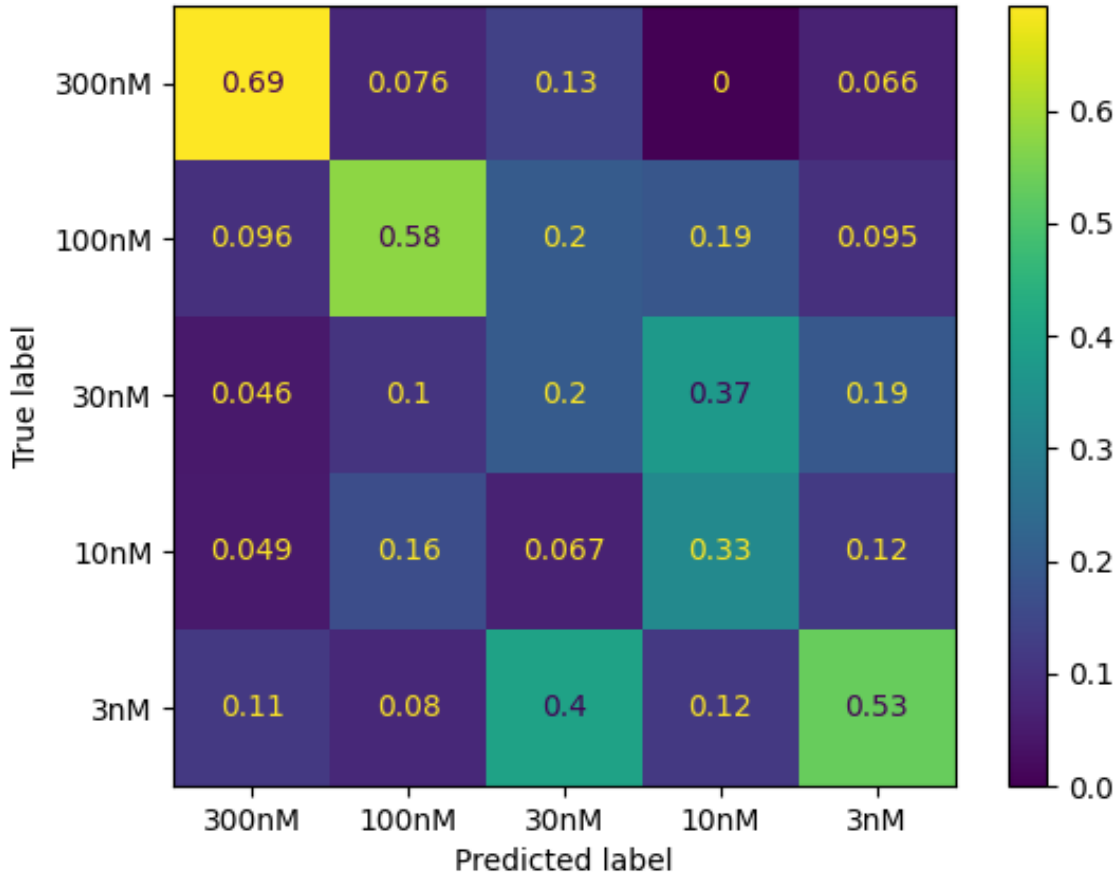

Figure 7. Confusion Matrix for Best Model's Performance on MRGX Binding Dataset

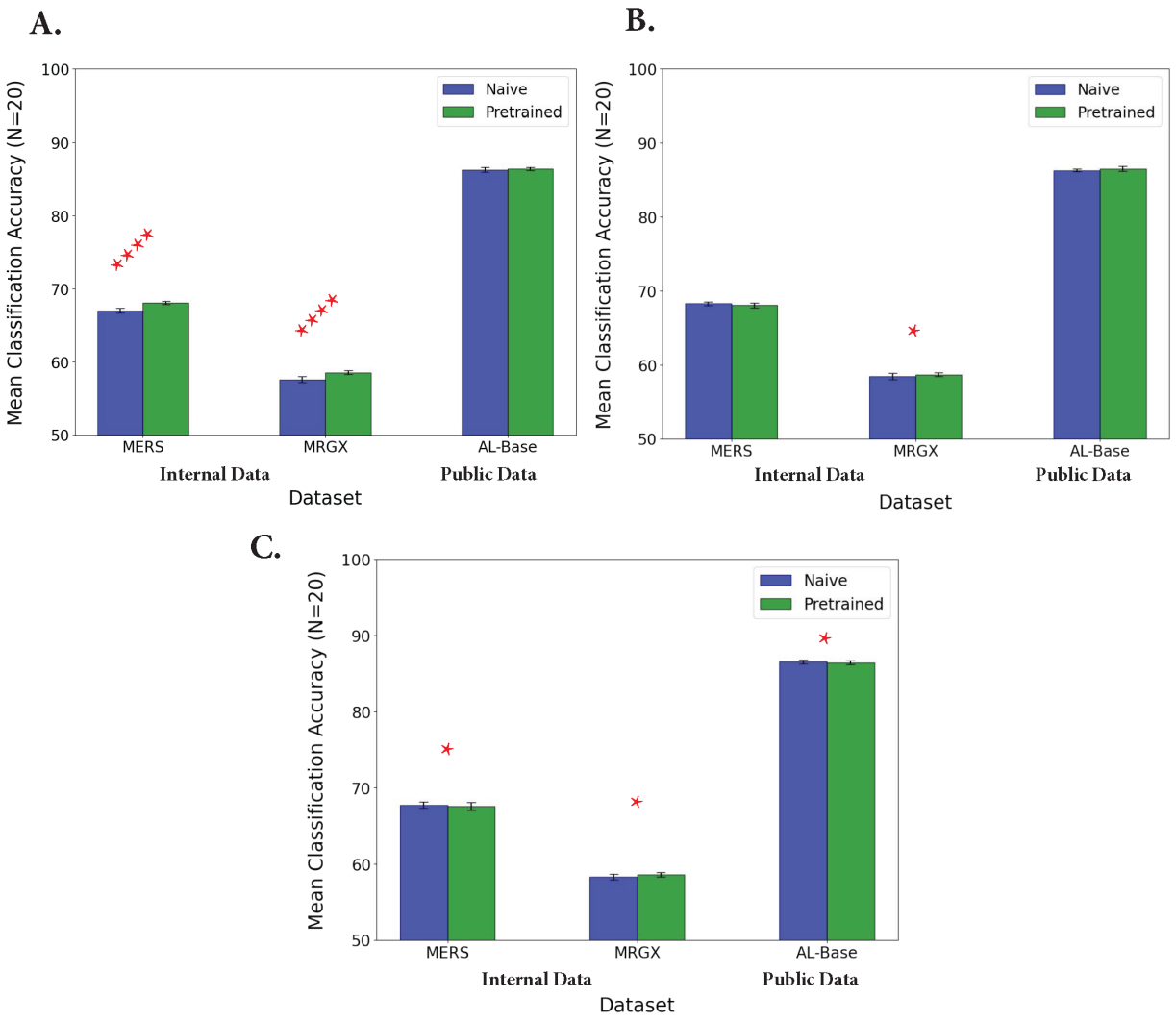

Figure 8. Effect of MLM Sequence Pretraining for A. AbPROP-Seq and Structural Inverse Folding Pretraining for B. AbPROP-GVP and C. AbPROP-SGN on property prediction performace. Error bars show a 95% confidence interval for the mean perfect classification accuracy from n = 20 trials, with each trial having a random set of hyper-parameters for both naive and pretrained. We include p-value derived certainty from difference of means t-tests. *, **, ***, **** correspond to 80%, 90%, 95% and 99% certainty, respectively.

Table 3. Effect of Structural Pretraining on Downstream Performance for Geometric Vector Perceptron (GVP). As with **Figure 3B**, error bars show 95% confidence interval for mean Spearman coefficient or classification accuracy from 20 trials, with hyper parameters varying between trials. Average scores which are significantly greater by at least 80% certainty are in bold, and same as **Figure 3B**, *, **, ***, **** correspond to 80%, 90%, 95% and 99% certainty from a difference of means t-test.

