nature biotechnology

BRIEF COMMUNICATION

https://doi.org/10.1038/s41587-021-01156-3

Check for updates

OPEN

SignalP 6.0 predicts all five types of signal peptides using protein language models

Felix Teufel[1,2], José Juan Almagro Armenteros[3], Alexander Rosenberg Johansen[4], Magnús Halldór Gíslason[5], Silas Irby Pihl[1], Konstantinos D. Tsirigos[6], Ole Winther[5,7,8], Søren Brunak[3], Gunnar von Heijne[9,10] and Henrik Nielsen[1]✉

Signal peptides (SPs) are short amino acid sequences that control protein secretion and translocation in all living organisms. SPs can be predicted from sequence data, but existing algorithms are unable to detect all known types of SPs. We introduce SignalP 6.0, a machine learning model that detects all five SP types and is applicable to metagenomic data.

SPs are short N-terminal amino acid sequences that target proteins to the secretory (Sec) pathway in eukaryotes and for translocation across the plasma (inner) membrane in prokaryotes. As comprehensive experimental identification of SPs is impractical, computational prediction of SPs has high relevance to research in cell biology[1]. SP prediction tools enable identification of proteins that follow the general secretory or twin-arginine translocation (Tat) pathway and predict the position in the sequence where a signal peptidase (SPase) cleaves the SP[2,3]. SignalP 5.0 is able to predict Sec substrates cleaved by SPase I (Sec/SPI) or SPase II (Sec/SPII, prokaryotic lipoproteins) and Tat substrates cleaved by SPase I (Tat/SPI)[4]. However, due to a lack of annotated data, SignalP 5.0 is unable to detect Tat substrates cleaved by SPase II or Sec substrates processed by SPase III (prepilin peptidase, sometimes referred to as SPase IV[2]). Such Sec/SPIII SPs control the translocation of type IV pilin-like proteins, which play a key role in adhesion, motility and DNA uptake in prokaryotes[5]. Furthermore, SignalP 5.0 is agnostic regarding the SP structure, as it cannot define the subregions (the N-terminal n-region, the hydrophobic h-region, and the C-terminal c-region) that underlie the biological function of SPs.

Here, we present SignalP 6.0, based on protein language models (LMs)[6–9] that use information from millions of unannotated protein sequences across all domains of life. LMs create semantic representations of proteins that capture their biological properties and structure. Using these protein representations, SignalP 6.0 can predict additional types of SPs that previous versions have been unable to detect while better extrapolating to both proteins distantly related to those used to create the model and metagenomic data of unknown origin. In addition, it is capable of identifying the subregions of SPs.

We compiled a comprehensive dataset of protein sequences that are known to harbor SPs, containing 3,352 Sec/SPI, 2,261 Sec/SPII, 113 Sec/SPIII, 595 Tat/SPI, 36 Tat/SPII, 16,421 intracellular sequences and 2,615 transmembrane sequences (Methods). Moreover, we defined region-labeling rules according to known properties of the SP types (Fig. 1a and Methods). We applied threefold nested cross-validation to train and evaluate the model (Methods and Supplementary Note 1). In our data-partitioning procedure, we ensured that homologous sequences were placed in the same partition to accurately measure the model's performance on unseen sequences.

For previous predictors, the SP types Sec/SPIII and Tat/SPII were omitted due to a lack of annotated samples, which makes learning their defining features challenging for models[4]. Notably, this lack does not correspond to prevalence in nature, as these types exist throughout most organisms present in the databases[10,11]. In addition, the available annotated sequences do not cover the full diversity encountered in nature, as they are biased towards well-studied organisms. Furthermore, existing predictors require data for which the organism of origin is known, as this allows the predictors to explicitly account for known differences in SP structure among Eukarya, Archaea and Gram-positive and Gram-negative bacteria.

Protein LMs have been shown to improve performance on problems with limited annotated data[12]. Moreover, LM protein representations directly capture the evolutionary context of a sequence[6,8]. We hypothesized that when using an LM, we would (1) obtain better performance on SP types with limited data availability, (2) achieve better generalization to sequences that are distantly related to training sequences and (3) enable the prediction of sequences for which the species of origin is unknown. We opted for the bidirectional encoder representations from transformers (BERT) protein LM, which is available in ProtTrans[6,7] and was trained on UniRef100 (ref. [13]) (Fig. 1b). The LM was subsequently optimized on our dataset to predict SPs. We found that even before optimization, the LM captured the presence of SPs in its protein representations (Fig. 1c). We combined the LM with a conditional random field (CRF) probabilistic model[14] to predict the SP region at each sequence position together with the SP type, yielding the SignalP 6.0 architecture (Fig. 1d).

