TMbed – Transmembrane proteins predicted through Language Model embeddings

Michael Bernhofer [1, 2, *], Burkhard Rost [1, 3]

1 Department of Informatics, Bioinformatics and Computational Biology - i12, Technical University of Munich (TUM), Boltzmannstr. 3, 85748 Garching, Germany

2 TUM Graduate School, Center of Doctoral Studies in Informatics and its Applications (CeDoSIA), Boltzmannstr. 11, 85748 Garching, Germany

3 Institute for Advanced Study (TUM-IAS), Lichtenbergstr. 2a, 85748 Garching, Germany & TUM School of Life Sciences Weihenstephan (TUM-WZW), Alte Akademie 8, Freising, Germany

* Corresponding author: bernhoferm@rostlab.org, https://www.rostlab.org/

Abstract

Background: Despite the immense importance of transmembrane proteins (TMP) for molecular biology and medicine, experimental 3D structures for TMPs remain about 4-5 times underrepresented compared to non-TMPs. Today's top methods can accurately predict structures for many, but the annotations of the transmembrane regions remains a limiting step for proteome-wide predictions.

Results: Here, we present a novel method, dubbed TMbed. Inputting embeddings from protein Language Models (in particular ProtT5), TMbed completes predictions of alpha helical and beta barrel TMPs for entire proteomes within hours on a single consumer-grade desktop machine at performance levels similar or better than methods, which are using evolutionary information (extracted from family alignments). On the per-protein level, TMbed correctly identified 61 of the 65 beta-barrel TMPs (94±7%) and 579 of the 593 alpha-helical TMPs (98±1%) in a non-redundant data set, at false positive rates well below 1% (erred on 31 of 5859 non-membrane proteins). On the per-segment level, TMbed correctly placed, on average, 9 of 10 transmembrane segments within five residues of the experimental observation. Although limited by GPU memory, our method can handle sequence of up to 4200 residues on standard graphics cards used in common desktop PCs (e.g., NVIDIA GeForce RTX 3060).

Conclusions: TMbed accurately predicts alpha helical and beta barrel TMPs. Utilizing protein Language Models and GPU acceleration it can predict the human in less than an hour.

Availability: Our code, method, and data sets are freely available in the GitHub repository, https://github.com/BernhoferM/TMbed.

Keywords: protein language models, protein structure prediction, transmembrane protein prediction.

Abbreviations used: CI, confidence interval; **CNN**, convolutional neural network; **MSA**, multiple sequence alignment; **OPM**, orientations of proteins in membranes database; **PDB**, protein data bank; **PDBTM**, protein data bank of transmembrane proteins; **pLM**, protein language model; **SIFTS**, structure integration with function, taxonomy and sequence; **SOM**, supplementary online material; **SOTA**, state-of-the-art; **SP**, signal peptide; **TMB**, transmembrane beta strand; **TMH**, transmembrane helix; **TMP**, transmembrane protein;

Background

Structural knowledge of TMPs 4-5 fold underrepresented. Transmembrane proteins (TMP) account for 20-30% of all proteins within any organism (1, 2); most TMPs cross the membrane with transmembrane helices (TMH). TMPs crossing with transmembrane beta strands (TMB), forming beta barrels, have been estimated to account for 1-2% of all proteins in Gram-negative bacteria; this variety is also present in mitochondria and chloroplasts (3). Membrane proteins facilitate many essential processes, including regulation, signaling, and transportation, rendering them targets for most known drugs (4, 5). Despite this immense relevance for molecular biology and medicine, only about 5% of all three-dimensional (3D) structures in the PDB (6, 7) constitute TMPs.

Accurate 3D predictions available for proteomes need classification. The prediction of protein structure from sequence leaped in quality through AlphaFold2 (8), Nature's method of the year 2021 (9). Although AlphaFold2 appears to provide accurate predictions for only very few novel "folds", it importantly increases the width of structural coverage (10). AlphaFold2 seems to work well on TMPs (11), but for proteome-wide high-throughput studies, we still need to filter out membrane proteins from the structure predictions. Most state-of-the-art (SOTA) TMP prediction methods rely on evolutionary information in the form of multiple sequence alignments (MSA) to achieve their top performance. In our tests we included 12 such methods, namely BetAware-Deep (12), BOCTOPUS2 (13), HMM-TM (14-16), OCTOPUS (17), Philius (18), PolyPhobius (19), PRED-TMBB2 (15, 16, 20), PROFtmb (3), SCAMPI (21), SPOCTOPUS (22), TMSEG (23), and TOPCONS2 (24).

pLMs capture crucial information without MSAs. Mimicking recent advances of Language Models (LM) in natural language processing (NLP), protein Language Models (pLMs) learn to reconstruct masked parts of protein sequences based on the unmasked local and global information (25-32). Such pLMs, trained on billions of protein sequences, implicitly extract important information about protein structure and function, essentially capturing aspects of the "language of life" (27). These aspects can be extracted from the last layers of the deep learning networks into vectors, referred to as embeddings, and used as exclusive input to subsequent methods trained in supervised fashion to successfully predict aspects of protein structure and function (25-29, 31, 33-38). Often pLM-based methods outperform SOTA methods, which are using evolutionary information on top, and they usually require substantially fewer compute resources. Just before submitting this

work, we became aware of another pLM-based TM-prediction method, namely DeepTMHMM (39) using ESM-1b (31) embeddings, and included it in our comparisons.

Here, we combined embeddings generated by the ProtT5 (29) pLM with a simple convolutional neural network (CNN) to create a fast and highly accurate prediction method for alpha helical and beta barrel transmembrane proteins and their overall inside/outside topology. Our new method, TMbed, predicted the presence and location of any TMBs, TMHs, and signal peptides for all proteins of the human proteome within 46 minutes on our server machine (SOM: Table S1) at the same or better level of performance as other methods, which require substantially more time.

Materials & Methods

Data set: membrane proteins (TMPs). We collected all primary structure files for alpha helical and beta barrel transmembrane proteins (TMP) from OPM (40) and mapped their PDB (6, 7) chain IDs to UniProtKB (41) protein sequences using SIFTS (42, 43). We discarded all chimeric PDB chains and model structures, as well as PDB chains in which we could not map the start or end positions for any of the transmembrane segments. This resulted in 2,053 and 206 sequence-unique PDB chains for alpha helical and beta barrel TMPs, respectively.

