communications chemistry

A Nature Portfolio journal

Article

https://doi.org/10.1038/s42004-025-01540-z

Leveraging pretrained deep protein language model to predict peptide collision cross section

Ayano Nakai-Kasai [1], Kosuke Ogata [2], Yasushi Ishihama [2,3,5] ✉ & Toshiyuki Tanaka [4,5] ✉

Collision cross section (CCS) of peptide ions provides an important separation dimension in liquid chromatography/tandem mass spectrometry-based proteomics that incorporates ion mobility spectrometry (IMS), and its accurate prediction is the basis for advanced proteomics workflows. This paper describes experimental data and a prediction model for challenging CCS prediction tasks including longer peptides that tend to have higher charge states. The proposed model is based on a pretrained deep protein language model. While the conventional prediction model requires training from scratch, the proposed model enables training with less amount of time owing to the use of the pretrained model as a feature extractor. Results of experiments with the novel experimental data show that the proposed model succeeds in drastically reducing the training time while maintaining the same or even better prediction performance compared with the conventional method. Our approach presents the possibility of prediction on the basis of "greener" manner training of various peptide properties in proteomic liquid chromatography/tandem mass spectrometry experiments.

Proteins are important biological elements responsible for various functions of living organisms, and a systematic understanding of when, where, and how these proteins are expressed is necessary for system-wide analysis of biological functions[1]. Therefore, an important issue in proteomics is how to efficiently identify and quantify the vast number of proteins present in cells and tissues[2]. Recent advances in liquid chromatography/tandem mass spectrometry (LC/MS/MS) have significantly improved the coverage of bottom-up proteomics[3]. However, a typical human proteome sample consists of more than ten million protease-digested peptides[4], whose complexity is beyond the separation capabilities of current LC/MS/MS systems[5].

Recently, ion mobility spectrometry (IMS) has gained attention as yet another promising separation method to be combined with LC/MS/MS[6–10]. IMS separates molecules in terms of their charge and shape by measuring the mobility of ions moving in a buffer gas flow under the influence of an electric field[11]. The frequency of ion-gas collisions, also known as the collision cross section (CCS), determines the ion mobility in the gas phase[12]. Thus, even ion species of the same m/z may exhibit different CCS values due to different conformations they take[13]. The extended separation space provided by ion mobility resolves various problems caused by the insufficient separation of peptide ions in the conventional LC/MS/MS. For example, it can improve the separation of peptide isomers, which are peptides with the same sequence but different positions of post-translational modifications. Additionally, the improvement of peptide separation can lead to better quantitation accuracy[7,9,14].

IMS is not only effective for improving the separation efficiency of peptide ions, but it also has the potential for improving peptide identification[15,16]. While peptide identification primarily relies on MS/MS spectra, utilizing additional information such as peptide retention time can aid in the identification process, particularly for data from target acquisitions or data-independent acquisitions. However, accurate prediction of peptide retention time is necessary for approaches utilizing retention time information to be effective[17,18]. It is also the case with those utilizing IMS data: For better analysis of proteomic IMS data, it would be desirable if one can accurately predict CCS values of peptide ions. Several groups have so far proposed CCS prediction algorithms. Clemmer and coworkers established the model called intrinsic size parameter (ISP), which represents the relative size of each amino acid residue in a peptide sequence[13,19,20]. While this model works for a specific set of peptides, it has inherent limitations due to the use of the peptide's amino acid compositions but not the sequences. This ISP parameter has been further expanded to incorporate some sequence information[16].

On the other hand, inspired by great successes of deep learning in various research fields, the use of a deep neural network (NN) model for the

[1]Graduate School of Engineering, Nagoya Institute of Technology, Nagoya, Aichi, 466-8555, Japan. [2]Department of Molecular Systems Bioanalysis, Graduate School of Pharmaceutical Sciences, Kyoto University, Kyoto, 606-8501, Japan. [3]Laboratory of Proteomics for Drug Discovery, National Institute of Biomedical Innovation, Health and Nutrition, Ibaraki, Osaka, 567-0085, Japan. [4]Graduate School of Informatics, Kyoto University, Kyoto, 606-8501, Japan. [5]These authors jointly supervised this work: Yasushi Ishihama, Toshiyuki Tanaka. ✉e-mail: yishiham@pharm.kyoto-u.ac.jp; tt@i.kyoto-u.ac.jp

CCS prediction problem has been proposed in ref. 15, where a bidirectional LSTM model[21], trained from scratch with a dataset of 559,979 unique peptide ion data, was used for CCS prediction. The dataset contains peptide sequences of less than 30 amino acid residues, and mostly with the doubly charged species, which are typically observed in proteomic experiments. The prediction of CCS values of longer peptides is challenging due to the limited availability of data and their enhanced structural variability. Longer peptides tend to have higher charge states, such as triple, quadruple charges and more, which results in higher variability in CCS among species even within similar m/z ranges. Therefore, it is desirable to provide data consisting of a greater variety of peptides and verify the predictive performance of prediction models on such a dataset. In deep NN models, training time and computational load become bottlenecks for such performance verification. There is generally a trade-off between models' performance and such complexities. This problem is serious, especially in many laboratories where computational resources are limited. Developing models that can achieve reasonable performance at lower cost is a new direction to aim for.

Our proposal for CCS prediction is to use a pretrained deep protein language model as a feature extractor from peptide sequences, and train a separate NN, which we call a prediction NN, to predict CCS values on the basis of the extracted features. The overall model architecture of our proposal, which we name the pretrained protein language model-based network (PPLN), is depicted in Fig. 1. It has been argued[22] that a deep (natural) language model trained with a large corpus of a language implicitly acquires grammatical information of that language. Likewise, a pretrained deep protein language model trained with a large-scale database of protein sequences is believed to acquire structural information of proteins[23] (so called "protein grammar"). The possibility of utilizing pretrained protein language models for various prediction tasks was suggested in ref. 24. Indeed, it has been demonstrated in ref. 25 that the feature representation provided by a pretrained deep protein language model named evolutionary scale modeling-1b (ESM-1b) is useful in prediction of secondary structure and residue-residue contacts in proteins. As the CCS value of a peptide ion is regarded as being determined by the conformation of the ion particles in the drift tube, one can expect that the features obtained by such a deep protein language model that encodes structural information of proteins will be useful in prediction of CCS values of peptide ions as well.