As the baseline for evaluation, we retrained SignalP 5.0 on our new dataset. We measure performance for each SP type separately per organism group (Archaea, Eukarya, Gram-positive and Gram-negative bacteria), reporting the Matthews correlation coefficient (MCC) for correctly detecting the SP type among both non-SP and other types of sequences as negative samples. For all categories

[1]Section for Bioinformatics, Department of Health Technology, Technical University of Denmark, Kongens Lyngby, Denmark. [2]Department of Biosystems Science and Engineering, ETH Zurich, Basel, Switzerland. [3]Novo Nordisk Foundation Center for Protein Research, Faculty of Health and Medical Sciences, University of Copenhagen, Copenhagen, Denmark. [4]Department of Computer Science, Stanford University, Stanford, CA, USA. [5]Center for Genomic Medicine, Rigshospitalet (Copenhagen University Hospital), Copenhagen, Denmark. [6]EMBL-EBI, Wellcome Genome Campus, Cambridge, UK. [7]Department of Biology, Bioinformatics Centre, University of Copenhagen, Copenhagen, Denmark. [8]Section for Cognitive Systems, Department of Applied Mathematics and Computer Science, Technical University of Denmark, Kongens Lyngby, Denmark. [9]Department of Biochemistry and Biophysics, Stockholm University, Stockholm, Sweden. [10]Science for Life Laboratory, Stockholm University, Solna, Sweden. ✉e-mail: henni@dtu.dk

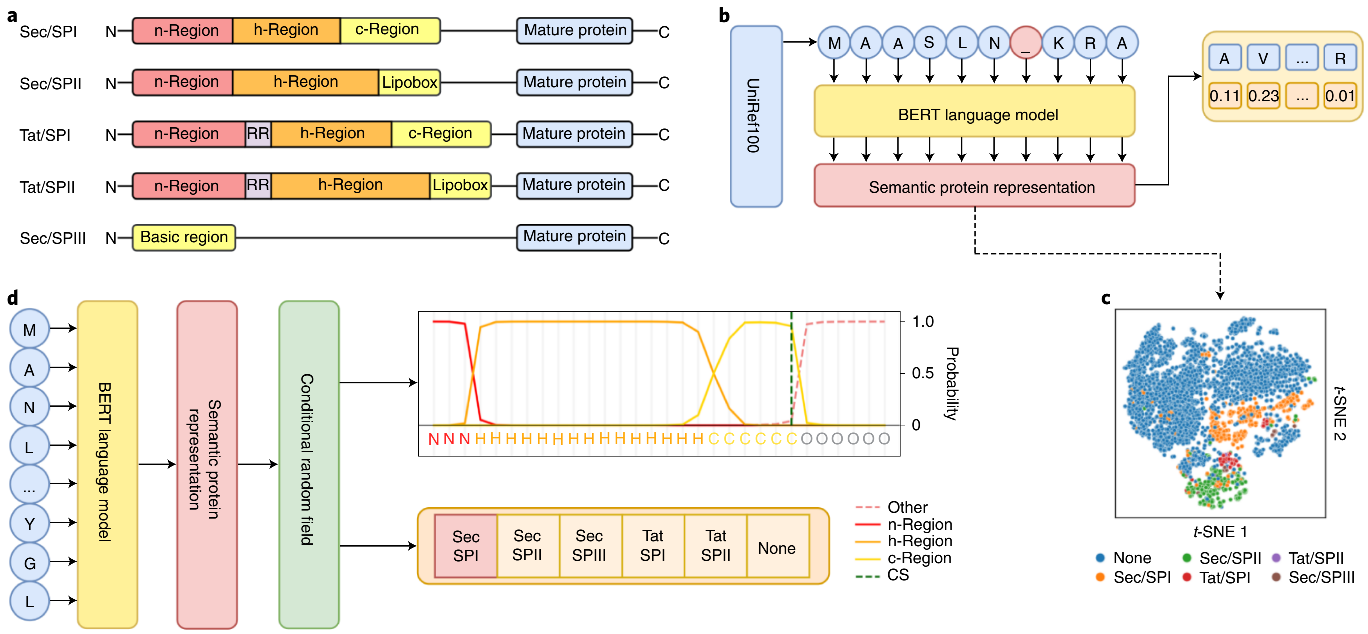
a
Sec/SPI N– n-Region h-Region c-Region Mature protein –C
Sec/SPII N– n-Region h-Region Lipobox Mature protein –C
Tat/SPI N– n-Region RR h-Region c-Region Mature protein –C
Tat/SPII N– n-Region RR h-Region Lipobox Mature protein –C
Sec/SPIII N– Basic region Mature protein –C
b
UniRef100
M A A S L N _ K R A
BERT language model
Semantic protein representation
A V ... R
0.11 0.23 ... 0.01
c
t-SNE 1
t-SNE 2
None
Sec/SPI
Sec/SPII
Tat/SPI
Tat/SPII
Sec/SPIII
d
M A N L ... Y G L
BERT language model
Semantic protein representation
Conditional random field
1.0
0.5
0
Probability
N N N H H H H H H H H H H H H H H C C C C C C O O O O O O
Sec SPI
Sec SPII
Sec SPIII
Tat SPI
Tat SPII
None
Other
n-Region
h-Region
c-Region
CS

Fig. 1 | Modeling SP structure using protein LMs. a, Region structures of the five SP types. Twin arginine (RR)-translocated SPs feature a twin-arginine motif, while SPs cleaved by SPase II feature a C-terminal lipobox. Sec/SPIII SPs have no substructure. **b**, Protein LM training procedure. BERT learns protein features by predicting masked amino acids in sequences from UniRef100. **c**, *t*-Distributed stochastic neighbor embedding (*t*-SNE) projection of protein representations before prediction training. Different SP types form distinct clusters, separated from sequences without SPs. **d**, SignalP 6.0 architecture. An amino acid sequence is passed through the LM, and the resulting representation serves as input for the CRF, which predicts region probabilities at each position and the SP type. CS, cleavage site.