We used the ATOM coordinates inside the OPM files to assign the inside/outside orientation of sequence segments not within the membrane. We manually inspected inconsistent annotations (e.g., if both ends of a transmembrane segment had the same inside/outside orientation) and cross-referenced them with PDBTM (44-46), PDB, and UniProtKB. We then either corrected such inconsistent annotations or discarded the whole sequence. As OPM does not include signal peptide annotations, we compared our TMP data sets to the set used by SignalP 6.0 (47) and all sequences in UniProtKB/Swiss-Prot with experimentally annotated signal peptides using CD-HIT (48, 49). For any matches with at least 95% global sequence identity, we transferred the signal peptide annotation onto our TMPs. We removed all sequences with less than 50 residues to avoid noise from incorrect sequencing fragments, and all sequences with more than 15,000 residues to save energy (lower computational costs).

Finally, we removed redundant sequences from the two TMP data sets by clustering them with MMseqs2 (50) to at most 20% local pairwise sequence identity (PIDE) with 40% minimum alignment coverage, i.e., no pair had more than 20% PIDE for any local alignment covering at least 40% of the shorter sequence. The final non-redundant TMP data sets contained 593 alpha helical TMPs and 65 beta barrel TMPs, respectively.

Data set: globular non-membrane proteins. We used the SignalP 6.0 (SP6) dataset for our globular proteins. As the SP6 dataset contained only the first 70 residues of each protein, we took the full sequences from UniProtKB/Swiss-Prot and transferred the signal peptide annotations. To remove any potential membrane proteins from this non-TMP data set, we compared it with CD-HIT (48, 49) against three other data sets: (1) our TMP data sets before redundancy reduction, (2) all protein sequences from UniProtKB/Swiss-Prot with any annotations of transmembrane segments, and (3) all proteins from UniProtKB/Swiss-Prot with any subcellular location annotations for membrane. We removed all proteins from our non-TMP data set with more than 60% global PIDE to any protein in sets 1-3. Again,

we dropped all sequences with less than 50 or more than 15,000 residues and applied the same redundancy reduction as before (20% PIDE at 40% alignment coverage). The final non-redundant data set contained 5,859 globular, water-soluble non-TMP proteins; 698 of these have a signal peptide.

Embeddings. We generated embeddings with protein Language Models (pLMs) for our data sets using a transformer-based pLM ProtT5-XL-U50 (short: ProtT5) (29). We discarded the decoder part of ProtT5, keeping only the encoder for increased efficiency (note: encoder embeddings are more informative (29)). The encoder model converts a protein sequence into an embedding matrix that represents each residue in the protein, i.e., each position in the sequence, by a 1024-dimensional vector containing global and local contextualized information. We converted the ProtT5 encoder from 32-bit to 16-bit floating-point format to reduce the memory footprint on the GPU. We took the pre-trained ProtT5 model as is without any further task-specific fine-tuning.

We chose ProtT5 over other embedding models, such as ESM-1b (31), based on our experience with the model and comparisons during previous projects (29, 33). Furthermore, ProtT5 does not require splitting long sequences, which might remove valuable global context information, while ESM-1b can only handle sequences of up to 1022 residues.

Model architecture. Our TMbed model architecture contained three modules (SOM: Fig. S1): a convolutional neural network (CNN) to generate per-residue predictions, a Gaussian smoothing filter, and a Viterbi decoder to find the best class label for each residue. We implemented the model in PyTorch (51).

Module 1: CNN: The first component of TMbed is a CNN with four layers (SOM: Fig. S1). The first layer is a pointwise convolution, i.e., a convolution with kernel size of 1, which reduces the ProtT5 embeddings for each residue (position in the sequence) from 1024 to 64 dimensions. Next, the model applies layer normalization (52) along the sequence and feature dimensions, followed by a ReLU (Rectified Linear Unit) activation function to introduce non-linearity. The second and third layers consist of two parallel depthwise convolutions; both process the output of the first layer. As depthwise convolutions process each input dimension (feature) independently while considering consecutive residues, those two layers effectively generate sliding weighted sums for each dimension. The kernel sizes of the second and third layer are 9 and 21, respectively, corresponding to the average length of transmembrane beta strands and helices. As before, the model normalizes the

output of both layers and applies the ReLU function. It then concatenates the output of all three layers, constructing a 192-dimensional feature vector for each residue (position in the sequence). The fourth layer is a pointwise convolution combining the outputs from the previous three layers and generates scores for each of the five classes: transmembrane beta strand (B), transmembrane helix (H), signal peptide (S), non-membrane inside (i), and non-membrane outside (o).

Module 2: Gaussian filter: This module smooths the output from the CNN for adjacent residues (sequence positions) to reduce noisy predictions. The filter allows flattening isolated single-residue peaks. For instance, peaks extending of only one to three residues for the classes B and H are often non-informative; similarly short peaks for class S are unlikely correct. The filter uses a Gaussian distribution with standard deviation of 1 and a kernel size of 7, i.e., its seven weights correspond to three standard deviation intervals to the left and right, as well as the central peak. A softmax function then converts the filtered class scores to a class probability distribution.

Module 3: Viterbi decoder: The Viterbi algorithm decodes the class probabilities and assigns a class label to each residue (position in the sequence; SOM: Note S3, Fig. S2). The algorithm uses no trainable parameter; it scores transitions according to the predicted class probabilities. Its purpose is to enforce a simple grammar such that (1) signal peptides can only start at the N-terminus (first residue in protein), (2) signal peptides and transmembrane segments must be at least five residues long (a reasonable trade-off between filtering out false positives and still capturing weak signals), and (3) the prediction for the inside/outside orientation has to change after each transmembrane segment (to simulate crossing through the membrane). Unlike the Gaussian filter, we did not apply the Viterbi decoder during training. This simplified backpropagation and sped up training.