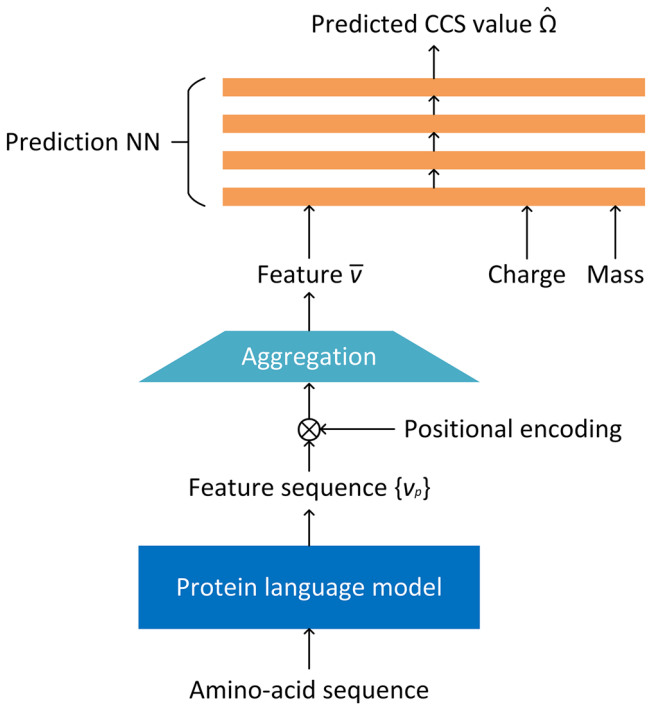

Fig. 1 | Model architecture of pretrained protein language model-based network (PPLN) for prediction of collision cross section (CCS) value. An amino-acid sequence is input into a pretrained protein language model. Positional encoding is applied to the obtained feature sequence. The feature sequence is aggregated to form a fixed-size feature, which is then input into the prediction neural network (NN) along with charge and mass.

Use of a pretrained protein language model as a feature extractor for CCS prediction might limit performance of CCS prediction compared with the approach of training a dedicated complex deep NN model from scratch, when the quality of the extracted features by the pretrained model is not good. It will have several advantages, on the other hand, when the quality of the features is good. It will make the CCS prediction problem easier to solve: One can use a simpler prediction NN, and train it with a smaller-sized training dataset and with less amount of time. Training of the prediction NN will thus be performed in a "greener" manner than training a dedicated complex NN model from scratch, on cheaper computer hardware and with less energy consumption.

In this study, we used a newly measured CCS dataset containing a wider variety of peptides to investigate how effective PPLN is compared with traditional deep learning models and how "green" the steps required to create a CCS predictor can be performed.

Results

Model architecture of PPLN

A number of deep protein language models for general purpose[25–31] and specific tasks[32,33] have so far been proposed. Such a deep protein language model, especially one for general purpose, can be used as a feature extractor, by removing the output layer of the model and regarding the outputs of the pre-output layer as a feature representation of the input. Although we used ESM-1b[25] as the feature extractor of PPLN in our experiments, any pretrained protein language model can in principle be used as the feature extractor. A feature extractor that is based on a deep protein language model typically takes as its input a variable-length amino-acid sequence, and outputs a sequence of features, whose length is the same as the length of the input sequence. One then has to aggregate the variable-length feature sequence into a fixed-size representation to feed it into the prediction NN, as shown in Fig. 1.

Although aggregation with simple averaging as used in the original ESM-1b would work well in certain tasks as in ref. 25, we found that introduction of aggregation considering amino acid positions, i.e., positional encoding (PE) in the aggregation, worked better than simple averaging. The fixed-size feature representation, along with the charge number and the mass of the ion, is then fed into the prediction NN, which outputs a prediction $\hat{\Omega}$ of the CCS value Ω of the input peptide ion. We included the charge number and the mass as the input of the prediction NN because the masses of peptides obviously have a strong correlation with the CCS values, and the charge numbers of peptide ions also have a strong influence on the CCS values. Inclusion of the charge number and the mass to the input of the prediction NN is thus expected to facilitate the learning of the prediction NN.

We next discuss our design of PE. The CCS value of a peptide ion can be affected by several factors. Among them, one can expect that amino acid residues located near the terminals have stronger effects than those located in the central part of the peptide, as suggested, e.g., by the length-specific multiple linear regression (LS-MLR) study[16]. We thus designed our PE in the aggregation stage in such a way that it reflects features of those residues located near either of the terminals of a peptide ion, rather than the absolute positions of the residues relative to the N-terminal, the latter of which we call the unidirectional PE in this paper.

The concrete mathematical explanations of feature extraction and PE, and the detailed settings of the prediction model PPLN in the following experiments are shown in "Methods".

