Supporting Online Material

Robert Schmirler[*] [1] [2] [3] **Michael Heinzinger**[2] [3] **Burkhard Rost**[2] [4] [5]

1 Method Details

Here we give more details on datasets, models and model training.

1.1 Datasets

Table S1: Datasets

Prediction	Sequence	Dataset	# of Sequences			Average Seq.
Level	Diversity		Train	Validation	Test	Length
per Protein	Mutational Landscapes	GFP	21446	5362	27217	237.0
		AAV	28626	3181	50776	736.3
		GB1	2691	299	5743	265.0
	Diverse Datasets	Stability	53614	2512	12851	45.0
		Meltome	22335	2482	3134	544.5
		Sub. Loc.	9503	1678	490	519.9
per Residue		Disorder	1056	118	117	118.1
		Sec. Str.	9712	1080	364	255.0

We subdivided the datasets according to two aspects, prediction task level and sequence diversity (Table 1). The prediction task level indicates whether a prediction is either done for each residue in a protein individually, e.g. secondary structure, or for each protein, e.g. subcellular localization. The sequence diversity field distinguishes between datasets containing multiple, diverse proteins from datasets containing mutational landscapes of a single protein which are commonly generated in deep mutational scanning experiments [1].

Mutational landscapes. This subgroup comprises fitness landscapes for the green fluorescent protein (GFP), the adeno-associated virus 2 (AAV2) capsid protein VP-1 and the GB1 binding domain of the Protein G. All three are per protein regression tasks that measure prediction performance by ranking the correlation between the predicted and the experimentally measured property on the respective test sets. For the GFP dataset this means that the property/fitness is a measure of fluorescence intensity, with experimental data being generated by Sarkisyan et al. [2] and the data used here being split by Rao et al. [3]. Training and validation set sequences are all within Hamming distance 3 of the wildtype sequence. Sequences in the test set show four or more mutations. Fitness for the AAV task measures viability for packaging of a DNA payloads. Bryant et al. [4] mutated a 28-amino acid window to create the original data. We used the “2-vs-rest” data split from the FLIP benchmark [5]. This means sequences up to two mutations from wildtype are in the training and

[*] Correspondence to: Robert Schmirler <robert.schmirler@abbvie.com> [1] Innovation Center, BTS IR LU, AbbVie Deutschland GmbH & Co. KG, Ludwigshafen, Germany [2] Faculty of Informatics, TUM (Technical University of Munich), Munich, Germany [3] TUM School of Computation, Information and Technology (CIT), TUM Graduate School, Garching, Germany [4] Institute for Advanced Study (TUM-IAS), TUM, Garching, Germany [5] TUM School of Life Sciences Weihenstephan (WZW), TUM, Freising, Germany

3rd NeurIPS Workshop on Efficient Natural Language and Speech (ENLSP 2023).

validation set, while all variants with more mutations belong to the test set. The fitness score for the GB1 binding domain landscape represents a combined measure of stability and binding affinity. In the original experiment Wu et al. [6] mutate four positions, the „three-vs-rest“ data split is also taken from FLIP which is a Hamming distance 3 split like the GFP dataset.

Diverse datasets, per protein prediction. The diverse per protein subgroup also consists of three prediction tasks. The first two datasets focus on stability prediction formulated as regression tasks alike the fitness landscapes above. Stability uses the experimental dataset by Rocklin et al. [7] which measured protease susceptibility of de novo designed miniproteins. We reused the data split from Rao et al. [3], where training and validation sets consist of sequences from four design cycles while the test set holds seventeen 1-Hamming distance neighborhoods to promising candidates. It simulates a common challenge in guided protein design. Meltome utilizes data from the meltome atlas [8], which measures thermostability for proteins from 13 species. We used the "mixed" split from the FLIP benchmark, which divides the entire dataset in clusters using MMseqs [9] with a 20% sequence identity threshold. This allows to study generalization capabilities by minimizing information leakage between training/validation data (80% of the clusters) and the test set (remaining 20% of the clusters). The third dataset is based on the DeepLoc [10] data for training with the training/validation split and a novel test dataset ("setHARD") curated by Stärk et al. [11]. The task for this dataset is subcellular localization prediction, presented as a 10-class multiclass prediction problem. Again, MMseqs was used to remove all sequences with more than 20% pairwise sequence similarity to the training and validation data to minimize information leakage to the test dataset.