Training details. We performed a stratified five-fold nested cross-validation for model development (SOM: Fig. S3). First, we separated our protein sequences into four groups: beta barrel TMPs, alpha helical TMPs with only a single helix, those with multiple helices, and non-membrane proteins. We further subdivided each group into proteins with and without signal peptides. Next, we randomly and evenly distributed all eight groups into five data sets. As all of our data sets were redundancy reduced, no two splits contained similar protein sequences for any of the classes. However, similarities between proteins of two different classes were allowed, not the least to provide more conservative performance estimates.

During development, we used four of the five splits to create the model and the fifth for testing (SOM: Fig. S3). Of the first four splits, we used three to train the model and the fourth for validation (optimize hyperparameters). We repeated this 3-1 split three more times, each time using a different split for the validation set, and calculated the average performance for every hyperparameter configuration. Next, we trained a model with the best configuration on all four development splits and estimated its final performance on the independent test split. We performed this whole process a total of five times, each time using a different of the five splits as test data and the remaining four for the development data. This resulted in five final models; each trained, optimized, and tested on independent data sets.

We applied weight decay to all trained weights of the model and added a dropout layer right before the fourth convolutional layer, i.e., the output layer of the CNN. For every training sample (protein sequence), the dropout layer randomly sets 50% of the features to zero across the entire sequence, preventing the model from relying on only a specific subset of features for the prediction.

We trained all models for 15 epochs using the AdamW (53) optimizer and cross-entropy loss. We set the beta parameters to 0.9 and 0.999, used a batch size of 16 sequences, and applied exponential learning rate decay by multiplying the learning rate with a factor of 0.8 every epoch. The initial learning rate and weight decay values were part of the hyperparameters optimized during cross-validation (SOM: Table S2).

The final TMbed model constitutes an ensemble over the five models obtained from the five outer cross-validation iterations (SOM: Fig. S3), i.e., one for each training/test set combination. During runtime, each model generates its own class probabilities (CNN, plus Gaussian filter), which are then averaged and processed by the Viterbi decoder to generate the class labels.

Evaluation and other methods. We evaluated the test performance of TMbed on a per-protein level and on a per-segment level (SOM: Note S1). For protein level statistics, we calculated recall and false positive rate (FPR). We computed those statistics for three protein classes: alpha helical TMPs, beta barrel TMPs, and globular proteins.

We distinguished correct and incorrect segment predictions using two constraints: 1) the observed and predicted segment must overlap such that the intersection of the two is at least half of their union, and 2) neither the start nor the end positions may deviate by more than five residues between the observed and predicted segment (SOM: Fig. S4). All segments predicted meeting both these criteria were considered as "correctly predicted

segments", all others as "incorrectly predicted segments". This allowed for a reasonable margin of error regarding the position of a predicted segment, while punishing any gaps introduced into a segment. For per-segment statistics, we calculated recall and precision. We also computed the percentage of proteins with the correct number of predicted segments (Q_{num}), the percentage of proteins for which all segments are correctly predicted (Q_{ok}), and the percentage of correctly predicted segments that also have the correct orientation within the membrane (Q_{top}). We considered only proteins that actually contain the corresponding type of segment when calculating per-segment statistics, e.g., only beta barrel TMPs for transmembrane beta strand segments.

We compared TMbed to other prediction methods for alpha helical and beta barrel TMPs (details in SOM Note S2): BetAware-Deep (12), BOCTOPUS2 (13), DeepTMHMM (39), HMM-TM (14-16), OCTOPUS (17), Philius (18), PolyPhobius (19), PRED-TMBB2 (15, 16, 20), PROFtmb (3), SCAMPI (21), SPOCTOPUS (22), TMSEG (23), and TOPCONS2 (24). We chose those methods based on their good prediction accuracy and public popularity. For methods predicting only either alpha helical or beta barrel TMPs, we considered the corresponding other type of TMPs as globular proteins for the per-protein statistics. In addition, we generated signal peptide predictions with SignalP 6.0 (47). The performance of older TMH prediction methods could be triangulated based on previous comprehensive estimate of such methods (23, 54).

Unless stated otherwise, all reported performance values constitute the average performance over the five independent test sets during cross-validation (c.f. *Training details*) and their error margins reflect the 95% confidence interval (CI), i.e., 1.96 times the sample standard error over those five splits (SOM: Tables S5 & S6). We considered two values A and B statistically significantly different if they differ by more than their composite 95% confidence interval:

$$|A - B| > CI_c = \sqrt{CI_A^2 + CI_B^2} \quad \text{(Eq. 1)}$$

Results & Discussion

We have developed a new machine learning model, dubbed TMbed, using only embeddings from the ProtT5 (29) pLM as input to predict for each residue in a protein sequence to which of the following four "classes" it belongs: transmembrane beta strand (TMB), transmembrane helix (TMH), signal peptide (SP), or non-transmembrane segment. It also predicts the inside/outside orientation of TMBs and TMHs within the membrane, indicating which parts of a protein are inside or outside a cell or compartment. Although the prediction of signal peptides was primarily integrated to improve TMH predictions by preventing the confusion of TMHs with SPs and *vice versa*, we also evaluated and compared the performance for SP prediction of TMbed to that of other methods.

Reaching SOTA in protein sorting. TMbed detected TMPs with TMHs and TMBs at levels similar or numerically above the best state-of-the-art (SOTA) methods using evolutionary information from multiple sequence alignments (MSA; Table 1: Recall). Compared to MSA-based methods, TMbed achieved this parity or improvement at a significantly lower false positive rate (FPR), tied only with DeepTMHMM (39), another embedding-based method (Table 1: FPR). Given those numbers, we expect TMbed to misclassify only about 209 proteins for a proteome with 20,000 proteins (SOM: Table S10), e.g., the human proteome, while the other methods would make hundreds more mistakes (DeepTMHMM: 322, TOPCONS2: 685, BOCTOPUS2: 863). Such low FPRs suggest our method as an automated high-throughput filter for TMP detection, e.g., for the creation and annotation of databases, or the decision which AlphaFold2 (8, 55) predictions to parse through advanced software annotating transmembrane regions in 3D structures or predictions (40, 44, 56). In the binary prediction of whether or not a protein has a signal peptide, TMbed achieved similar levels as the specialist SignalP 6.0 (47) and as DeepTMHMM (39), reaching 99% recall at 0.1% FPR (SOM: Table S3).