DATA SET	NAIVE	PRETAINED	P-VALUE
MERS	63.4± 3.9	63.8± 3.6	0.446
AL BASE	**90.0± 0.6**	88.2± 1.6	0.048***
HIC-RRT	0.61± 0.10	0.61± 0.12	0.491
PSR	0.24± 0.05	0.25± 0.06	0.395
HIC-RT	0.25± 0.05	**0.32± 0.13**	0.195*
T-MID	0.36± 0.13	0.36± 0.06	0.472
T-AGG	0.30± 0.12	**0.43± 0.08**	0.080**
MRGX	61.0± 0.4	61.0± 0.05	0.462

Table 4. Effect of Structural Pretraining on Downstream Performance for Structured Graph Neural Net (SGN). As with **Figure 3C**, error bars show 95% confidence interval for mean Spearman coefficient or classification accuracy from 20 trials, with hyper parameters varying between trials. Average scores which are significantly greater by at least 80% certainty are in bold, and same as **Figure 3C**, *, **, ***, **** correspond to 80%, 90%, 95% and 99% certainty from a difference of means t-test.

DATA SET	NAIVE	PRETAINED	P-VALUE
MERS	64.3± 6.0	**68.3± 0.5**	0.141*
AL BASE	89.5± 0.7	89.6± 1.4	0.467
HIC-RRT	0.36± 0.10	**0.56± 0.08**	0.011**
PSR	0.22± 0.07	0.20± 0.06	0.343
HIC-RT	0.25± 0.12	**0.38± 0.12**	0.114*
T-MID	0.32± 0.06	**0.41± 0.07**	0.061**
T-AGG	0.26± 0.14	0.24± 0.11	0.396
MRGX	**61.5± 0.1**	61.2± 0.3	0.096**