Diverse datasets, per residue prediction. The disorder dataset is based on the CheZOD [12] dataset which uses nuclear magnetic resonance spectroscopy data. We use the data splits generated by Ilzhöfer et al. [13] which apply MMseqs clustering at 20% identify threshold to reduce sequence similarity of the test set to the training data. The task is to predict CheZOD scores [14] which resemble a continuous scale to quantify the level of disorder of each individual residue in a protein. Finally, we also benchmarked the commonly used secondary structure prediction task using data from "NetSurfP-2.0 [15], to classify for each residue in a protein whether it is either helix (H), strand (E), or other/random coil (C)). We reused the redundancy reduced split created by Elnaggar et al. [16] including their "NEW364" test set. Both datasets also have binary flags indicating for each residue whether they were experimentally resolved and should be used for training and evaluation.

1.2 Models

All models used in this work are listed in Table 2 including their corresponding Huggingface checkpoint.

Table S2: Models

Model	Architecture (Pretraining)	Number of Parameters (Encoder)	Encoder Layers	Emb. Size	Huggingface Model Checkpoint*
Ankh Base	Enc-Dec	736 M	48	768	ankh-base
Ankh Large		1.9 B	48	1536	ankh-large
ProtT5		1.2 B	24	1024	prot_t5_xl_uniref50
ProstT5		1.2 B	24	1024	ProstT5
ESM2 8M	Enc	8 M	6	320	esm2_t6_8M_UR50D
ESM2 35M		35 M	12	480	esm2_t12_35M_UR50D
ESM2 150M		150 M	30	640	esm2_t30_150M_UR50D
ESM2 650M		650 M	33	1280	esm2_t33_650M_UR50D
ESM2 3B		3.0 B	36	2560	esm2_t36_3B_UR50D

* to get the complete checkpoint name usable in transformers, query this name on https://huggingface.co/

1.3 Model Training

Intra-model evaluation. We chose the following method to compare pretrained with finetuned models: For the pretrained embedding results we generated embeddings for all mentioned datasets. For per protein tasks, we took the average over the sequence length which results in a 1 x embedding_size vector per protein. For per residue tasks all valid residue embeddings and their respective label were used. This also leads to a vector of size 1 x embedding_size per residue. Afterwards we trained a single fully connected layer of size 32, which takes these embeddings as input and outputs either a single value (regression) or is followed by an output layer with one neuron for each possible output class, followed by a softmax layer to get a probability distribution. Training was done until training loss did not reduce any further. Each individual training was run 5 times with different random seed initialization.

For finetuning we put the same fully connected layer (size 32) on top of the language model encoder as a prediction head. For ProtT5 and Ankh we also used average pooling of the last hidden states over the sequence length dimension during training on per protein tasks. ESM2 suggests connecting the prediction head only to the very first token of the sequence (special token "<CLS>") which we therefore applied. Each individual training was run three times with different random seed initialization. Training was also run until training and validation loss flattened out. Due to hardware constaints we applied PEFT [17] to the T5 based models ProtT5 and Ankh. We used LoRA [18] which is a well-established PEFT method. For the ESM2 models we finetuned all model weights.

Early during evaluation, we noticed large variance in test performance within the same experiment setup but between different random seeds. We found that for some datasets validation and test performance do not correlate well (SOM 3). If early stopping is applied based on the validation set this introduces large variance in test performance. To avoid introducing a large random variance into our comparison we decided to directly measure test performance during the entire training period and take an average of the best 10 single measures. This can be seen as a measure of the model's theoretical upper bound test performance. These performance values should therefore not be taken as benchmark value or basis for comparing to other work but are only valid for our intra-model comparison.

For both, embedding and finetuning experiments we did a limited hyperparameter optimization at the beginning of our study. The selected hyperparameters were left unchanged for most of the comparison. Parameters selected for each individual experiment can be found in Table 3 and 4. We used the transformers Adam optimizer for all experiments with it's standard parameters. To realize the batch size for model finetuning we applied gradient accumulation as needed with the given hardware. All training runs were performed on single NVIDIA A10G GPUs with 24GB.

Table S3: Training Parameters - pretrained embeddings

Dataset	Epochs	Validation per epoch	learning rate	batch size
GFP	240	1	1e-04	8 sequences
AAV	120	1	1e-04	8 sequences
GB1	240	1	1e-04	8 sequences
Stability	120	1	1e-04	8 sequences
Meltome	120	1	1e-04	8 sequences
Sub. Loc.	120	1	1e-04	8 sequences
Disorder	50	1	1e-04	8 residues
Sec. Str.	10	1	1e-04	8 residues

Disorder prediction. We trained two finetuned model variants to compare them with previous results [13]. SETH LoRA is based on ProtT5 embeddings and utilizes the same two-layer CNN from the original SETH model. During previous experiments ESM2 150M performed best on the disorder dataset among the ESM2 models (Table S8). Therefore we also investigated it in this experiment. We reused the ESM2 setup from the intra-model evaluation here (last hidden states of "<CLS>" token with single dense layer prediction head).