Many of the beta barrel TMPs that prediction methods missed had only two or four transmembrane beta strands (TMB). Such proteins cannot form a pore on their own, instead they have to form complexes with other proteins to function as TMPs, either by binding to other proteins or by forming multimers with additional copies of the same proteins by, e.g., trimerization. In fact, all four beta barrel TMPs missed by TMbed fell into this category. For another protein with four TMBs, TMbed predicted one single TMB, which would probably be ignored by expert users without additional knowledge. Thus, as all other methods, TMbed performed, on average, worse for beta barrel TMPs that cannot form pores alone. This appeared unsurprising, as the input to all methods were single proteins.

For TMPs with TMHs, we also observed lower performance in the distinction between TMP/other for TMPs with a single TMH (recall: 93±3%) compared to those with multiple TMHs (recall: 99±1%). However, TMPs with single helices can function alone.

The embedding-based methods TMbed (introduced here using ProtT5 (29)) and DeepTMHMM (39) (based on ESM-1b (31)) performed at least on par with the SOTA using evolutionary information from MSA (Table 1). While this was already impressive, the real advantage was in the speed. For instance, our method, TMbed, predicted all 6,517 proteins in our data set in about 13 minutes (i.e., about eight sequences per second) on our server machine (SOM: Table S1); this runtime included generating the ProtT5 embeddings. The other embedding-based method, DeepTMHMM, needed about twice as long (23 minutes). Meanwhile, methods that search databases and generate MSAs usually take several seconds or minutes for a single protein sequence (57), or require significant amounts of computing resources (e.g., often more than 100 GB of memory) to achieve comparable runtimes (50).

Excellent transmembrane segment prediction performance. TMbed reached the highest performance for transmembrane segments amongst all methods evaluated (Tables 2 & 3). With recall and precision values of 89±1% for TMHs, it significantly outperformed the second best and only other embedding-based method, DeepTMHMM, (79±2%, Table 2). TMbed essentially predicted almost 62% of all transmembrane helical (TMH) TMPs completely correctly (Q_{ok}, i.e., all TMHs within ±5 residues of true annotation). DeepTMHMM reached second place with Q_{ok} of 46±4%. This difference between TMbed and DeepTMHMM was over twice that between DeepTMHMM and the two methods performing third-best by this measure, TOPCONS2 (24) and PolyPhobius (19), which are based on evolutionary information.

The results were largely similar for beta-barrel TMPs (TMBs) with TMbed achieving the top performance by all measures: reaching 95% recall and an almost perfect 99% precision. The most pronounced difference was a 20 percentage points lead in Q_{ok} with 79%, compared to BOCTOPUS2 (13) with 59% in second place. Overall, TMbed predicted the correct number of transmembrane segments in 86% of TMPs and correctly oriented 99% of TMBs and 96% of TMHs. For signal peptides, TMbed performed on par with SignalP 6.0, reaching 94% recall and 95% precision (SOM: Table S3). For this task, both methods appeared to be slightly outperformed by DeepTMHMM. However, none of those differences exceeded the 95% confidence interval, i.e., the numerically consistent differences were not statistically significant. On top, the signal peptide expert method

SignalP 6.0 is the only of the three that distinguishes between different types of signal peptides.

As for the overall per-protein distinction between TMP and non-TMP, the per-segment recall and precision also slightly correlated with the number of transmembrane segments, i.e., the more TMHs or TMBs in a protein the higher the performance (SOM: Table S4). Again, as for the TMP/non-TMP distinction, beta barrel TMPs with only two or four TMBs differed most to those with eight or more.

Gaussian filter and Viterbi decoder improve segment performance. TMbed introduced a Gaussian filter smoothing over some local peaks in the prediction and a Viterbi decoder implicitly enforcing some "grammar-like" rules (Materials & Methods). We investigated the effect of these concepts by comparing the final TMbed architecture to three simpler alternatives: one variant used only the CNN, the other two variants combined the simple CNN with either the Gaussian filter or the Viterbi decoder, not both as TMbed. For the variants without the Gaussian filter, we retrained the CNN using the same hyperparameters but without the filter. Individually, both modules (filter and decoder) significantly improved precision and Q_{ok} for both TMH and TMB, while recall remained largely unaffected (SOM: Table S9). Clearly, either step already improved over just the CNN. However, which of the two was most important depended on the type of TMP: for TMH proteins Viterbi decoder mattered more, for TMB proteins the Gaussian filter. Both steps together performed best throughout without adding any significant overhead to the overall computational costs compared to the other components.

Self-predictions reveal potential membrane proteins. We checked for potential overfitting of our model by predicting the complete data set with the final TMbed ensemble. This meant that four of the five models had seen each of those proteins during training. While the number of misclassified proteins went down, we found that there were still some false predictions, indicating that our models did not simply learn the training data by heart (SOM: Tables S7 & S8). In fact, upon closer inspection of the 11 false positive predictions (8 alpha helical and 3 beta barrel TMPs), those appear to be transmembrane proteins incorrectly classified as globular proteins in our data set due to missing annotations in UniProtKB/Swiss-Prot, rather than incorrect predictions. Two of them, P09489 and P40601, have automatic annotations for an autotransporter domain, which facilitates transport through the membrane. Further, we processed the predicted AlphaFold2 (8, 55) structures of all 11 proteins using the PPM (40) algorithm, which tries

to embed 3D structures into a membrane bilayer. For eight of those, the predicted transmembrane segments correlated well with the predicted 3D structures and membrane boundaries (Fig. 1; SOM: Fig. S5). For the other three, the 3D structures and membrane boundaries still indicate transmembrane domains within those proteins, but the predicted transmembrane segments only cover parts of those domains (SOM: Fig. S5, last row). Together, these predictions provided convincing evidence for considering all eleven proteins as TMPs.