Dataset construction

The emergence of proteomic IMS technologies and the availability of large-scale peptide CCS data have significantly improved the performance of peptide CCS prediction[15,16]. However, the current proteomic data mainly cover peptides with a length of less than 30 amino acids, which limits our

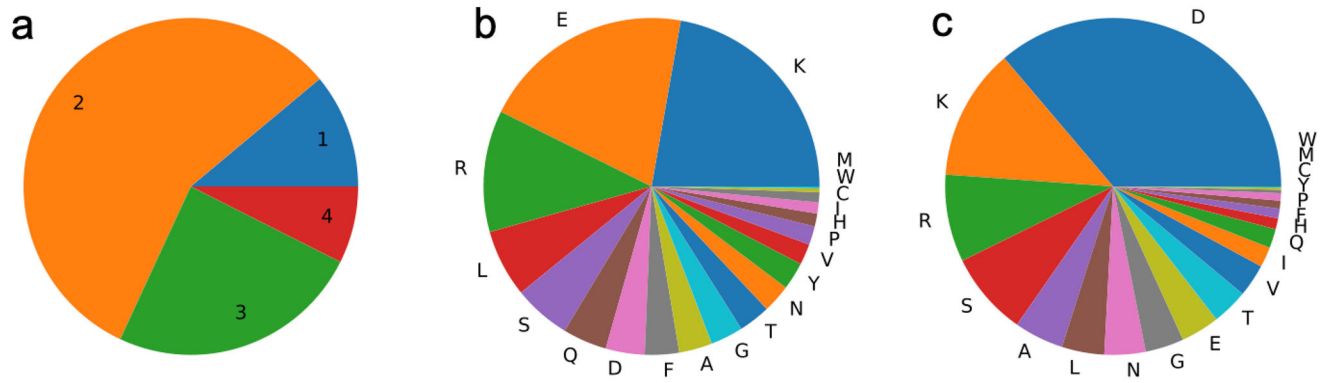

Fig. 2 | Summary statistics for the peptide dataset prepared from HeLa lysate using seven proteases. a Frequency of peptide charge numbers. **b** Frequency of peptide C-terminal amino acids. **c** Frequency of peptide N-terminal amino acids.

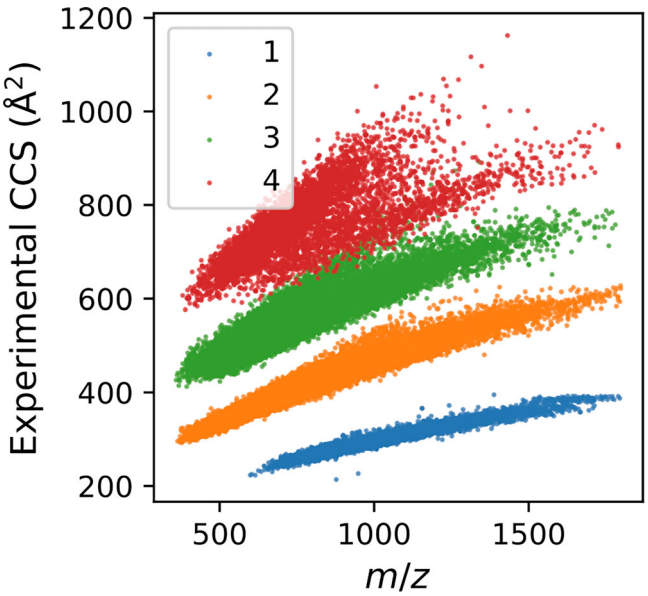

Fig. 3 | Distribution of 91,677 unique peptide ions. The dots 1, 2, 3, and 4 means the experimental CCS values of singly, doubly, triply, and quadruply charged ions vs. *m*/*z*, respectively.

understanding of the behaviors of longer peptides that are of general interest. To address this gap in knowledge and to make CCS data have a wide variety, we constructed an experimental peptide CCS dataset using phosphoproteome data. Phosphopeptides are known to have more missed cleavages and tend to be longer than non-phosphopeptides[34]. To obtain a unique set of peptides, we digested HeLa cell extracts with seven proteases (trypsin, LysargiNase, Lys-C, Lys-N, Glu-C, Asp-N, and chymotrypsin) as described previously[16], enriched phosphopeptides from the digests[35], and fractionated the resulting phosphopeptides with SCX-StageTip[36]. We then dephosphorylated the phosphopeptides using calf intestine alkaline phosphatase[34] to generate non-phosphorylated peptides with more missed cleavages. Mascot automated database search algorithm was used to identify the peptides, and IonQuant[37] was used to extract the peptide precursor ion features (mass, retention time, ion mobility, and intensity). We filtered out peptide ions bearing variable modifications and only considered the most abundant feature for each peptide ion. Phosphoproteomes contain distinct, longer sequences compared with conventional peptide CCS datasets, although they do not occupy the majority of our dataset (Supplementary Fig. 1).

The dataset consists of 91,677 unique peptide ion data. It includes 11% of singly charged, 57% of doubly charged, 25% of triply charged, and 7% of quadruply charged ions, which are shown in Fig. 2a. Frequencies of peptide C-terminal and N-terminal amino acids are also summarized in Fig. 2b, c, respectively. Figure 3 shows a plot of the experimental CCS values Ω vs. the *m*/*z* values. The experimental CCS values and *m*/*z* range from 289 to 1162 Å^2 and 381 to 1798, respectively.

Evaluation of prediction error from and correlation with experimental CCS values

For performance evaluations, the dataset mentioned in the previous section was randomly divided in two parts, where 73,342 ions (80% of the total) for training and the remaining 18,335 (20%) for testing. Figure 4 shows scatter plots of the predicted CCS values obtained by the proposed PPLN. Figure 4a includes results of predicted CCS values vs. experimental CCS values and four statistics, Pearson's correlation coefficient *r*, root mean squared error (RMSE), mean absolute error (MAE), and Δ95% error. The definitions are summarized in "Methods". This figure also includes the results of LS-MLR[16] and the bidirectional LSTM-based method[15]. The LS-MLR model was constructed to be fitted on all data of the dataset. The deep NN used in the bidirectional LSTM-based method was trained from scratch using the training dataset of this paper, rather than the pretrained model using the dataset provided in ref. 15. Prediction error evolutions for the bidirectional LSTM-based method and PPLN are shown in Supplementary Figs. 3 and 4.