For both variants, we each trained five models, initialized with different random seeds, for ten epochs. We calculated validation loss twice per epoch during training and selected the model with the lowest

Table S4: Training Parameters - finetuning

Model	Dataset	Epochs	Validation	learning	batch
			per epoch	rate	size
ESM2	GFP	20	5	2e-05	8 sequences
	AAV	10	5	2e-05	8 sequences
	GB1	20	5	2e-05	8 sequences
	Stability	10	10	2e-05	8 sequences
	Meltome	10	10	2e-05	8 sequences
	Sub. Loc.	10	10	2e-05	8 sequences
	Disorder*	20	10	2e-5 / 2e-4	1 / 8 residues
	Sec. Str.	5	20	2e-05	1 residue
Ankh	GFP	50	1	3e-04	8 sequences
	AAV	20	1	3e-04	8 sequences
	GB1	50	1	3e-04	8 sequences
	Stability	50	1	3e-04	8 sequences
	Meltome	10	2	3e-04	8 sequences
	Sub. Loc.	10	10	3e-04	8 sequences
	Disorder	20	10	3e-04	1 residue
	Sec. Str.	5	20	3e-04	1 residue
ProtT5	GFP	50	1	3e-04	8 sequences
	AAV	20	1	3e-04	8 sequences
	GB1	50	1	3e-04	8 sequences
	Stability	50	1	3e-04	8 sequences
	Meltome	20	1	3e-04	8 sequences
	Sub. Loc.	5	10	3e-04	8 sequences
	Disorder	20	10	3e-04	1 residue
	Sec. Str.	5	20	3e-04	1 residue

* for the disorder dataset, ESM2 150M and 650M did not show stable conversion with the standard parameters and we had to increase learning rate and batch size to 2e-4 and 8 residues

validation loss out of the 100 available checkpoints (5 random seeds, 10 epochs, 2 validations per epoch). For the selected models we estimated test performance and confidence intervals using bootstrapping.

Sub-cellular Localization prediction. For the 10 class multi-class classification task, we reused the single layer dense network from our intra-model evaluation. We trained five models using different random seeds for five epochs. Validation loss was calculated twice per epoch during training. From each individual run we selected the model with the lowest validation loss. We then calculated Q10 accuracy and standard deviation for all five selected models on the "set_hard" [11] and report the average values.

Secondary Structure. Again five models were finetuned for both, the T5 based ProtT5 [16] and ProstT5 [19] models, initializing training with different random seeds. For ProstT5 we added the prefix token "<fold2aa>" to each sequence, to make the model aware of the input type it is receiving. Training was performed for 5 epochs and validation loss was calculated at the end. For both models a two-layer CNN prediction head was used, to keep results comparable with the raw embedding predictions. We selected the model with the lowest validation loss from these five runs and measured model performance on the previously established CAPS12 [20] and NEW364 [16] datasets. The mean accuracy values and confidence intervals were estimated using bootstrapping.

2 Detailed results - Intra-model comparison

Here we provide the results of all individual training runs for useing (frozen) pretrained embeddings (Table 5) and finetuning Table 6. These results are also available in their aggregated form with 95% confidence intervals in tables 7-10.