Predicting the human proteome in less than an hour. Given that our new method already outperformed the SOTA using evolutionary information from MSAs, the even more important advantage was speed. To estimate prediction throughput, we applied TMbed to all human proteins in 20,375 UniProtKB/Swiss-Prot (version: April 2022; excluding TITIN_HUMAN due to its extreme length of 34,350 residues). Overall, it took our server machine (SOM: Table S1) only 46 minutes to generate all embeddings and predictions (estimate for consumer-grade PC in the next section). TMbed identified 14 beta barrel TMPs and 4,953 alpha helical TMPs, matching previous estimates for alpha helical TMPs (1, 23). Two of the 14 TMBs appear to be false positives as TMbed predicted only a single TMB in each protein. The other 12 proteins are either part of the Gasdermin family (A to E), or associated with the mitochondrion, including three proteins for a voltage-dependent anion-selective channel and the TOM40 import receptor.

Further, we generated predictions for all proteins from UniProtKB/Swiss-Prot (version: May 2022), excluding sequences above 10,000 residues (20 proteins). Processing those 566,976 proteins took about 8.5 hours on our server machine. TMbed predicted 1,702 beta barrel TMPs and 77,296 alpha helical TMPs (predictions available via our GitHub repository).

Hardware requirements. Our model needs about 2.5 GB of memory on the GPU when in 16-bit format. The additional memory needed during inference grows with the square of sequence length due to the attention mechanism of the transformer architecture. On our consumer-grade desktop PC (SOM: Table S1), this translated to a maximum sequence length of about 4,200 residues without maxing out the 12 GB of GPU memory. This barred 76 (0.4%) of the 20,376 human proteins from analysis on a personal consumer-hardware solution (NVIDIA GeForce RTX 3060). The prediction (including embedding generation) for 99.6% of the human proteome (20,376 proteins) took about 57 minutes on our desktop PC. While it is possible to run the model on a CPU, instead of on a GPU, we do not

recommend this due to over 10-fold larger runtimes. More importantly, the current lack of support of 16-bit floating-point format on CPUs would imply doubling the memory footprint of the model and computations.

Conclusions

TMbed predicts alpha helical (TMH) and beta barrel (TMB) transmembrane proteins (TMPs) with high accuracy (Table 1), performing at least on par or even better than state-of-the-art (SOTA) methods, which depend on evolutionary information from multiple sequence alignments (MSA; Tables 1-3). In contrast, TMbed exclusively inputs sequence embeddings from the protein language model (pLM) ProtT5. Our novel method shines, in particular, through its low false positive rate (FPR; Table 1), incorrectly predicting less than 1% of globular proteins to be TMPs. TMbed also numerically outperformed all other tested methods in terms of correctly predicting transmembrane segments (on average, 9 out of 10 segments were correct; Tables 2 & 3). Despite its top performance, the even more significant advantage of TMbed is speed: the high throughput rate of the ProtT5 (29) encoder enables predictions for entire proteomes within an hour, given a suitable GPU (SOM: Table S1). On top, the method runs on consumer-grade GPUs as found in more recent gaming and desktop PCs. Thus, TMbed can be used as a proteome-scale filtering step to scan for transmembrane proteins. Validating the predicted segments with AlphaFold2 (8, 55) structures and the PPM (40) method could be combined into a fast pipeline to discover new membrane proteins, as we have demonstrated with a few proteins. Finally, we provide predictions for 566,976 proteins from UniProtKB/Swiss-Prot (version: May 2022) via our GitHub repository.

Declarations

Ethics approval and consent to participate: Not applicable.

Consent for publication: Not applicable.

Competing interests: The authors declare that they have no competing interests.

Availability of data and materials: Our code, method, and data sets are freely available in the GitHub repository, https://github.com/BernhoferM/TMbed.

Authors' contributions: MB collected the data sets, developed and evaluated the TMbed model, and took the lead in writing the manuscript. BR supervised and guided the work, and co-wrote the manuscript. All authors read and approved the final manuscript.

Acknowledgements: Thanks to Tim Karl and Inga Weise for their help with technical and administrative issues; to Tobias Olenyi, Michael Heinzinger, and Christian Dallago for thoughtful discussions, help with ProtT5, and help with the manuscript; to Konstantinos Tsirigos and Ioannis Tamposis for their support with setting up HMM-TM and PRED-TMBB2; to Pier Luigi Martelli for providing us with BetAware-Deep predictions. Last but not least, thanks to all who deposit their experimental data in public databases, and to those who maintain them.

Funding: The server machine to run the ProtT5 model was funded by Software Campus Funding (BMBF 01IS17049).

References

1. Fagerberg L, Jonasson K, von Heijne G, Uhlen M, Berglund L. Prediction of the human membrane proteome. Proteomics. 2010;10(6):1141-9.
2. Liu J, Rost B. Comparing function and structure between entire proteomes. Protein Sci. 2001;10(10):1970-9.
3. Bigelow HR, Petrey DS, Liu J, Przybylski D, Rost B. Predicting transmembrane beta-barrels in proteomes. Nucleic Acids Res. 2004;32(8):2566-77.
4. Overington JP, Al-Lazikani B, Hopkins AL. How many drug targets are there? Nat Rev Drug Discov. 2006;5(12):993-6.
5. von Heijne G. The membrane protein universe: what's out there and why bother? J Intern Med. 2007;261(6):543-57.
6. ww PDBc. Protein Data Bank: the single global archive for 3D macromolecular structure data. Nucleic Acids Res. 2019;47(D1):D520-D8.
7. Berman H, Henrick K, Nakamura H. Announcing the worldwide Protein Data Bank. Nat Struct Biol. 2003;10(12):980.
8. Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with AlphaFold. Nature. 2021;596(7873):583-9.
9. Marx V. Method of the Year: protein structure prediction. Nat Methods. 2022;19(1):5-10.
10. Bordin N, Sillitoe I, Nallapareddy V, Rauer C, Lam SD, Waman VP, et al. AlphaFold2 reveals commonalities and novelties in protein structure space for 21 model organisms. bioRxiv. 2022:2022.06.02.494367.
11. Hegedus T, Geisler M, Lukacs GL, Farkas B. Ins and outs of AlphaFold2 transmembrane protein structure predictions. Cell Mol Life Sci. 2022;79(1):73.
12. Madeo G, Savojardo C, Martelli PL, Casadio R. BetAware-Deep: An Accurate Web Server for Discrimination and Topology Prediction of Prokaryotic Transmembrane beta-barrel Proteins. J Mol Biol. 2021;433(11):166729.
13. Hayat S, Peters C, Shu N, Tsirigos KD, Elofsson A. Inclusion of dyad-repeat pattern improves topology prediction of transmembrane beta-barrel proteins. Bioinformatics. 2016;32(10):1571-3.
14. Bagos PG, Liakopoulos TD, Hamodrakas SJ. Algorithms for incorporating prior topological information in HMMs: application to transmembrane proteins. BMC Bioinformatics. 2006;7:189.
15. Tamposis IA, Sarantopoulou D, Theodoropoulou MC, Stasi EA, Kontou PI, Tsirigos KD, et al. Hidden neural networks for transmembrane protein topology prediction. Comput Struct Biotechnol J. 2021;19:6090-7.
16. Tamposis IA, Theodoropoulou MC, Tsirigos KD, Bagos PG. Extending hidden Markov models to allow conditioning on previous observations. J Bioinform Comput Biol. 2018;16(5):1850019.
17. Viklund H, Elofsson A. OCTOPUS: improving topology prediction by two-track ANN-based preference scores and an extended topological grammar. Bioinformatics. 2008;24(15):1662-8.
18. Reynolds SM, Kall L, Riffle ME, Bilmes JA, Noble WS. Transmembrane topology and signal peptide prediction using dynamic bayesian networks. PLoS Comput Biol. 2008;4(11):e1000213.
19. Kall L, Krogh A, Sonnhammer EL. An HMM posterior decoder for sequence feature prediction that includes homology information. Bioinformatics. 2005;21 Suppl 1:i251-7.
20. Tsirigos KD, Elofsson A, Bagos PG. PRED-TMBB2: improved topology prediction and detection of beta-barrel outer membrane proteins. Bioinformatics. 2016;32(17):i665-i71.
21. Bernsel A, Viklund H, Falk J, Lindahl E, von Heijne G, Elofsson A. Prediction of membrane-protein topology from first principles. Proc Natl Acad Sci U S A. 2008;105(20):7177-81.