From Fig. 4, the predicted CCS values by LS-MLR tended to overestimate the experimental CCS values, especially for ions with larger CCS values. It is worth noting that the distribution of larger CCS peptides was split into two populations: One with overestimated CCS values predicted by LS-MLR and the other with fairly predicted CCS values. This can be mostly explained by the length of the peptides: The peptides with overestimated CCS were longer than the other ones (Supplementary Fig. 2). On the other hand, the proposed PPLN provided better predictive performance with the lower RMSE and the higher correlation coefficient. Figure 4b, c shows the relation to *m*/*z* and length of the ions. We can see that scatter plots of LS-MLR and the others differ, especially in higher *m*/*z* and longer-length regions. These results indicate that the CCS prediction of longer peptide ions with higher *m*/*z* is difficult with simpler prediction models. To establish a better CCS prediction model, improving the prediction accuracy of these ions is mandatory.

The predicted CCS values and the four statistics are summarized by charge number in Fig. 5. From the figure, we can see that the predictions for triply and quadruply charged ions were more difficult than the singly and doubly charged ions because the performance of all the three methods compared became worse as the charge number was higher. We can also see in Fig. 5d that the peptides with CCS values overestimated by LS-MLR are typically quadruply charged. This is consistent with the fact that longer peptides tend to have higher charge states. The proposed PPLN achieved the best performance among the methods in all the cases. The results imply the high applicability of the proposed method.

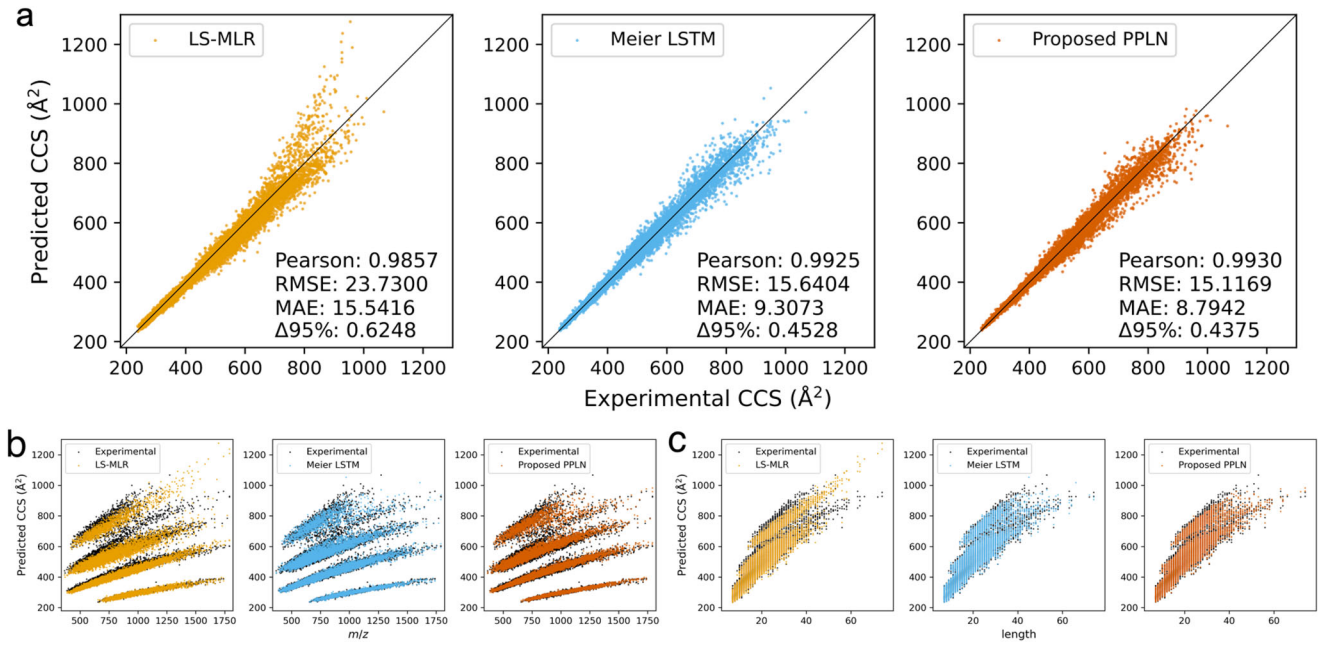

Fig. 4 | Scatter diagrams and statistics in terms of predicted CCS values. a Predicted CCS values vs. experimental CCS values including Pearson's correlation coefficient, RMSE, MAE, and Δ95% error. **b** Predicted CCS values vs. *m/z*. **c** Predicted CCS values vs. length.

It should be noted that the proposed PPLN does not always achieve better predictive performance than the conventional LSTM-based method, although it performs better in the results of Fig. 4 using the training/test data splitting in this section. Supplementary Table 1 summarizes the average performance metrics of 10 trials including other random splittings of the 80% training/20% test data and the corresponding results of statistical testing (paired *t*-test, $p = 0.05$). The results show that the performance of the PPLN is better or worse depending on the performance metrics. Therefore, it can be concluded that the proposed PPLN achieves predictions that are competitive with the LSTM-based method. This is consistent with the original goal of the paper to leverage a pretrained deep protein language model to achieve reasonable performance.

We further analyzed and visualized the contributions of amino acid positions using Shapley additive explanation (SHAP) values to make the relation between input sequences and prediction of the proposed PPLN more interpretable. The results are shown in Supplementary Figs. 5 and 6.

Ablation study

In this section, we verify necessity of the components of the proposed PPLN via ablation study, where we compared the predictive performance of the proposed method with the same method except that a part of the components was removed from it, in order to see if the removed part was important. Table 1 summarizes RMSE and Pearson's correlation coefficient *r* obtained by the proposed method and methods with the removal.

We first validate the necessity of including the charge number and the mass into the input of the prediction NN in the proposed PPLN. The performance of the method that does not use charge numbers of ions was much worse than the proposed method, indicating that charge number bears important information for CCS value prediction. It is in contrast with the conventional CCS prediction method[15] which does not use the charge number information. The method which excludes mass and that which excludes both charge and mass also showed worse performance. From the results, we can argue that the information of charge and mass of the ions is valuable for the CCS prediction.