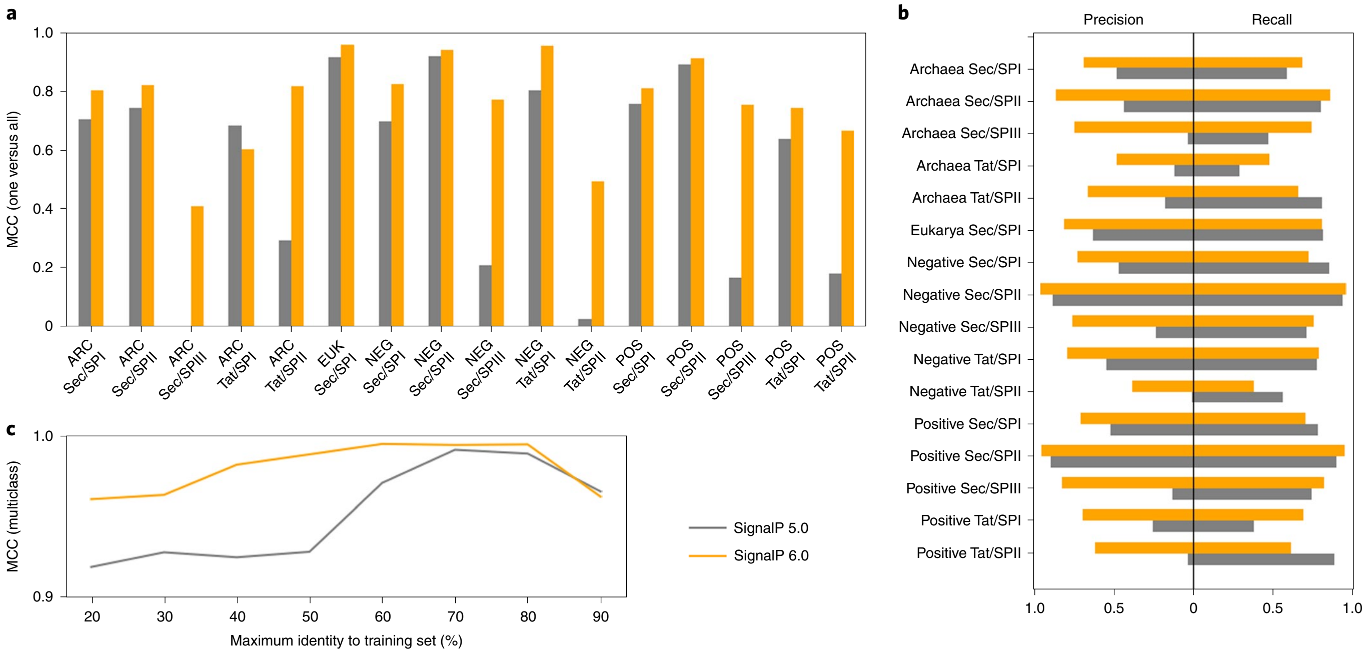

Fig. 2 | SignalP 6.0 shows strong performance on all types and organism groups. a, SP detection performance (ARC, Archaea; EUK, Eukarya; NEG, Gram-negative bacteria; POS, Gram-positive bacteria). SignalP 6.0 substantially improves performance on underrepresented types. **b**, CS prediction performance. SignalP 6.0 has improved precision for all categories. **c**, Dependence of performance on identity to sequences in the training data. At sequence identities lower than 60%, SignalP 6.0 outperforms SignalP 5.0.

except Tat/SPI in Archaea, SignalP 6.0 showed improved performance. Detection performance improved substantially, especially for the two underrepresented types, Sec/SPIII and Tat/SPII (Fig. 2a and Supplementary Fig. 1), whereas the performance of SignalP 5.0 remained too low to make it practically useful. This confirms the importance of LMs for low-data problems, making SignalP 6.0

a model capable of simultaneously detecting all five types of SPs. In addition, we found substantial precision gains for predicting cleavage sites (CSs) (Fig. 2b).

We further benchmarked SignalP 6.0 against other publicly available predictors. In some cases, specialized predictors show stronger performance on the specific tasks they were optimized for (Supplementary Figs. 2 and 3 and Supplementary Tables 1–6). However, none of these predictors are capable of detecting all SP types, and the results are further biased, as they cannot be evaluated in a cross-validated setup.

When predicting a set of test sequences grouped by identity to any sequence in the training data, we find that detection performance at high sequence identities remained comparable. However, at identities lower than 60%, SignalP 6.0 outperformed SignalP 5.0, showing better generalization to proteins distantly related to those present in the training data (Fig. 2c).

Most SP predictors require knowledge of a sequence's organism group of origin for optimal performance[4,15,16]. SignalP 6.0 does not show reduced performance if this information is removed, indicating that the evolutionary context, as encoded in the LM representation, already captures the organism group (Supplementary Fig. 4). Ultimately, this makes SignalP 6.0 a multiclass SP prediction tool that is applicable to sequences of unknown origin, as is typically the case in metagenomic and metatranscriptomic assemblies. However, SignalP 6.0 still relies on start codons being correctly identified before application. For context, 1.7% of UniProt release 2021_02 entries (i.e., 3.5 million sequences) have no organism specified.