22. Viklund H, Bernsel A, Skwark M, Elofsson A. SPOCTOPUS: a combined predictor of signal peptides and membrane protein topology. Bioinformatics. 2008;24(24):2928-9.
23. Bernhofer M, Kloppmann E, Reeb J, Rost B. TMSEG: Novel prediction of transmembrane helices. Proteins. 2016;84(11):1706-16.
24. Tsirigos KD, Peters C, Shu N, Kall L, Elofsson A. The TOPCONS web server for consensus prediction of membrane protein topology and signal peptides. Nucleic Acids Res. 2015;43(W1):W401-7.
25. Asgari E, Mofrad MR. Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. PLoS One. 2015;10(11):e0141287.
26. Alley EC, Khimulya G, Biswas S, AlQuraishi M, Church GM. Unified rational protein engineering with sequence-based deep representation learning. Nat Methods. 2019;16(12):1315-22.
27. Heinzinger M, Elnaggar A, Wang Y, Dallago C, Nechaev D, Matthes F, et al. Modeling aspects of the language of life through transfer-learning protein sequences. BMC Bioinformatics. 2019;20(1):723.
28. Bepler T, Berger B. Learning the protein language: Evolution, structure, and function. Cell Syst. 2021;12(6):654-69 e3.
29. Elnaggar A, Heinzinger M, Dallago C, Rehawi G, Wang Y, Jones L, et al. ProtTrans: Towards Cracking the Language of Lifes Code Through Self-Supervised Deep Learning and High Performance Computing. IEEE Trans Pattern Anal Mach Intell. 2021;PP.
30. Ofer D, Brandes N, Linial M. The language of proteins: NLP, machine learning & protein sequences. Comput Struct Biotechnol J. 2021;19:1750-8.
31. Rives A, Meier J, Sercu T, Goyal S, Lin Z, Liu J, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proc Natl Acad Sci U S A. 2021;118(15).
32. Wu Z, Johnston KE, Arnold FH, Yang KK. Protein sequence design with deep generative models. Curr Opin Chem Biol. 2021;65:18-27.
33. Marquet C, Heinzinger M, Olenyi T, Dallago C, Erckert K, Bernhofer M, et al. Embeddings from protein language models predict conservation and variant effects. Hum Genet. 2021.
34. Littmann M, Heinzinger M, Dallago C, Weissenow K, Rost B. Protein embeddings and deep learning predict binding residues for various ligand classes. Sci Rep. 2021;11(1):23916.
35. Littmann M, Heinzinger M, Dallago C, Olenyi T, Rost B. Embeddings from deep learning transfer GO annotations beyond homology. Sci Rep. 2021;11(1):1160.
36. Sledzieski S, Singh R, Cowen L, Berger B. D-SCRIPT translates genome to phenome with sequence-based, structure-aware, genome-scale predictions of protein-protein interactions. Cell Syst. 2021;12(10):969-82 e6.
37. Heinzinger M, Littmann M, Sillitoe I, Bordin N, Orengo C, Rost B. Contrastive learning on protein embeddings enlightens midnight zone. bioRxiv. 2022:2021.11.14.468528.
38. Weißenow K, Heinzinger M, Rost B. Protein language model embeddings for fast, accurate, alignment-free protein structure prediction. bioRxiv. 2021:2021.07.31.454572.
39. Hallgren J, Tsirigos KD, Pedersen MD, Almagro Armenteros JJ, Marcatili P, Nielsen H, et al. DeepTMHMM predicts alpha and beta transmembrane proteins using deep neural networks. bioRxiv. 2022:2022.04.08.487609.
40. Lomize MA, Pogozheva ID, Joo H, Mosberg HI, Lomize AL. OPM database and PPM web server: resources for positioning of proteins in membranes. Nucleic Acids Res. 2012;40(Database issue):D370-6.
41. UniProt C. UniProt: the universal protein knowledgebase in 2021. Nucleic Acids Res. 2021;49(D1):D480-D9.
42. Dana JM, Gutmanas A, Tyagi N, Qi G, O'Donovan C, Martin M, et al. SIFTS: updated Structure Integration with Function, Taxonomy and Sequences resource allows 40-fold increase in coverage of structure-based annotations for proteins. Nucleic Acids Res. 2019;47(D1):D482-D9.