We also validate the necessity of the bidirectional PE by comparing the predictive performance of the following four methods: the method with the bidirectional PE (i.e., the proposed method), the one with the unidirectional PE, the one with the bidirectional aggregation without PE (i.e., setting $\boldsymbol{c}_p$ in Eq. (4) to be equal to the all-1 vector), and the one adopting aggregation with simple averaging (i.e., the original ESM-1b[25], setting $\boldsymbol{c}_p$ in Eq. (5) to be equal to the all-1 vector). From Table 1, none of the compared methods showed better performance than the proposed method. The disuse of the bidirectional aggregation especially degraded the performance. This implies that the N-terminal and C-terminal sides of the ions have different effects on peptide structures and on the resulting CCS values, and those effects can be successfully learned by the network with the bidirectional aggregation.

Comparison of execution time for training

The proposed PPLN can simplify the training process by using the pretrained model as the feature extractor, compared with the conventional bidirectional LSTM-based method[15] that needs training from scratch. We numerically evaluated the execution time required for training of the proposed PPLN and the conventional bidirectional LSTM-based method.

Figure 6 shows time (in seconds) spent on training for each method and that required for preprocessing, i.e., feature extraction, for the proposed PPLN. We used a Linux computer with two CPUs (Intel Xeon Gold 5320, 26 cores, 2.2 GHz base clock) and 256 GB RAM. The preprocessing time means the time taken to obtain the features for all peptides in the dataset. The training time was measured when 20%, 50%, and 80% of peptides in the dataset were used for training (the numbers of training samples were 18,335, 45,839, and 73,342, respectively). The average time of three runs is shown in the figure. All measured values were within plus or minus 120 s of the shown average values. For the bidirectional LSTM-based model, the results were obtained using the recommended configuration provided in a GitHub repository tied to the original paper[15], where the number of training iterations is fixed to 55,000 regardless of the number of training samples. The results of comparison when the total number of the training data used is the same are shown in Supplementary Fig. 7.

From Fig. 6, the training of the proposed PPLN not including the preprocessing was executed in 1/78, 1/30, and 1/18 of the time of the training of the conventional bidirectional LSTM-based method, when 20%, 50%, and 80% samples were used for training, respectively. Even taking the preprocessing time into consideration, execution time for the proposed PPLN was reduced to 1/4 to 1/3 of that of the conventional method. Moreover, it should be noted that the proposed PPLN achieved test prediction with Pearson's correlation coefficient over 0.99 for all runs. CPU

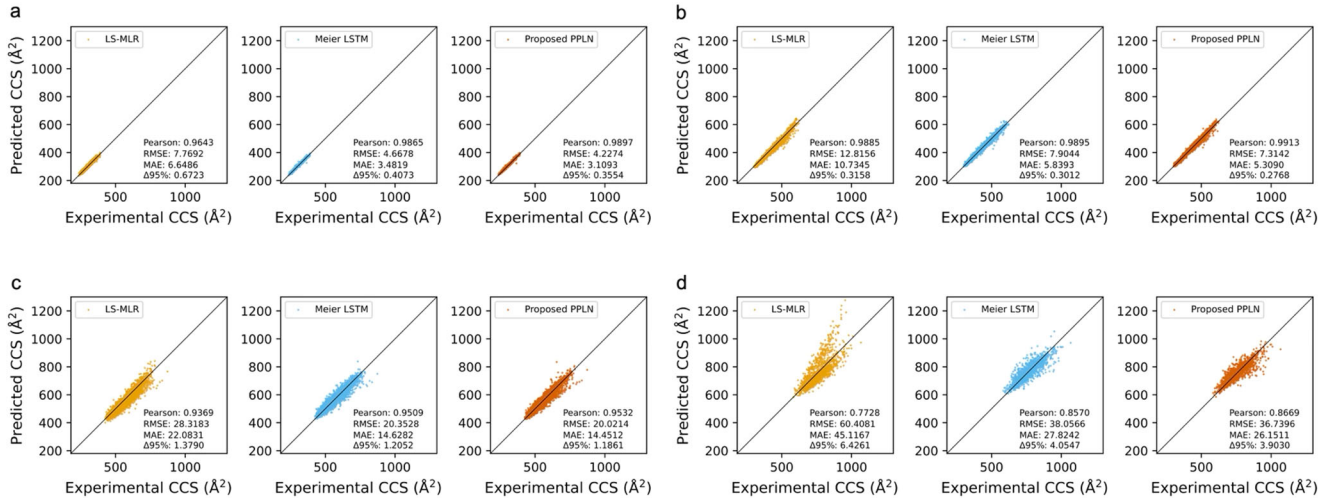

Fig. 5 | Predicted CCS values and statistics by charge. a Scatter diagram for singly charged ions including Pearson's correlation coefficient, RMSE, MAE, and Δ95% error. **b** For doubly charged ions. **c** For triply charged ions. **d** For quadruply charged ions.

usage during the training process was 1057% and 176.1% for the bidirectional LSTM-based method and the proposed PPLN, respectively. These were measured per 10 s and averaged over 10 training trials. One can therefore state that the proposed model is also "greener" in terms of energy usage.

The execution time for prediction was also measured in terms of 80% training data case. The average prediction time over three trials was 44.0 s for the bidirectional LSTM-based method and 0.5 s for the proposed PPLN, respectively. In other words, in this training and test data splitting, the proposed method could also perform prediction in a shorter time than the conventional method, owing to the simple model structure of the prediction NN.

These results indicate that the proposed PPLN enables a significant reduction in training and test time by virtue of the simplification of backward and forward calculation for training and test, respectively, through the use of the pretrained model, while maintaining high predictive performance and lower energy usage.