2.1 Individual predictor results

Table S5: Individual training runs - pretrained embeddings

Model	Rand. Seed	Stab.	GFP	AAV	GB1	Melt.	Sub. Loc.	Dis.	Sec. Str.
ESM2 8M	99	74,2%	64,0%	69,2%	83,2%	57,7%	52,6%	70,2%	75,3%
	98	76,0%	63,9%	68,3%	83,0%	57,8%	53,3%	70,1%	75,2%
	97	76,7%	64,0%	68,5%	82,8%	57,8%	53,2%	69,6%	75,3%
	96	77,6%	64,4%	68,2%	83,0%	57,3%	52,6%	70,0%	75,3%
	95	78,6%	64,3%	68,6%	82,4%	57,5%	52,2%	70,1%	75,2%
ESM2 35M	99	74,5%	65,1%	64,5%	82,6%	58,7%	55,4%	68,8%	78,2%
	98	73,2%	65,2%	64,6%	83,4%	59,2%	55,4%	69,1%	78,2%
	97	73,4%	64,7%	60,3%	83,0%	58,9%	58,0%	69,2%	78,2%
	96	76,1%	64,9%	61,3%	82,8%	59,2%	55,8%	69,0%	78,2%
	95	73,0%	65,2%	63,6%	83,5%	58,8%	56,1%	69,1%	78,2%
ESM2 150M	99	81,1%	64,0%	67,8%	85,0%	61,8%	60,0%	71,5%	82,1%
	98	78,0%	64,0%	67,2%	84,6%	62,7%	59,9%	71,2%	82,1%
	97	78,5%	64,0%	64,3%	84,7%	62,9%	61,6%	71,4%	82,1%
	96	78,3%	64,3%	67,2%	83,7%	62,5%	59,9%	71,2%	82,1%
	95	82,1%	63,9%	68,4%	84,3%	63,0%	60,3%	71,5%	82,1%
ESM2 650M	99	73,2%	64,9%	64,5%	86,6%	66,2%	63,4%	72,3%	84,7%
	98	69,1%	64,8%	61,2%	86,6%	67,0%	64,5%	72,1%	84,6%
	97	69,3%	64,6%	60,3%	86,7%	67,1%	63,7%	72,3%	84,7%
	96	68,1%	64,8%	59,8%	85,7%	66,6%	63,6%	72,3%	84,6%
	95	73,7%	64,9%	67,5%	86,0%	66,1%	64,4%	72,3%	84,6%
ESM2 3B	99	79,2%	64,8%	77,0%	85,4%	67,1%	63,4%	71,1%	85,4%
	98	76,9%	64,8%	76,9%	86,5%	67,3%	63,0%	70,7%	85,5%
	97	76,8%	64,9%	76,6%	86,8%	67,1%	64,6%	71,0%	85,4%
	96	76,8%	65,1%	76,7%	86,6%	67,5%	63,6%	70,4%	85,5%
	95	78,1%	64,9%	77,3%	86,3%	67,0%	64,0%	70,4%	85,5%
ProtT5	99	79,4%	61,6%	72,0%	86,3%	69,9%	62,2%	71,5%	84,2%
	98	79,3%	61,5%	73,2%	86,2%	70,1%	63,9%	71,2%	84,2%
	97	79,5%	61,1%	72,3%	86,4%	70,9%	60,3%	71,4%	84,2%
	96	81,2%	62,1%	71,5%	85,6%	70,3%	60,8%	71,0%	84,2%
	95	79,3%	61,6%	73,0%	85,6%	70,4%	62,4%	71,1%	84,2%
Ankh base	99	79,8%	66,0%	65,7%	85,4%	58,1%	60,9%	71,0%	84,3%
	98	80,1%	66,0%	64,4%	85,1%	58,6%	61,8%	71,0%	84,3%
	97	78,5%	65,9%	64,2%	85,3%	58,5%	61,9%	70,8%	84,3%
	96	81,3%	66,0%	66,8%	85,4%	58,8%	60,4%	71,4%	84,3%
	95	80,5%	66,3%	69,6%	84,4%	57,9%	60,4%	70,8%	84,3%
Ankh large	99	74,4%	67,2%	74,8%	86,3%	62,2%	61,7%	70,0%	86,0%
	98	74,8%	67,1%	74,2%	86,7%	62,7%	60,2%	70,2%	85,9%
	97	73,3%	66,8%	73,6%	86,7%	63,3%	60,8%	70,1%	86,0%
	96	76,8%	67,2%	74,0%	86,2%	61,6%	62,2%	70,0%	86,0%
	95	79,7%	67,1%	75,2%	86,6%	63,0%	61,8%	70,1%	86,0%

For finetuning (Table 6) we generally did three reruns for each experiment. The only exception was a clear outlier for ESM2 150M, AAV, random seed 98 which we therefore reran (with random seed 96). The 69,2% result was excluded for all further calculations. For ProtT5 unrelated experiments not published here resulted in additional values for three of the datasets (Subcellular Localization, Disorder and Secondary Structure prediction), which we included as well.