SPs are traditionally described as consisting of three regions. We benchmarked our region identification by comparing the properties of predicted regions to known properties[17], with predictions matching all expected properties (Supplementary Note 2 and Supplementary Fig. 5). We additionally predicted a library of synthetic SPs that are either functional or nonfunctional in *Bacillus subtilis*[18], revealing significant differences in the two groups' regions that could not be identified before by traditional sequence analysis (Supplementary Fig. 6 and Supplementary Table 7).

This study presents SignalP 6.0, a machine learning model covering all five known types of SPs that accurately predicts both sequences of unknown origin and evolutionarily distant proteins. Through the use of protein LMs, SignalP 6.0 is able to predict types with very limited training data available. By making the full spectrum of SPs accessible, the model allows us to further improve our understanding of protein translocation throughout evolution (Supplementary Note 3 and Supplementary Tables 8 and 9). In addition, identification of SP regions opens up new avenues into researching the defining properties that govern SP functionality. Given the potential of SPs as drug targets[19] and their emerging role in synthetic biology[18], investigating SPs and their properties at scale may lead to further advances in these fields.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41587-021-01156-3.

Received: 22 June 2021; Accepted: 8 November 2021;
Published online: 3 January 2022

References

1. Nielsen, H., Tsirigos, K. D., Brunak, S. & von Heijne, G. A brief history of protein sorting prediction. *Protein J.* **38**, 200–216 (2019).
2. Dalbey, R. E., Wang, P. & van Dijl, J. M. Membrane proteases in the bacterial protein secretion and quality control pathway. *Microbiol. Mol. Biol. Rev.* **76**, 311–330 (2012).
3. Pohlschroder, M., Pfeiffer, F., Schulze, S. & Halim, M. F. A. Archaeal cell surface biogenesis. *FEMS Microbiol. Rev.* **42**, 694–717 (2018).
4. Almagro Armenteros, J. J. et al. SignalP 5.0 improves signal peptide predictions using deep neural networks. *Nat. Biotechnol.* **37**, 420–423 (2019).
5. Craig, L., Forest, K. T. & Maier, B. Type IV pili: dynamics, biophysics and functional consequences. *Nat. Rev. Microbiol.* **17**, 429–440 (2019).
6. Elnaggar, A. et al. ProtTrans: towards cracking the language of life's code through self-supervised learning. *Trans. Pattern Anal. Mach. Intell.* https://pubmed.ncbi.nlm.nih.gov/34232869/ (2021).
7. Dallago, C. et al. Learned embeddings from deep learning to visualize and predict protein sets. *Curr. Protoc.* **1**, e113 (2021).
8. Alley, E. C., Khimulya, G., Biswas, S., AlQuraishi, M. & Church, G. M. Unified rational protein engineering with sequence-based deep representation learning. *Nat. Methods* **16**, 1315–1322 (2019).
9. Rives, A. et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl Acad. Sci. USA* **118**, e2016239118 (2021).
10. Storf, S. et al. Mutational and bioinformatic analysis of haloarchaeal lipobox-containing proteins. *Archaea* **2010**, 410975 (2010).
11. Hutchings, M. I., Palmer, T., Harrington, D. J. & Sutcliffe, I. C. Lipoprotein biogenesis in Gram-positive bacteria: knowing when to hold 'em, knowing when to fold 'em. *Trends Microbiol.* **17**, 13–21 (2009).
12. Biswas, S., Khimulya, G., Alley, E. C., Esvelt, K. M. & Church, G. M. Low-N protein engineering with data-efficient deep learning. *Nat. Methods* **18**, 389–396 (2021).
13. Suzek, B. E., Wang, Y., Huang, H., McGarvey, P. B. & Wu, C. H. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* **31**, 926–932 (2015).
14. Lafferty, J. D., McCallum, A. & Pereira, F. C. N. In *Proc. 18th International Conference on Machine Learning* (eds. Brodley, C.E. & Danyluk, A.P.) 282–289 (Morgan Kaufmann Publishers, 2001).
15. Savojardo, C., Martelli, P. L., Fariselli, P. & Casadio, R. DeepSig: deep learning improves signal peptide detection in proteins. *Bioinformatics* **34**, 1690–1696 (2018).
16. Zhang, W.-X., Pan, X. & Shen, H.-B. Signal-3L 3.0: improving signal peptide prediction through combining attention deep learning with window-based scoring. *J. Chem. Inf. Model.* **60**, 3679–3686 (2020).
17. Owji, H., Nezafat, N., Negahdaripour, M., Hajiebrahimi, A. & Ghasemi, Y. A comprehensive review of signal peptides: structure, roles, and applications. *Eur. J. Cell Biol.* **97**, 422–441 (2018).
18. Wu, Z. et al. Signal peptides generated by attention-based neural networks. *ACS Synth. Biol.* **9**, 2154–2161 (2020).
19. Lumangtad, L. A. & Bell, T. W. The signal peptide as a new target for drug design. *Bioorg. Med. Chem. Lett.* **30**, 127115 (2020).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.
© The Author(s) 2022

Methods

Sequence data. The dataset for SignalP 6.0 was obtained by extending the data published with SignalP 5.0 (ref. [4]). For all classes that were already part of the original data (Sec/SPI, Sec/SPII, Tat/SPI and soluble and transmembrane proteins), we added sequences that had become available in the respective source databases (UniProt[20] and Prosite[21] for SPs and UniProt and TOPDB[22] for soluble and transmembrane proteins) from 2018 until 7 November 2020, following the original selection criteria.