CCS prediction for improving peptide identification

In this section, we show that accurate CCS prediction provided by the proposed PPLN allows us to improve performance of downstream tasks. We take the peptide identification task as our demonstrative example, as it has been shown that it is possible to reduce the false hits using the difference between the predicted and experimental CCS values after the candidate sequences are determined by the search engine[16,38,39]. We used tryptic peptides from *E. coli* K12 strain BW25113 cells to verify the improvement in the identification number using PPLN, as reported previously[16]. The additional parameter, CCS error, defined as the difference between the predicted and experimental values divided by the experimental value, was used for the Mascot/Percolator approach[40]. The CCS predictions were performed by both the LSTM-based method and the PPLN trained with the HeLa dephosphorylated dataset. Figure 7 shows the Venn diagram of the identification results with or without CCS error. From the figure, Percolator with CCS error identified more peptide spectrum matches (PSMs) and stripped sequences at 1% FDR compared with Percolator without CCS error, indicating the utility of PPLN-based CCS prediction for peptide identification in proteomics.

Table 1 | Summary of ablation study

Status	RMSE	Pearson *r*
w/o charge	31.8161	0.9721
w/o mass	16.7563	0.9914
w/o charge, mass	38.7438	0.9533
unidirectional PE	15.6560	0.9927
bidirectional aggregation w/o PE	15.9156	0.9929
aggregation with simple averaging (original ESM-1b)	16.3448	0.9923
Proposed PPLN	**15.1169**	**0.9930**

The bold values mean the best of the methods.

Discussion

In this work, we proposed a novel approach to predict the CCS values of peptides using a deep learning model called the PPLN. The proposed PPLN incorporates a pretrained deep protein language model trained with a large-scale database of protein sequences as a feature extractor to perform the training process in a "greener" manner. Our results demonstrated that the proposed PPLN can achieve a more accurate prediction of CCS values compared with the conventional LS-MLR and competitive performance with the bidirectional LSTM-based approach. A remarkable point is that the training with the PPLN was made in a significantly shorter time than with the conventional bidirectional LSTM-based method requiring training from scratch. The LSTM-based method includes 0.4M trainable parameters and the PPLN with the architecture used in the experiments of the paper includes 650M frozen and 11M trainable parameters. The PPLN includes a larger number of weights to be optimized, but the computational load for forward path and backpropagation is larger for LSTM. That is the reason for the difference in execution times of both models.

PPLN can also achieve reasonable performance in predicting CCS values of longer peptides, which is difficult with simple prediction models such as LS-MLR. This can be attributed to the use of the pretrained protein language model, such as ESM-1b, in PPLN: A pretrained protein language model can capture the complex relationships between amino acid sequences and protein structures more effectively than traditional methods, which might contribute to the better prediction performance of PPLN because the peptide CCS reflects the peptide structure, which is the part of the protein structure. The difficulty in predicting triply and quadruple charged peptide ions is due to the variety of their structures and the small number of them in the training data. The latter problem corresponds to imbalanced data in machine learning[41], which causes performance degradation for data with minor attributes, in our case for highly charged ions. By constructing a dataset rich in highly charged ions, one can expect to significantly improve the performance of prediction models.

Our study demonstrates the potential effectiveness of utilizing pretrained protein language models in predicting various peptide properties in proteomic LC/MS experiments, including not only CCS but also peptide

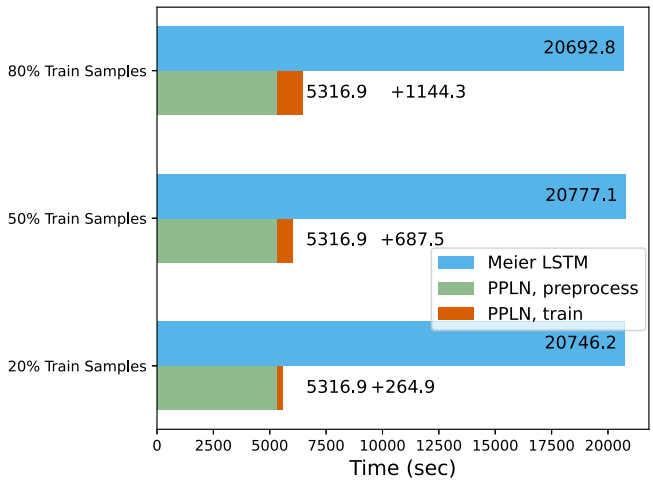

Fig. 6 | Execution time (in seconds) for the proposed PPLN and the conventional bidirectional LSTM-based method in the cases of using 20%, 50%, and 80% samples for training. For the PPLN, the training time for prediction NN and execution time for preprocessing required for obtaining features via ESM-1b are shown.

retention time, MS/MS fragmentation pattern, and detectability. In addition to the improved prediction performance, utilizing a pretrained protein language model offers several advantages, such as requiring less training data for accurate predictions, leading to lower computational resource requirements and reducing the time and energy consumption. By utilizing these advantages, transfer learning approaches allow the model to be readily applicable to the prediction of CCS values from different experimental setup such as the use of the ion mobility spectrometer with different modes of separation. Although the PPLN cannot directly incorporate modified residues resulting from chemical and post-translational modifications—due to its reliance on the protein language model, which does not account for these modifications—it is possible to extend its applicability to modified peptides by considering modification tokens (the modification position with the CCS shift), as reported recently[42].

It has been suggested that model size contributes to downstream task performance[27,43] so that the introduction of protein language models with a huge number of parameters, such as ESM-2[27] and xTrimoPGLM[31] instead of ESM-1b used in our PPLN is expected to offer even better prediction accuracy. On the other hand, the magnitude of the pretrained model influences the execution time of preprocessing. The use of smaller models directly leads to time savings in the training and test processes of the proposed model. The preprocessing time using ESM-1b was 5316.9/91,677 = 0.058 s per peptide ion. The amount of preprocessing time may be a limitation when the number of peptide ions to be predicted is large. One of the prompt solutions to reduce the prediction time is to replace it with a smaller protein language model. For example, ESM-2[27] is a model with a minimum of 8M and a maximum of 15B parameters. It would be interesting to verify prediction accuracy and execution time using a larger or smaller protein language model as the feature extractor in future work. Moreover, the current model is not applicable to variable modifications. The exploration of applicable models is one of the important future directions.