Table S6: Individual training runs - finetuning

Model	Rand. Seed	GFP	AAV	GB1	Stab.	Melt.	Sub. Loc.	Dis.	Sec. Str.
ESM2 8M	99	68,5%	83,6%	89,1%	79,6%	58,8%	58,2%	72,4%	76,1%
	98	69,0%	83,6%	88,2%	78,9%	59,1%	58,4%	72,2%	76,1%
	97	69,1%	82,1%	87,8%	80,9%	60,3%	57,9%	72,6%	76,1%
ESM2 35M	99	69,2%	79,2%	87,7%	81,1%	60,2%	59,6%	74,0%	79,0%
	98	69,0%	83,0%	88,4%	80,6%	60,6%	60,7%	74,7%	79,1%
	97	69,2%	83,7%	88,3%	81,2%	62,3%	59,9%	73,7%	79,1%
ESM2 150M	99	69,6%	85,5%	88,6%	83,1%	65,0%	63,2%	74,2%	82,8%
	98	69,1%	69,2%	88,1%	81,9%	63,7%	62,8%	74,0%	82,8%
	97	69,1%	84,8%	88,7%	82,4%	63,5%	62,7%	75,0%	82,7%
	96	-	84,4%	-	-	-	-	-	-
ESM2 650M	99	68,9%	78,3%	87,7%	83,0%	68,5%	65,9%	73,9%	85,5%
	98	69,2%	82,6%	89,0%	82,6%	69,6%	64,8%	73,4%	85,5%
	97	68,8%	77,9%	88,7%	82,4%	67,5%	64,5%	73,7%	85,6%
ProtT5	99	68,7%	76,5%	88,0%	81,7%	72,4%	66,6%	71,5%	85,0%
	98	69,1%	77,3%	87,7%	83,4%	72,7%	65,9%	72,4%	84,9%
	97	69,1%	74,6%	88,4%	84,7%	72,0%	66,3%	72,8%	84,9%
	96	-	-	-	-	-	65,9%	72,6%	84,9%
	95	-	-	-	-	-	65,2%	71,5%	84,9%
Ankh base	99	69,6%	83,3%	86,7%	80,1%	61,1%	62,4%	69,5%	84,0%
	98	69,6%	81,2%	87,7%	83,2%	60,5%	61,8%	68,8%	84,0%
	97	69,7%	79,2%	86,3%	80,5%	60,2%	61,4%	70,2%	84,0%
Ankh large	99	69,8%	84,3%	88,0%	80,7%	57,1%	59,8%	70,8%	85,8%
	98	69,7%	84,5%	89,3%	81,5%	57,4%	62,6%	68,4%	85,7%
	97	69,9%	85,4%	89,2%	83,0%	63,4%	62,1%	70,0%	85,7%

2.2 Aggregated results

Table S7: ESM2 - pretrained embeddings

	ESM2 8M	ESM2 35M	ESM2 150M	ESM2 650M	ESM2 3B
GFP	64,1% ± 0,19	65,0% ± 0,17	64,0% ± 0,13	64,8% ± 0,10	64,9% ± 0,11
AAV	68,6% ± 0,35	62,8% ± 1,69	67,0% ± 1,40	62,7% ± 2,87	76,9% ± 0,25
GB1	82,9% ± 0,27	83,1% ± 0,34	84,5% ± 0,42	86,3% ± 0,39	86,3% ± 0,49
Stability	76,6% ± 1,45	74,0% ± 1,15	79,6% ± 1,62	70,7% ± 2,27	77,6% ± 0,93
Meltome	57,6% ± 0,18	59,0% ± 0,19	62,6% ± 0,43	66,6% ± 0,39	67,2% ± 0,17
Sub. Loc.	52,8% ± 0,39	56,1% ± 0,97	60,3% ± 0,64	63,9% ± 0,45	63,7% ± 0,53
Disorder	70,0% ± 0,18	69,1% ± 0,15	71,4% ± 0,12	72,3% ± 0,09	70,7% ± 0,30
Sec. Str.	75,2% ± 0,03	78,2% ± 0,01	82,1% ± 0,02	84,6% ± 0,04	85,5% ± 0,02

Table S8: ESM2 - finetuning

	ESM2 8M	ESM2 35M	ESM2 150M	ESM2 650M	ESM2 3B
GFP	68,9% ± 0,36	69,1% ± 0,14	69,2% ± 0,31	69,0% ± 0,26	-
AAV	83,1% ± 1,00	82,0% ± 2,76	84,9% ± 0,62	79,6% ± 2,94	-
GB1	88,4% ± 0,78	88,2% ± 0,45	88,5% ± 0,38	88,5% ± 0,77	-
Stability	79,8% ± 1,10	81,0% ± 0,37	82,5% ± 0,70	82,7% ± 0,36	-
Meltome	59,4% ± 0,88	61,1% ± 1,30	64,1% ± 0,91	68,5% ± 1,14	-
Sub. Loc.	58,2% ± 0,29	60,1% ± 0,62	62,9% ± 0,33	65,1% ± 0,82	-
Disorder	72,4% ± 0,21	74,2% ± 0,59	74,4% ± 0,55	73,7% ± 0,27	-
Sec. Str.	76,1% ± 0,04	79,1% ± 0,06	82,8% ± 0,05	85,5% ± 0,03	-

We were not able to finetune the ESM2 3B model (Table 8) on available hardware due to GPU memory constraints. Therefore it is not included in the intra-model comparison heatmap.