43. Velankar S, Dana JM, Jacobsen J, van Ginkel G, Gane PJ, Luo J, et al. SIFTS: Structure Integration with Function, Taxonomy and Sequences resource. Nucleic Acids Res. 2013;41(Database issue):D483-9.
44. Kozma D, Simon I, Tusnady GE. PDBTM: Protein Data Bank of transmembrane proteins after 8 years. Nucleic Acids Res. 2013;41(Database issue):D524-9.
45. Tusnady GE, Dosztanyi Z, Simon I. Transmembrane proteins in the Protein Data Bank: identification and classification. Bioinformatics. 2004;20(17):2964-72.
46. Tusnady GE, Dosztanyi Z, Simon I. PDB_TM: selection and membrane localization of transmembrane proteins in the protein data bank. Nucleic Acids Res. 2005;33(Database issue):D275-8.
47. Teufel F, Almagro Armenteros JJ, Johansen AR, Gislason MH, Pihl SI, Tsirigos KD, et al. SignalP 6.0 predicts all five types of signal peptides using protein language models. Nat Biotechnol. 2022.
48. Fu L, Niu B, Zhu Z, Wu S, Li W. CD-HIT: accelerated for clustering the next-generation sequencing data. Bioinformatics. 2012;28(23):3150-2.
49. Li W, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics. 2006;22(13):1658-9.
50. Mirdita M, Steinegger M, Soding J. MMseqs2 desktop and local web server app for fast, interactive sequence searches. Bioinformatics. 2019;35(16):2856-8.
51. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. 2019.
52. Lei Ba J, Kiros JR, Hinton GE. Layer Normalization2016 July 01, 2016:[arXiv:1607.06450 p.]. Available from: https://ui.adsabs.harvard.edu/abs/2016arXiv160706450L.
53. Loshchilov I, Hutter F. Decoupled Weight Decay Regularization2017 November 01, 2017:[arXiv:1711.05101 p.]. Available from: https://ui.adsabs.harvard.edu/abs/2017arXiv171105101L.
54. Reeb J, Kloppmann E, Bernhofer M, Rost B. Evaluation of transmembrane helix predictions in 2014. Proteins. 2015;83(3):473-84.
55. Varadi M, Anyango S, Deshpande M, Nair S, Natassia C, Yordanova G, et al. AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. Nucleic Acids Res. 2022;50(D1):D439-D44.
56. Tunyasuvunakool K, Adler J, Wu Z, Green T, Zielinski M, Zidek A, et al. Highly accurate protein structure prediction for the human proteome. Nature. 2021;596(7873):590-6.
57. Bernhofer M, Dallago C, Karl T, Satagopam V, Heinzinger M, Littmann M, et al. PredictProtein - Predicting Protein Structure and Function for 29 Years. Nucleic Acids Res. 2021;49(W1):W535-W40.
58. Sehnal D, Bittrich S, Deshpande M, Svobodova R, Berka K, Bazgier V, et al. Mol* Viewer: modern web app for 3D visualization and analysis of large biomolecular structures. Nucleic Acids Res. 2021;49(W1):W431-W7.

Figures

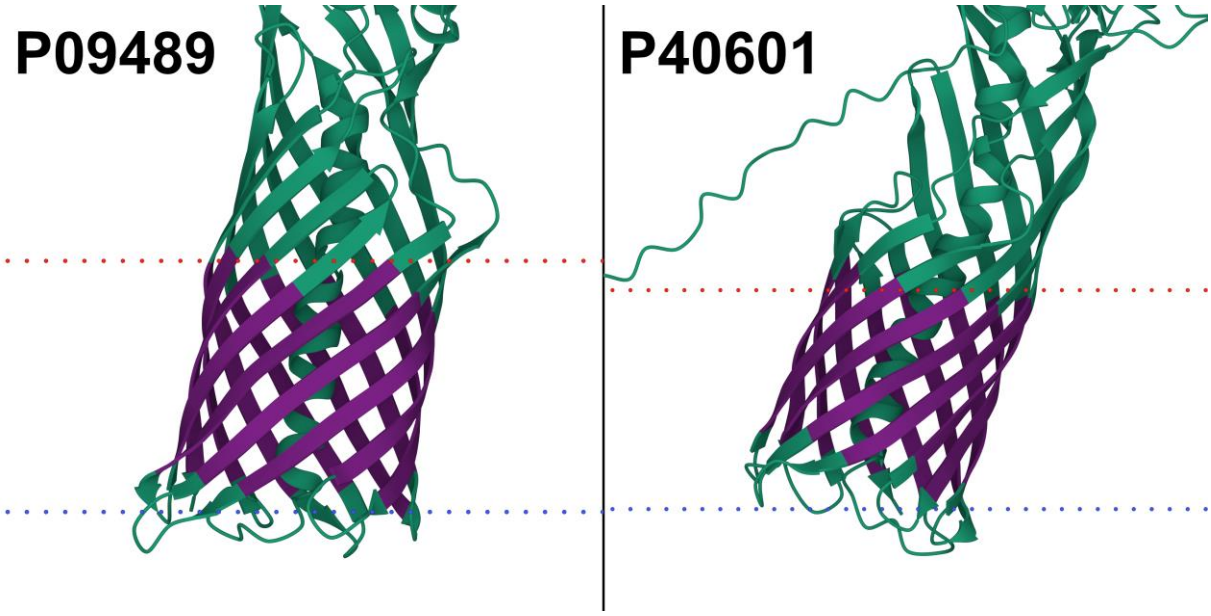

Fig. 1: Potential transmembrane proteins in the globular data set. AlphaFold2 (8, 55) structure of extracellular serine protease (P09489) and Lipase 1 (P40601). Transmembrane segments (dark purple) predicted by TMbed correlate well with membrane boundaries (dotted lines: red=outside, blue=inside) predicted by the PPM (40) web server. Images created using Mol* Viewer (58). Though our data set lists them as globular proteins, the predicted structures indicate transmembrane domains, which align with segments predicted by our method. The predicted domains overlap with autotransporter domains detected by the UniProtKB (41) automatic annotation system. Transmembrane segment predictions were made with the final TMbed ensemble model.