Tat/SPII sequences were identified using the combination of Prosite profiles PS51318 (Tat motif) and PS51257 (lipoprotein motif). By default, PS51318 is subject to postprocessing that prevents both profiles from matching the same sequence. As there is experimental evidence for the existence of Tat-translocated lipoproteins[10,11], we considered this postprocessing rule to be biologically implausible. We disabled it manually in ScanProsite[23] and scanned all prokaryotic sequences in Swiss-Prot, yielding a total of 25 sequences in which both profiles matched. Additional Tat/SPII sequences were found by training a simplified SignalP 6.0 model to discriminate SPII from non-SP sequences. We used this model to predict all Tat/SPI sequences in the training data, as we assumed that PS51257 is not sensitive enough to find all lipoproteins. We investigated the resulting hits in UniProt for supporting evidence that the proteins were true lipoproteins, yielding 12 sequences that we relabeled Tat/SPII from Tat/SPI. One additional sequence with manual evidence was found in the TatLipo 1.03 training data[10]. For Sec/SPIII sequences, we used Prosite pattern PS00409 for bacteria and Pfam[24] family PF04021 for Archaea, yielding 103 and 10 sequences, respectively.

We improved the organism type classification of sequences by defining Gram-negative and Gram-positive bacteria more stringently, as we found that for edge cases such as Thermotogae, in which both gram stains can be observed[25], the classification in SignalP 5.0 was unclear. We redefined Gram positive as all bacterial phyla that have a single membrane (monoderm): Actinobacteria, Firmicutes, Tenericutes, Thermotogae, Chloroflexi and Saccharibacteria. All remaining phyla have a double membrane (diderm) and were classified as Gram negative.

We followed the methodology introduced by Gíslason et al.[26] for homology partitioning of the dataset into three partitions at 30% sequence identity. In brief, it achieves partitioning by computing the pairwise global sequence identities of all sequences using the Needleman–Wunsch algorithm[27], followed by single-linkage clustering. The resulting clusters were grouped together into the desired number of partitions. If there were sequences in a partition that had pairwise identities to any sequence in another partition that were higher than the defined threshold, then those sequences were iteratively removed until the maximum sequence identity criterion was fulfilled. We performed the partitioning procedure separately for each SP type and the negative set, yielding three partitions for each of the six classes. The algorithm was further constrained to ensure that each generated partition was balanced for the four organism groups. We concatenated the resulting 3 × 6 partitions to yield the three final partitions for cross-validation, thereby ensuring that both the SP types and the organism groups were equally represented across partitions.

The CD-HIT clustering method[28] that was employed in SignalP 5.0 enforces the homology threshold for cluster centers. However, as the training set was not homology reduced but rather homology clustered, other data points can have a homology overlap notably above the chosen threshold of 20% (Supplementary Fig. 7). When using the partitioning method of Gíslason et al., which strictly enforces the defined threshold, 20% maximum identity was impossible to achieve. Even at the relaxed threshold of 30%, the procedure resulted in the removal of a substantial part of the dataset to achieve separation in three partitions (Supplementary Table 10).

For benchmarking, we reused the benchmark set of SignalP 5.0, from which we excluded all sequences that were removed in the homology partitioning procedure of the new dataset. For sequences that were reclassified (to Gram positive or Tat/SPII), we changed the label accordingly (Supplementary Table 11).

For the synthetic SP dataset, we used the data reported by Wu et al.[18]. We gathered all synthetic SP-mature protein pairs that were experimentally characterized, yielding 57 nonfunctional and 52 functional sequences. For the region analysis, we only considered sequences predicted as Sec/SPI SPs by SignalP 6.0, reducing the number of nonfunctional sequences to 55.

Reference proteomes and proteins of unknown origin were obtained from UniProt release 2021_02. To identify sequences of unknown origin, we used taxonomy identifiers 48479 (environmental samples), 49928 (unclassified bacteria) and 2787823 (unclassified entries).

Generation of SP region labels. We defined the task of learning SP regions as a multilabel classification problem at each sequence position. Multilabel differs from multiclass in the sense that more than one label can be true at a given position. This approach was motivated by the fact that there is no strict definition of region borders that is commonly agreed upon, making it impossible to establish ground-truth region labels for models to train on. We thus used the multilabel framework as a method for training with weak supervision, allowing us to use overlapping region labels during the learning phase that could be generated from the sequence data using rules. For inference, we did not make use of the multilabel framework, as we only predicted the single most probable label at each position using Viterbi decoding, yielding a single unambiguous solution.

We defined a set of three rules based on known properties of the n-, h-, and c-regions. The initial n-region must have a minimum length of two residues and the terminal c-region a minimum length of three residues. The most hydrophobic position, which is identified by sliding a seven-amino-acid window across the SP and computing the hydrophobicity using the Kyte–Doolittle scale[29], belongs to the h-region. All positions between these six labeled positions are labeled as either both n and h or h and c, yielding multitag labels.