Table S9: T5 models - pretrained embeddings

	ProtT5	Ankh base	Ankh large
GFP	61,6% ± 0,29	66,1% ± 0,14	67,1% ± 0,15
AAV	72,4% ± 0,61	66,1% ± 1,92	74,4% ± 0,57
GB1	86,0% ± 0,34	85,1% ± 0,36	86,5% ± 0,23
Stability	79,8% ± 0,74	80,0% ± 0,89	75,8% ± 2,20
Meltome	70,3% ± 0,34	58,4% ± 0,30	62,6% ± 0,58
Sub. Loc.	61,9% ± 1,25	61,1% ± 0,64	61,4% ± 0,73
Disorder	71,3% ± 0,16	71,0% ± 0,20	70,1% ± 0,09
Sec. Str.	84,2% ± 0,01	84,3% ± 0,02	86,0% ± 0,02

Table S10: T5 models - finetuning

	ProtT5	Ankh base	Ankh large
GFP	69,0% ± 0,30	69,6% ± 0,08	69,8% ± 0,13
AAV	76,1% ± 1,54	81,2% ± 2,32	84,7% ± 0,70
GB1	88,0% ± 0,40	86,9% ± 0,78	88,8% ± 0,79
Stability	83,3% ± 1,70	81,3% ± 1,94	81,7% ± 1,34
Meltome	72,4% ± 0,42	60,6% ± 0,49	59,3% ± 4,00
Sub. Loc.	66,0% ± 0,45	61,9% ± 0,58	61,5% ± 1,71
Disorder	72,2% ± 0,52	69,5% ± 0,82	69,8% ± 1,41
Sec. Str.	84,9% ± 0,03	84,0% ± 0,05	85,7% ± 0,05

3 Figures - Finetuning

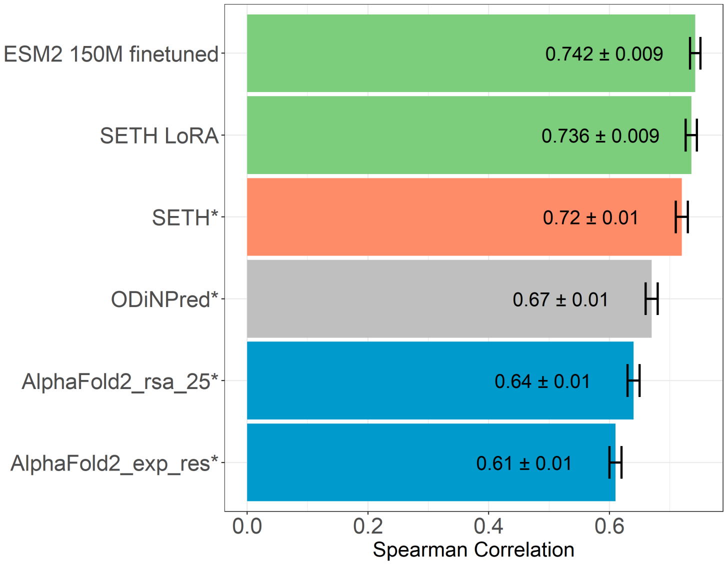

Figure S1: **Disorder prediction improved.** Intrinsically disordered residues can be described by so-called CheZOD scores [12]. The x-axis shows the Spearman correlation between experimental and predicted CheZOD scores, for five different methods. Values marked by asterisks (*) taken from the literature [13]. Our results shown in green, previous results from [13] in orange (pLM-based without MSA) and blue (MSA-based) and the MSA-based SOTA in gray [14, 12]. For each fine-tuned model we trained five models with different random seeds. We select the model checkpoint corresponding to the lowest validation loss. Error bars mark the 95% confidence intervals (CI), estimated via bootstrapping.

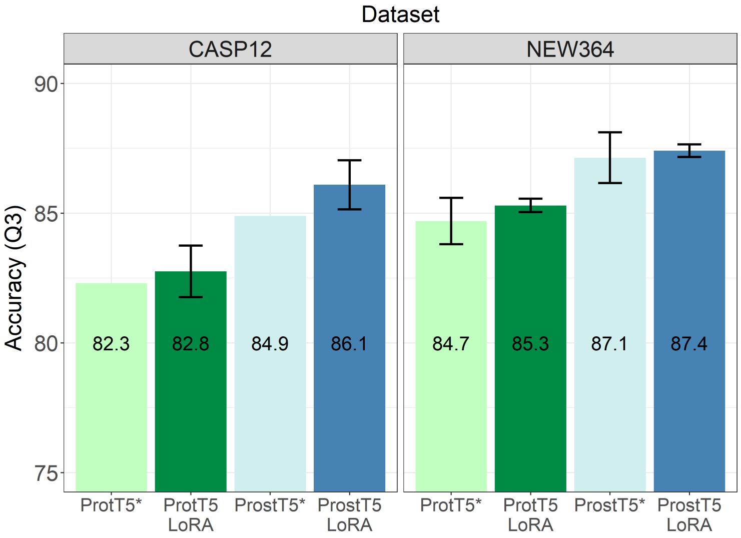

Figure S2: **Secondary structure prediction hardly affected.** Values for the pre-trained models (ProtT5 [16] and ProstT5 [19]) taken from literature [19] and marked by (*). We included two previously used test data sets (CASP12 [20] and NEW364 [16]). For each fine-tuned model (ProtT5-LoRA and PostT5-LoRA) we trained five models with different random seeds and selected the model with the lowest validation error after the 5th epoch. Error bars mark the 95% confidence intervals, estimated via bootstrapping (values not available for pre-trained CASP12 results).