The CCS dataset provided in this paper contains longer and more highly charged peptide ions than that in ref. 15. It allows the training of the model on a wider variety of peptide ions. Note that the architecture of the proposed PPLN is not specialized for this dataset. The PPLN also achieves reasonable performance on another dataset, which is shown in Supplementary Fig. 8. In other words, the proposed PPLN is robust to data of different nature while enjoying the benefits of "greener" training processes.

In summary, our approach represents a significant advance in the prediction of peptide CCS values. By leveraging pretrained protein language models, we have shown that it is possible to accurately predict CCS values for longer peptides with short training times. We believe that our approach can be extended to predict other peptide properties and that the use of pretrained models will become increasingly important for efficient and accurate peptide property prediction.

Methods

Materials

Titanium dioxide (TiO_2) beads were obtained from GL Sciences (Tokyo, Japan). 2-amino-2-(hydroxymethyl)-1,3-propanediol hydrochloride (Tris-HCl), acetonitrile, acetic acid, ammonium bicarbonate (ABC), trifluoroacetic acid, lysyl endopeptidase (Lys-C), V8 protease (Glu-C), and other chemicals were purchased from Fujifilm Wako (Osaka, Japan). Modified trypsin and chymotrypsin were purchased from Promega (Madison, WI, USA). Asp-N was purchased from Roche diagnostics (Indianapolis, IN, USA). Lys-N was purchased from Thermo Fisher Scientific (Waltham, MA, USA). LysargiNase was purchased from Merck (Darmstadt, Germany). Alkaline phosphatase was purchased from Takata Bio Inc. (Shiga, Japan). Empore C8, Empore SDB-XC (polystyrenedivinylbenzene) and Empore SCX (strong cation exchange) extraction disks were purchased from CDS Analytical (Oxford, PA, USA). Water was purified by a Millipore Milli-Q system (Bedford, MA, USA).

Sample preparation

The HeLa S3 cell line (Japan Health Sciences Foundation) was cultured in 10 cm diameter dishes following standard protocols. The cells were collected, and pelleted down by centrifugation. The cell pellets were suspended in a lysis buffer, reduced and alkylated as previously reported[44]. The samples were diluted five times with 50 mM ABC buffer or ten times with 10 mM $CaCl_2$ in the case of LysargiNase digestion. The digestion was performed at 37 °C by adding trypsin, Lys-C, Lys-N, Asp-N, LysargiNase, chymotrypsin or Glu-C. The appropriate enzyme-to-protein ratios were used for each enzyme (Supplementary Table 2). The resulting peptides were desalted and purified via SDB-XC StageTip according to the previously published protocol[45,46]. The desalted peptides were further processed with C8 StageTips packed with TiO_2 to enrich phosphopeptides as previously reported[35]. Phosphopeptides were eluted with 0.5% piperidine followed by 5% pyrrolidine, acidified immediately by adding equal volume of 20% phosphoric acid (final concentration: 10%), and desalted using SDB-XC StageTips[47]. Enriched phosphopeptides were fractionated using SCX StageTips as described previously[36], followed by dephosphorylation with 6 units of alkaline phosphatase in 30 µL of 100 mM Tris-HCl buffer (pH 9.0), incubated for 3 h at 37 °C. After the reaction, the buffer was acidified by adding 10% TFA 10 µL. The samples were desalted using SDB-XC StageTips.

LC/IMS/MS/MS analysis

LC/IMS/MS/MS was performed on an Ultimate 3000 RSLCnano (Thermo Fisher Scientific, Waltham, MA, USA) LC pump coupled with a PAL HTC-xt (CTC analytics, Zwingen, Switzerland) autosampler and a timsTOF Pro (Bruker Daltonics, Bremen, Germany) mass spectrometer. Peptides were separated on a 15 cm × 100 µm in-house-packed with 1.9 µm Reprosil-Pur AQ C18 beads (Dr. Maisch, Ammerbuch, Germany) column at a flow rate of 500 nL/min with an PRSO-V2 (Sonation, Biberach, Germany) column oven heated at 50 °C. Mobile phases A and B were water and 20%/80% water/acetonitrile (v/v), respectively, both with 0.5% acetic acid as an ion-pair reagent[48]. A total run time was 120 min with gradient starting with a linear increase from 5% B to 40% B over 90 min followed by linear increases to 99% B in 1 min. The mass spectrometer was operated in data-dependent PASEF[9] mode, with 1 survey TIMS-MS followed by 10 PASEF MS/MS scans per acquisition cycle. An ion mobility scan range from $1/K_0 = 0.6$ to 1.5 Vs/cm^2 was employed with 100 ms ion accumulation and ramp time. Precursor ions for MS/MS analysis were selected and isolated in a window of 2 m/z for $m/z < 700$ and 3 m/z for $m/z > 700$. Singly charged ions were excluded from the precursor ions according to their m/z and $1/K_0$ values. The TIMS elution voltage was calibrated linearly to obtain the reciprocal of reduced ion mobility ($1/K_0$) using three selected ions ($m/z = 622$, 922, and 1222) of the ESI-L Tuning Mix (Agilent, Santa Clara, CA, USA)

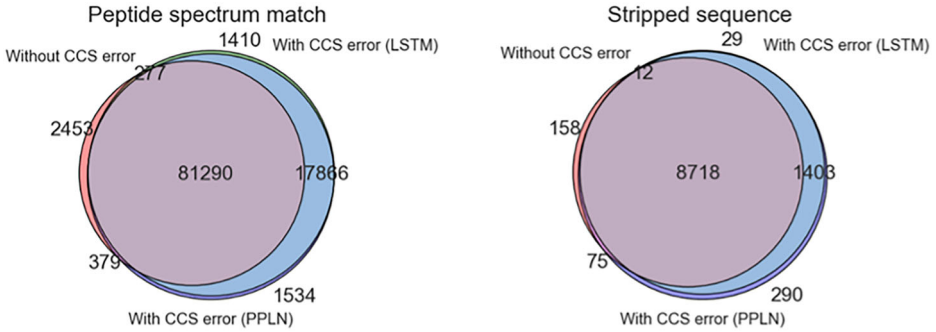

Fig. 7 | *E. coli* peptide identification results with or without CCS error. CCS predictions were performed with the conventional LSTM-based method and the proposed PPLN trained with the HeLa dephosphorylated dataset.