This procedure was adapted for different SP classes, with only Sec/SPI completely following it. For Tat SPs, the n–h border was identified using the twin-arginine motif. All positions before the motif were labeled n, followed by two dedicated labels for the motif, again followed by a single position labeled n. For SPII SPs, we did not label a c-region, as the C-terminal positions cannot be considered as such[30]. The last three positions were labeled as the lipobox, all positions before that as h only. For SPIII SPs, no region labels were generated within the SP.

Modeling. SignalP 6.0 uses a pretrained protein LM to encode the amino acid sequence and a CRF[14] decoder to predict the regions, CSs and sequence class labels. Specifically, we used the 30-layer BERT LM [31] that is available in ProtTrans[6], which was pretrained on UniRef100 (ref. [13]). We removed the last layer of the pretrained model and extended the pretrained embedding layer by four additional randomly initialized vectors to represent the tokens for the four organism group identifiers. We prepend the organism group identifier to each sequence s of length T and encode it. From the resulting sequence of hidden states, we trim the positions corresponding to the organism group token and the special sequence start and end tokens used by BERT (CLS and SEP) to obtain a sequence of hidden states h of equal length as the original amino acid input x:

$$h = \mathrm{BERT}(x).$$

The hidden states serve as input for a linear-chain CRF. The CRF models the conditional probability of a sequence of states $y = y_1 \ldots y_t$ given a sequence of hidden states $h = \mathbf{h}_1 \ldots \mathbf{h_t}$ using the following factorization:

$$P(y|h) = \frac{1}{Z(h)} \prod_{t=1}^{T} \exp(\psi(\mathbf{h_t})) \prod_{t=1}^{T-1} \exp(\varphi_{y_t, y_{t+1}}),$$

where $Z(h)$ is the normalization constant of the modeled distribution; φ is the learnable transition matrix of the CRF with $C \times C$ parameters, with C being the number of states (labels) modeled by the CRF; and ψ is a learnable linear transformation that maps from the dimension of the hidden state h to the number of CRF states C, yielding the emissions for the CRF:

$$\psi(h_t) = W_{\psi} \mathbf{h_t} + b_{\psi}.$$

For each class of SP G, there are multiple possible CRF states, corresponding to the defined regions of the SP class. We constrained the transitions in φ to ensure that regions are predicted in the correct order, leading to the possible state sequences depicted in Supplementary Fig. 8.

For inference, we compute both the most probable state sequence (using Viterbi decoding) and the marginal probabilities at all sequence positions (using the forward-backward algorithm). The most probable state sequence is used to predict the CS, which is inferred from the last predicted SP state as indicated in Supplementary Fig. 8.

As each SP consists of multiple regions, multiple states of C belong to a single global sequence class G. To predict the global class probabilities, we sum the marginal probabilities of all states that belong to a given class and divide the sum by the sequence length. This transforms a matrix of probabilities of shape C × T to a G × 1 vector of global class probabilities:

$$p(G_i|x) = \frac{1}{T} \prod_{t=1}^{T} \sum_{C \in G_i} p(y_{Ct}|x).$$

Training. For training, we minimize the negative log likelihood of the CRF. As we can have multiple true labels y_t at a given position, we use an extension of the equation known as multitag CRF. Multiple labels are handled by summing over the set of true labels M_t at each position:

$$-\log\left(P\left(y|h\right)\right) = \log\left(Z\left(h\right)\right) - \log\left(\exp\left(\sum_{t=1}^{T} \sum_{y_t \in M_t} \psi\left(\mathbf{h_t}\right) + \varphi\left(y_t, y_{t-1}\right)\right)\right).$$

As we designed our region labels to be overlapping, the model is free to distribute its probability mass in any ratio between the correct labels at a given position. There are thus multiple solutions for the specific borders of n-, h- and c-regions that yield the same negative log likelihood but are not equally biologically plausible. For instance, the model could learn a solution where it uniformly predicts an n-region of length 2 in all SPs, irrespective of the actual sequence. We employ regularization to promote the finding of biologically plausible solutions. Our regularization is based on the fact that the three SP regions have divergent amino acid compositions, which we can quantify by computing the cosine similarity between the amino acid distributions.

The most obvious approach would be to compute the amino acid distribution of each region based on the region borders inferred from the predicted most probable path of the sequence. This, however, cannot be used for regularization, as we require the term to be differentiable, which our Viterbi decoding implementation is not. We therefore based our regularization term on the marginal probabilities of the CRF computed by the forward-backward algorithm, which are used to compute a score for each amino acid for each region, approximating the discrete amino acid distributions.

For each region $r \in \{n, h, c\}$, we sum the marginal probabilities of all CRF states c belonging to region r at position t, yielding $s_{t,r}$. We sum $s_{t,r}$ of all positions t of the sequence that have amino acid a, yielding the elements of the score vector $\mathbf{score}_r$ for each region. We compute the cosine similarity between the normalized score vectors of n and h and h and c:

$$s_{t,r} = \sum_{c \in r} p\left(y_{t,c}|x\right)$$

$$\mathrm{score}_{a,r} = \sum_{t \in I} s_{t,r}$$

$$I = \{t \in T | x_t = a\}$$

$$\mathbf{score_r}' = \mathbf{score_r} / \sum_{a=1}^{A} \mathrm{score}_{a,r}$$

We perform this operation for each sequence. Sequences for which a region does not exist (for example no c-region in Sec/SPII) are ignored for the respective similarity. The mean over all sequences for both similarities, multiplied by a factor α, was added to the loss. We observed that for about half the random seeds we tested, training runs with regularization enabled converged to a n-region length of 2 after one epoch. This is a degenerate solution, as this causes the n-region amino acid distribution to be nonzero at a single position, yielding low similarity scores while being biologically implausible (a length of 2 is expected as the minimum, not the average over all sequences). Such runs were stopped and discarded after one epoch.