4 Validation set issues

As mentioned earlier, instead of selecting the best performing models using the validation set, we measured performance directly on the test set for the intra-model comparison. Figure S3 shows the same experimental results when early stopping on the validation set is applied instead.

We saw three different kind of behaviours during training:

First we encountered noisy test performance, i.e. the test loss and performance metric are not converging cleanly but stay fluctuating even though training and validation loss flatten out smoothly. This occurred for the Stability and AAV datasets.

Second, for some tasks we see over-fitting. This occurs for Meltome, Subcellular Localization and Disorder prediction and is normal behaviour. But for two of those three (Disorder and to a lesser extend Meltome) the validation loss does not reflect this. Which leads to an inability to early stop at the correct moment.

The third very forgiving behaviour is clean convergence without overfitting. This happens for the GFP, GB1 and secondary structure tasks. Here training can be continued beyond initial convergence without any effect on test performance.

This is reflected in the heatmap (Figure S3). Values for AAV and Stability change significantly due to a basically random selection from their noisy convergence range. Because both, embedding based predictions and finetuning, are affected similarly this leads to random increases/decreases. For Meltome and Disorder the difference values gets smaller, because finetuned models simply have a higher potential to overfit (because more parameters are updated).

Everything mentioned above was found for these datasets using the data splits mentioned. We also only performed a limited general hyperparameter search and did not optimize those for each individual task. It is possible, that more optimal hyperparameters can mitigate some unwanted behaviour.

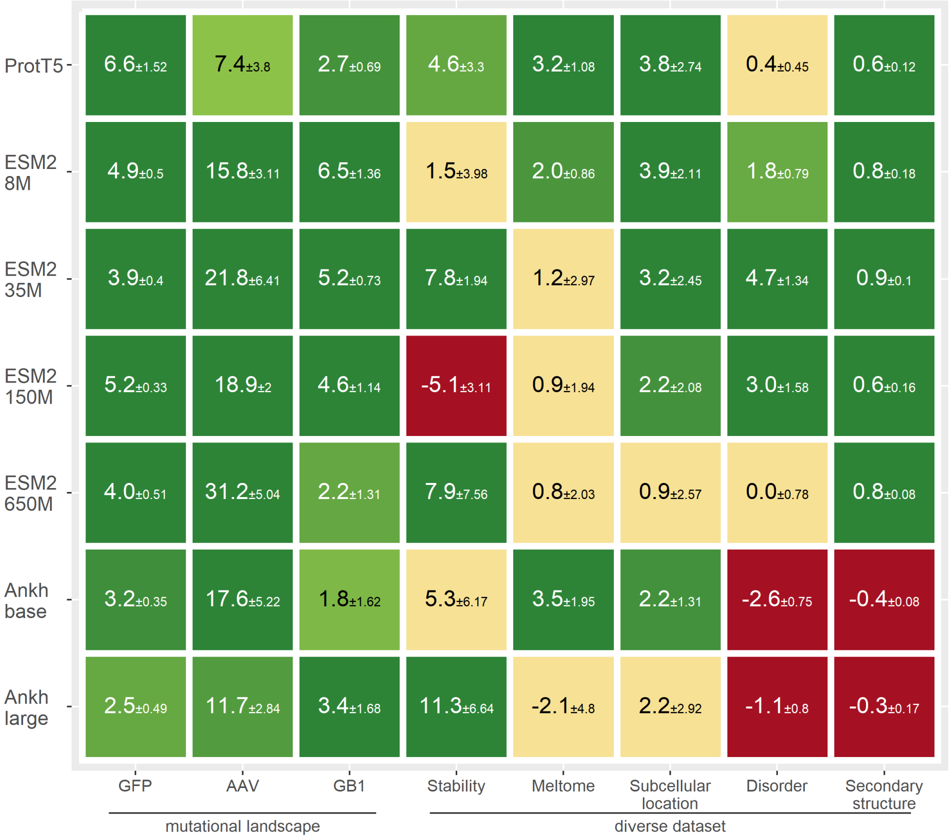

Figure S3: **Comparison of models and tasks - with early stopping**

References

[1] Douglas M. Fowler and Stanley Fields. Deep mutational scanning: a new style of protein science. *Nature methods*, 11(8):801–807, 2014.

