Light Attention Predicts Protein Location from the Language of Life

Hannes Stärk, Christian Dallago,
Michael Heinzinger, Burkhard Rost

Technical University of Munich

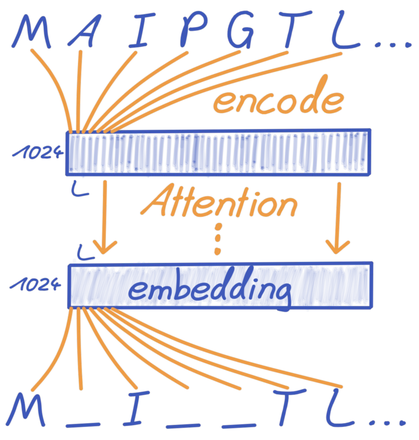
Subcellular localization predictions with 8 percentage points higher accuracy than before

Proteins are the machinery of life involved in all essential biological processes. A part of finding out how they function is knowing their location in a cell. To predict this subcellular localization, we use language models that were pre-trained on protein sequences to generate protein embeddings. These are processed by our very simple and low-size Light Attention (LA) architecture to beat the previous methods. Advantages:

- Much more accurate
- Faster
- Only sequence input required
- No database search
- Cheap: training and inference on consumer hardware
- Simple!

Protein Language Models

We pre-train three language models either with masked language modeling or next-token-prediction. Each amino acid in a sequence is seen as a token. We try the LSTM SeqVec, and two transformers: ProtBert and ProtT5.

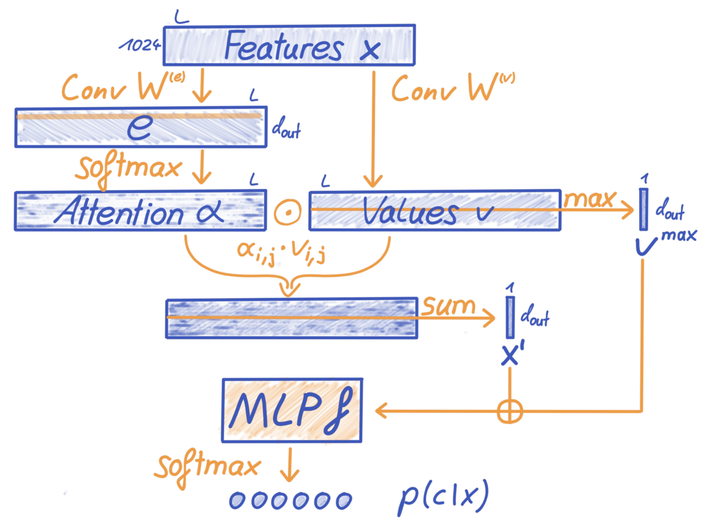


	SeqVec	ProtBert	ProtT5
parameters	93M	420M	3B
# sequences	33M	2.1B	2.1B
Sec per sequence	0.03	0.06	0.1
attention heads	-	16	32
float precision	32bit	32bit	16bit

Number of sequences used for pre-training and how many seconds it takes to embed a single sequence.

Light Attention Architecture

Takes protein language model embeddings as input and classifies their subcellular localization.



The only learnable weights are in the two convolutions and the final MLP. The “Light Attention” part is simply a weighted sum where the weights are given by a learned function of the inputs.

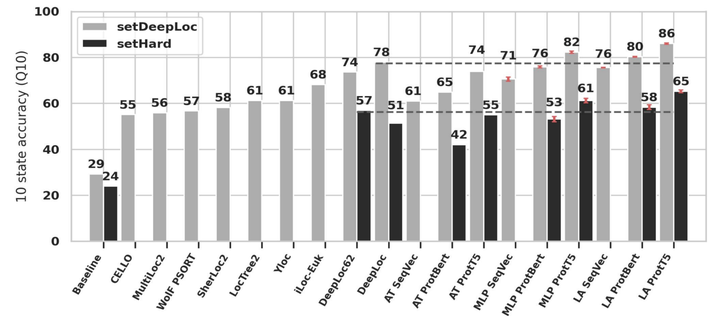
Localization prediction Datasets

We train on the standard localization dataset DeepLoc. It has 10 different subcellular localization classes. We evaluate on the DeepLoc [1] test set, which is redundancy reduced to the training data to **30% PIDE**. We also evaluate on a new, harder test set with at most **20% PIDE** to the train set.

Location	DeepLoc #	DeepLoc %	setHARD #	setHARD %
Nucleus	4043	28.9	99	20.2
Cytoplasm	2542	19.3	117	23.8
Extracellular	1973	14.0	92	18.8
Mitochondrion	1510	11.8	10	2.0
Cell Membrane	1340	9.5	98	20.0
ER	862	6.2	34	6.9
Plastid	757	5.4	11	2.6
Golgi apparatus	356	2.6	13	2.6
Lysosome/Vacuole	321	2.3	13	2.2
Peroxisome	154	1.1	3	0.6

The number of sequences in each class and the percentage the class makes up of the total set. Shown for the DeepLoc test set and our new harder test set.

Comparison with previous Methods



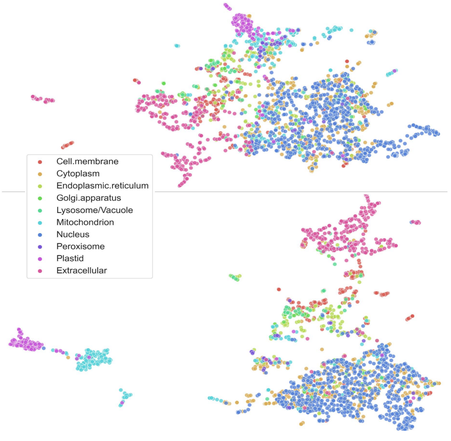
Bars give the ten-class accuracy (Q10) for popular location prediction methods on setDeepLoc (light-gray bars) and setHARD (dark-gray bars). Baseline is the most common class in each set. Horizontal gray dashed lines mark the previous SOTA on either set. Estimates for standard errors are marked in orange for the methods introduced here.

The LA architecture outperforms the previous SOTA by **8** percentage points on both test sets. Also, DeepLoc is worse on setHARD with profile input than with BLOSUM. Maybe profiles just leverage sequence redundancy?

Other methods on ProtT5 embeddings

Method	setDeepLoc	setHARD
LA	**86.01**± 0.34	**65.21**± 0.61
LA - Softmax	85.30± 0.32	64.72± 0.70
LA - Maxpool	84.79± 0.19	63.84± 0.67
Attention from v	85.41± 0.27	64.77± 0.93
DeepLoc LSTM	79.40± 0.88	59.36± 0.84
Conv + AdaPool	82.09± 0.92	60.79± 2.01
MeanPool + MLP	82.27± 0.51	61.27± 0.97
LA on OneHot	43.53± 1.48	32.57± 2.38
LA on profiles	43.78± 1.25	33.35± 1.82

Accuracy of additional baselines and ablations: drop softmax, drop maxpool, calculate attention coefficients from v, DeepLoc LSTM, stacked convolutions with adaptive pooling.



UMAP of per-protein embeddings in the DeepLoc test set colored according to subcellular location. Top: ProtT5 embeddings mean pooled over protein length. Bottom: ProtT5 embeddings weighted according to the attention distribution produced by LA (so this is not x’).

[1] Almagro Armenteros et al. (2017). “DeepLoc: prediction of protein subcellular localization using deep learning.” In: Bioinformatics, 33(21):3387–3395.