The model was trained end-to-end, including all layers of BERT for 15 epochs, using Adamax as the optimizer and a slanted triangular learning rate. We applied dropout on the hidden state outputs of BERT to avoid overfitting. Hyperparameters were optimized using SigOpt (https://app.sigopt.com/docs/intro/overview). We employed threefold nested cross-validation (outer loop is threefold and inner loop is twofold), yielding a total of 3 × 2 models for evaluation.

Evaluation and benchmarking. For comparability, we employed the same metrics that were used in SignalP 5.0. SP detection performance was measured using the MCC[32]. We computed the MCC twice, once with the negative set only consisting of transmembrane and soluble proteins (MCC1) and once with it additionally including sequences of all other SP types (MCC2). Most of the competing single-class predictors considered for benchmarking are optimized for detecting their respective SP type in a dataset of true and non-SP (soluble and transmembrane) sequences; thus, MCC1 best captures their performance on the task they were designed for. MCC2, on the other hand, includes the more challenging task of discriminating between SP types, which is difficult for single-class predictors because of the structural similarity of different SP types. MCC2 represents the performance in most real-world applications, as the presence of a specific SP type usually cannot be ruled out a priori in a set of unknown protein sequences. For CS prediction, we computed the precision and recall. Precision was defined as the fraction of correct CS predictions over the number of predicted CSs, and recall was defined as the fraction of correct CS predictions over the number of true CSs. In both cases, a CS was only considered correct if it was predicted in the correct SP class (e.g., when the model predicts a CS in a Sec/SPI sequence but predicts Sec/SPII as the sequence label, then the sample is considered 'no CS predicted'). To account for possible uncertainty of the CSs in the training data labels, we additionally report these metrics with tolerance windows of one, two and three residues left and right of the true CS (Supplementary Tables 2, 4 and 6).

For the predicted SP regions, in the absence of true labels, no quantitative performance metrics could be established. To still be able to assess the quality of the predictions, we compared the properties of predicted regions with characteristics of regions that are described in the literature. We followed the review by Owji et al.[17] as a guideline to identify region characteristics. Specifically, we evaluated the length, hydrophobicity and charge of each predicted region. Hydrophobicities were computed using the Kyte–Doolittle scale[29], and charges were computed by summing the net charges at pH 7 of all residues. The net charge computation differed between the groups, as in Eukarya and Archaea the N-terminal methionine is not formylated[33], thus contributing an additional positive charge to the n-region by its amino group.

We benchmarked our model against the state-of-the-art model SignalP 5.0, which was reimplemented in PyTorch. Hyperparameter optimization on the new dataset was performed using SigOpt. We also repeated the benchmarking experiment of SignalP 5.0 for all predictors using the adapted benchmark set. We could not add Signal-3L 3.0[16] to the experiment, as the implementation that is available does not allow for processing of more than one sequence at a time, rendering benchmarking intractable. Notably, predictions for all methods except for SignalP 5.0 and SignalP 6.0 were obtained from their publicly available web services, resulting in potential performance overestimation due to the lack of homology partitioning. In addition, performance overestimation is still present for the published version of SignalP 5.0 (named "SignalP 5.0 original" in Supplementary Tables 1–6 and Supplementary Figs. 1 and 2) due to insufficient homology partitioning of its training data by CD-HIT. We thus excluded its values from determining the best-performing tools in the benchmark.

To assess the effect of sequence identity to training sequences on performance, we used the set of sequences that were removed by the partitioning procedure. We predicted all sequences in the removed set and binned the sequences according to the maximum sequence identity to any sequence in the training set. We did this for all six cross-validated models and pooled the resulting binned predictions. For each bin, we computed the multiclass MCC as defined by Gorodkin[34].

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article

Data availability

The datasets used for training and testing SignalP 6.0 can be downloaded from https://services.healthtech.dtu.dk/service.php?SignalP-6.0. The investigated reference proteomes are available from UniProt at https://www.uniprot.org/proteomes. The dataset of synthetic SPs was extracted from the supplementary material of the original publication[18] and is included in our GitHub repository.

Code availability

SignalP 6.0 is available at https://services.healthtech.dtu.dk/service.php?SignalP-6.0. The web version of SignalP 6.0 is free for all users, while the standalone Python package is free for academic users (and can be provided upon request) but is licensed for a fee to commercial users. The model source code in PyTorch 1.7 is available at https://github.com/fteufel/signalp-6.0.