[2] Karen S Sarkisyan, Dmitry A Bolotin, Margarita V Meer, Dinara R Usmanova, Alexander S Mishin, George V Sharonov, Dmitry N Ivankov, Nina G Bozhanova, Mikhail S Baranov, Onuralp Soylemez, et al. Local fitness landscape of the green fluorescent protein. *Nature*, 533(7603):397–401, 2016.

[3] Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. Evaluating protein transfer learning with tape. *Advances in neural information processing systems*, 32:9689–9701, 2019.

[4] Drew H. Bryant, Ali Bashir, Sam Sinai, Nina K. Jain, Pierce J. Ogden, Patrick F. Riley, George M. Church, Lucy J. Colwell, and Eric D. Kelsic. Deep diversification of an aav capsid protein by machine learning. *Nature Biotechnology*, 39(6):691–696, 2021.

[5] Christian Dallago, Jody Mou, Kadina E. Johnston, Bruce J. Wittmann, Nicholas Bhattacharya, Samuel Goldman, Ali Madani, and Kevin K. Yang. Flip: Benchmark tasks in fitness landscape inference for proteins. *bioRxiv*, 2021. doi: 10.1101/2021.11.09.467890.

[6] Nicholas C. Wu, Lei Dai, C. Anders Olson, James O. Lloyd-Smith, and Ren Sun. Adaptation in protein fitness landscapes is facilitated by indirect paths. *Elife*, 5:e16965, 2016.

[7] Gabriel J Rocklin, Tamuka M Chidyausiku, Inna Goreshnik, Alex Ford, Scott Houliston, Alexander Lemak, Lauren Carter, Rashmi Ravichandran, Vikram K Mulligan, Aaron Chevalier, et al. Global analysis of protein folding using massively parallel design, synthesis, and testing. *Science*, 357(6347):168–175, 2017.

[8] Anna Jarzab, Nils Kurzawa, Thomas Hopf, Matthias Moerch, Jana Zecha, Niels Leijten, Yangyang Bian, Eva Musiol, Melanie Maschberger, and Gabriele Stoehr. Meltome atlas—thermal proteome stability across the tree of life. *Nature methods*, 17(5):495–503, 2020.

[9] Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

[10] José Juan Almagro Armenteros, Casper Kaae Sønderby, Søren Kaae Sønderby, Henrik Nielsen, and Ole Winther. Deeploc: prediction of protein subcellular localization using deep learning. *Bioinformatics*, 33 (21):3387–3395, 2017. doi: 10.1093/bioinformatics/btx548.

[11] Hannes Stärk, Christian Dallago, Michael Heinzinger, and Burkhard Rost. Light attention predicts protein location from the language of life. *Bioinformatics Advances*, 1(1):vbab035, 2021. doi: 10.1093/bioadv/vbab035.

[12] Rupashree Dass, Frans AA Mulder, and Jakob Toudahl Nielsen. Odinpred: comprehensive prediction of protein order and disorder. *Scientific Reports*, 10(1):14780, 2020.

[13] Dagmar Ilzhöfer, Michael Heinzinger, and Burkhard Rost. Seth predicts nuances of residue disorder from protein embeddings. *Frontiers in Bioinformatics*, 2, 2022. ISSN 2673-7647.

[14] Jakob T. Nielsen and Frans AA Mulder. Quantitative protein disorder assessment using nmr chemical shifts. *Intrinsically Disordered proteins: methods and protocols*, page 303–317, 2020.

[15] Michael Schantz Klausen, Martin Closter Jespersen, Henrik Nielsen, Kamilla Kjaergaard Jensen, Vanessa Isabell Jurtz, Casper Kaae Soenderby, Morten Otto Alexander Sommer, Ole Winther, Morten Nielsen, and Bent Petersen. Netsurfp-2.0: Improved prediction of protein structural features by integrated deep learning. *Proteins: Structure, Function, and Bioinformatics*, 87(6):520–527, 2019.

[16] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. Prottrans: Towards cracking the language of life's code through self-supervised learning. *bioRxiv*, 1 2021. doi: 10.1101/2020.07.12.199554.

[17] Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, and Weize Chen. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235, 2023.

[18] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[19] Michael Heinzinger, Konstantin Weissenow, Joaquin Gomez Sanchez, Adrian Henkel, Martin Steinegger, and Burkhard Rost. Prostt5: Bilingual language model for protein sequence and structure. *bioRxiv*, 2023. doi: 10.1101/2023.07.23.550085.

[20] Luciano A. Abriata, Giorgio E. Tamò, Bohdan Monastyrskyy, Andriy Kryshtafovych, and Matteo Dal Peraro. Assessment of hard target modeling in casp12 reveals an emerging role of alignment-based contact prediction methods. *Proteins: Structure, Function, and Bioinformatics*, 86:97–112, 2018.