Tables

Table 1: Per-protein performance. *

	β-TMP (65)		α-TMP (593)		Globular (5,859)	
	Recall (%)	FPR (%)	Recall (%)	FPR (%)	Recall (%)	FPR (%)
TMbed	**94.1± 7.1**	***0.1±0.1***	**97.6±0.6**	***0.5±0.2***	***99.5±0.2***	2.7± 1.2
DeepTMHMM	78.4±11.1	***0.1±0.1***	96.0±1.3	***0.5±0.2***	***99.5±0.2***	5.8± 1.9
TMSEG	-	-	96.6±0.9	2.2±0.3	97.8±0.3	3.4± 0.9
TOPCONS2[1]	-	-	94.3±1.4	2.7±0.3	97.3±0.3	5.7± 1.4
OCTOPUS[1]	-	-	94.3±2.0	9.5±0.5	90.5±0.5	5.7± 2.0
Philius[1]	-	-	92.6±1.5	2.6±0.2	97.4±0.2	7.4± 1.5
PolyPhobius[1]	-	-	97.3±1.1	5.2±0.5	94.8±0.5	2.7± 1.1
SPOCTOPUS[1]	-	-	**97.6±1.5**	17.1±0.7	82.9±0.7	**2.4± 1.5**
SCAMPI[1]	-	-	94.3±2.0	10.5±0.6	89.5±0.6	5.7± 2.0
HMM-TM (MSA)[2]	-	-	97.2±1.8	21.5±0.4	78.5±0.4	2.8± 1.8
BOCTOPUS2	84.9±12.4	4.2±0.5	-	-	95.8±0.5	15.1±12.4
BetAware-Deep	86.4± 8.8	4.7±0.4	-	-	95.3±0.4	13.6± 8.8
PRED-TMBB2[3]	89.5±11.2	7.0±0.3	-	-	93.0±0.3	10.5±11.2
PROFtmb	90.8± 7.4	6.1±0.5	-	-	93.9±0.5	9.2± 7.4

* Evaluation of the ability to distinguish between 65 beta barrel TMPs (β-TMP), 593 alpha helical TMPs (α-TMP) and 5,859 globular, water-soluble non-TMP proteins in our data set. Recall and false positive rate (FPR) were averaged over the five independent cross-validation test sets; error margins given for the 95% confidence interval (1.96*standard error); **bold:** best values for each column; *italics:* differences statistically significant with over 95% confidence (only computed between best and 2nd best, or all methods ranked 1 and those ranked lower).

[1] Evaluation missing for one of 5859 globular proteins.

[2] Evaluation includes only 57 β-TMPs, 571 α-TMPs, and 5721 globular proteins due to runtime errors.

[3] The local PRED-TMBB2 version did not include the pre-filtering step of the web server. This caused a FPR for β-TMP of almost 78%. Thus, we listed the statistics for the web server predictions, which did not include MSA input.

Table 2: Per-segment performance for TMH (transmembrane helices). *

	TMH (593 / 3123)				
	Recall (%)	**Precision (%)**	**Q_{ok} (%)**	**Q_{num} (%)**	**Q_{top} (%)**
TMbed	***88.5±1.0***	***88.6±1.3***	***61.9±3.1***	**85.9±1.9**	**95.7±2.4**
DeepTMHMM	79.4±2.2	79.9±2.0	45.9±4.0	85.5±3.5	**95.7±2.2**
TMSEG	74.4±3.0	77.2±2.1	35.4±2.7	69.6±3.6	83.1±4.4
TOPCONS2	76.1±1.8	78.3±0.8	40.5±2.4	74.7±3.2	91.4±3.0
OCTOPUS	71.2±2.0	75.6±1.6	35.6±3.0	66.8±3.7	87.1±2.8
Philius	70.7±2.9	73.8±1.5	33.9±4.1	67.1±4.0	86.9±3.5
PolyPhobius	75.8±2.3	76.5±1.2	40.5±3.6	74.5±2.8	86.6±2.3
SPOCTOPUS	71.1±1.4	75.9±1.3	35.6±2.6	67.0±5.1	87.1±3.3
SCAMPI	69.5±2.5	71.0±1.4	30.2±1.3	69.8±3.0	88.9±4.4
HMM-TM (MSA)[1]	73.0±2.3	72.6±1.7	33.3±1.5	72.2±3.0	88.6±4.2

* Segment performance for transmembrane helix (TMH) prediction based on 593 alpha helical TMPs with a total of 3,123 TMHs. Recall, Precision, Q_{ok}, Q_{num}, and Q_{top} were averaged over the five independent cross-validation test sets; error margins given for the 95% confidence interval (1.96*standard error); **bold:** best values for each column; *italics:* differences statistically significant with over 95% confidence (only computed between best and 2nd best).

[1] Evaluation includes only 571 of the 593 TMHs due to runtime errors of the method.

Table 3: Per-segment performance for TMB (transmembrane beta strands). *

	TMB (65 / 872)				
	Recall (%)	**Precision (%)**	**Q_{ok} (%)**	**Q_{num} (%)**	**Q_{top} (%)**
TMbed	**94.7±3.2**	***99.0± 0.6***	***78.8± 7.6***	**86.1± 3.1**	**98.5±3.0**
DeepTMHMM	86.4±5.0	92.9± 4.2	46.0± 5.4	75.3±11.1	97.8±3.4
BOCTOPUS2	86.7±8.1	96.9± 1.7	59.0±17.0	74.0±11.9	98.3±1.8
BetAware-Deep	68.6±7.0	64.0±11.2	12.7±11.2	65.1±13.3	96.4±4.3
PRED-TMBB2 (MSA)	85.7±2.3	77.1± 4.4	18.9±12.5	45.0±22.4	96.5±2.8
PROFtmb	79.0±9.2	79.1± 5.8	18.6± 9.6	47.5±10.5	97.4±0.8

* Segment performance for transmembrane beta strand (TMB) prediction based on 65 beta barrel TMPs with a total of 872 TMBs. Recall, Precision, Q_{ok}, Q_{num}, and Q_{top} were averaged over the five independent cross-validation test sets; error margins given for the 95% confidence interval (1.96*standard error); **bold:** best values for each column; *italics:* differences statistically significant with over 95% confidence (only computed between best and 2nd best).