References

20. The UniProt Consortium. UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Res.* **47**, D506–D515 (2019).
21. Sigrist, C. J. A. et al. New and continuing developments at PROSITE. *Nucleic Acids Res.* **41**, D344–D347 (2013).
22. Dobson, L., Langó, T., Reményi, I. & Tusnády, G. E. Expediting topology data gathering for the TOPDB database. *Nucleic Acids Res.* **43**, D283–D289 (2015).
23. de Castro, E. et al. ScanProsite: detection of PROSITE signature matches and ProRule-associated functional and structural residues in proteins. *Nucleic Acids Res.* **34**, W362–W365 (2006).
24. Mistry, J. et al. Pfam: the protein families database in 2021. *Nucleic Acids Res.* **49**, D412–D419 (2021).
25. Bhandari, V. & Gupta, R. S. in *The Prokaryotes: Other Major Lineages of Bacteria and The Archaea* (eds. Rosenberg, E., et al.) 989–1015 (Springer, 2014).
26. Gíslason, M. H., Nielsen, H., Almagro Armenteros, J. J. & Johansen, A. R. Prediction of GPI-anchored proteins with pointer neural networks. *Curr. Res. Biotechnol.* **3**, 6–13 (2021).
27. Needleman, S. B. & Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.* **48**, 443–453 (1970).
28. Fu, L., Niu, B., Zhu, Z., Wu, S. & Li, W. CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics* **28**, 3150–3152 (2012).
29. Kyte, J. & Doolittle, R. F. A simple method for displaying the hydropathic character of a protein. *J. Mol. Biol.* **157**, 105–132 (1982).
30. von Heijne, G. The signal peptide. *J. Membr. Biol.* **115**, 195–201 (1990).
31. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. Preprint at http://arxiv.org/abs/1810.04805 (2019).
32. Matthews, B. W. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim. Biophys. Acta BBA - Protein Struct.* **405**, 442–451 (1975).
33. Ramesh, V. & RajBhandary, U. L. Importance of the anticodon sequence in the aminoacylation of tRNAs by methionyl-tRNA synthetase and by valyl-tRNA synthetase in an Archaebacterium. *J. Biol. Chem.* **276**, 3660–3665 (2001).
34. Gorodkin, J. Comparing two K-category assignments by a K-category correlation coefficient. *Comput. Biol. Chem.* **28**, 367–374 (2004).

Acknowledgements

J.J.A.A. and S.B. were supported by the Novo Nordisk Foundation (grants NNF14CC0001 and NNF17OC0027594). O.W. was funded in part by the Novo Nordisk Foundation through the Center for Basic Machine Learning Research in Life Science (grant NNF20OC0062606). G.v.H. was supported by the Knut and Alice Wallenberg Foundation (grant 2017.0323), the Novo Nordisk Foundation (grant NNF18OC0032828) and the Swedish Research Council (grant 621-2014-3713).

Author contributions

F.T. collected the datasets, implemented the SignalP 6.0 model and performed the experiments with help from J.J.A.A. and A.R.J. S.I.P. implemented the multitask CRF and developed the Sec/SPI labeling procedure. M.H.G. provided the partitioning code and guided its application. K.D.T., O.W., S.B. and G.v.H. provided suggestions during the design of SignalP 6.0. J.J.A.A., A.R.J. and H.N. supervised and guided the project. All authors edited and approved the manuscript.

Competing interests

The downloadable version of SignalP 6.0 has been commercialized (it is licensed for a fee to commercial users). The revenue from these commercial sales is divided between the program developers and the Technical University of Denmark.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41587-021-01156-3.

Correspondence and requests for materials should be addressed to Henrik Nielsen.

Peer review information *Nature Biotechnology* thanks Rita Casadio and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

nature research

Corresponding author(s): Henrik Nielsen

Last updated by author(s): Nov 1, 2021

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
□	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☒	□	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
□	☒	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	□	A description of all covariates tested
☒	□	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
□	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
□	☒	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	□	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	□	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
□	☒	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	Python 3.6
Data analysis	Python 3.6, PyTorch 1.7, https://github.com/fteufel/signalp-6.0, https://services.healthtech.dtu.dk/service.php?SignalP-6.0

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

All data used for training and testing SignalP 6.0 can be downloaded at https://services.healthtech.dtu.dk/service.php?SignalP-6.0

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences　　□ Behavioural & social sciences　　□ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	The total number of protein sequences we used for development and benchmarking of the SignalP 6.0 model was 20,290. The synthetic SP dataset consisted of 109 sequences. 9,915 UniProt reference proteomes were used. Sample sizes were determined by the availability of data.
Data exclusions	Protein sequences with signal peptides longer than 70 amino acids were not included in the training set. These are very rare (currently, there are 3 examples in UniProt), and increasing the length beyond 70 would reduce computational speed. Viral reference proteomes were excluded from the signal peptide frequency analysis, since viral proteomes are generally so small that it would not make sense to do a frequency calculation.
Replication	No experimental data were generated in this study.
Randomization	No experimental data were generated in this study. No group allocation was performed.
Blinding	No group allocation was performed.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	□ Antibodies
☒	□ Eukaryotic cell lines
☒	□ Palaeontology and archaeology
☒	□ Animals and other organisms
☒	□ Human research participants
☒	□ Clinical data
☒	□ Dual use research of concern

Methods

n/a	Involved in the study
☒	□ ChIP-seq
☒	□ Flow cytometry
☒	□ MRI-based neuroimaging