Database search and data processing

MS raw files were first processed by Bruker DataAnalysis software to generate mgf files. Database search was performed with Mascot version 2.7.0 against Swissprot (downloaded on 20200318) human database containing isoforms using the appropriate digestion rules for each protease (Supplementary Table 1). Carbamidomethyl (C) was set as a fixed modification, and Phospho (STY), Oxidation (M) and Acetyl (Protein N-term) were set as variable modifications. The peptide tolerance of 20 ppm and MS/MS tolerance of 0.05 Da were used. Number of ^{13}C was set as 1 to consider monoisotopic + 1 peaks as precursor ions. Percolator[40] was used for controlling the false discovery rates (FDRs) at 1% on both the PSM and unique peptide level in terms of q values. The identified PSMs were further processed with IonQuant[37] (version 1.3.6) to re-assign precursor ions. The PSMs which could not be assigned to any precursor ions with IonQuant were removed for further analysis. Furthermore, the reduced ion mobility of peptide ions was calibrated linearly using three selected ions (352.33, 761.467, 933.919) which were constitutively detected in all of the raw data, to minimize the run-to-run variability of the obtained $1/K_0$ values. The CCS value Ω was calculated from the obtained value of $1/K_0$ using the Mason-Schamp equation[49]:

$$\Omega = \frac{3Ze}{16N_0}\left(\frac{2\pi}{\mu k_B T}\right)^{1/2}\frac{1}{K_0}, \tag{1}$$

where e is the charge on an electron, where Z is the charge number of the analyte ion, where k_B is the Boltzmann constant, where N_0 is Loschmidt constant, where T is the temperature, and where μ is the reduced mass of the ion and neutral given by the harmonic mean of the molecular masses of the drift gas and analyte ion, respectively. Peptide ions without any variable modifications were used for analysis. We only considered the most abundant feature for each modified peptide ion.

Mathematical explanation of feature extraction in PPLN

Assume that we use a feature extractor which, when fed with a length-l peptide ion sequence, outputs a length-l sequence $\{\boldsymbol{v}_p\}_{p=1}^{l}$ of d-dimensional feature vectors. We then propose the following PE, which we term the bidirectional PE: the feature vector sequence $\{\boldsymbol{v}_p\}_{p=1}^{l}$ is aggregated into a single $2d$-dimensional vector $\bar{\boldsymbol{v}}$ via:

$$\bar{\boldsymbol{v}} = \begin{pmatrix}\bar{\boldsymbol{v}}^{\mathrm{N}}\\ \bar{\boldsymbol{v}}^{\mathrm{C}}\end{pmatrix}, \tag{2}$$

$$\bar{\boldsymbol{v}}^{\mathrm{N}} = \frac{1}{l/2}\sum_{p=1}^{l/2}\boldsymbol{c}_p \circ \boldsymbol{v}_p, \tag{3}$$

$$\bar{\boldsymbol{v}}^{\mathrm{C}} = \frac{1}{l/2}\sum_{p=1}^{l/2}\boldsymbol{c}_p \circ \boldsymbol{v}_{l+1-p}, \tag{4}$$

where $\circ$ denotes the element-wise (Hadamard) product of vectors, and where $\boldsymbol{c}_p = (c_{p,1}, \ldots, c_{p,d})^{\top}$ is the encoding vector for position p from one of the terminals. That is, $\bar{\boldsymbol{v}}^{\mathrm{N}}$ summarizes the features from the half of the amino acid sequence on the N-terminal side, and $\bar{\boldsymbol{v}}^{\mathrm{C}}$ summarizes the features from the other half of the amino acid sequence on the C-terminal side. (When l is odd, one may have to introduce an appropriate rounding of the half-integer $l/2$. In our implementation used in the experiments, we used the python built-in function `round()`.) These vectors are concatenated to form the resulting feature vector $\bar{\boldsymbol{v}}$. One can argue that the bidirectional PE bears the same spirit as that in ref. 15 where they used the bidirectional LSTM rather than the (unidirectional) LSTM to deal with peptide sequences. As mentioned above, an alternative choice to the bidirectional PE might be the unidirectional PE, where the d-dimensional aggregated vector $\bar{\boldsymbol{v}}_{\mathrm{u}}$ is obtained via:

$$\bar{\boldsymbol{v}}_{\mathrm{u}} = \frac{1}{l}\sum_{p=1}^{l}\boldsymbol{c}_p \circ \boldsymbol{v}_p. \tag{5}$$

Experimental model settings based on pretrained deep protein language model

We used the pretrained ESM-1b model[25] as the feature extractor in PPLN, whose output is a sequence of $d = 1280$-dimensional feature vectors. As for the encoding vectors $\boldsymbol{c}_p$ used in PE, we considered the following functional form:

$$c_{p,i} = \begin{cases}\left(\sin\frac{p}{a^{(i-1)/d}}\right)^{b} + \gamma, & i:\ \text{odd},\\ \left(\cos\frac{p}{a^{(i-2)/d}}\right)^{b} + \gamma, & i:\ \text{even},\end{cases} \quad (p = 1, 2, \ldots, l, i = 1, 2, \ldots, d) \tag{6}$$

with user-tunable parameters (a, b, γ). This formulation is inspired by the positional encoding used in the attention mechanism of the transformer architecture for a deep natural language model[50]. We used $(a, b, \gamma) = (1000, 1, 0)$ in the following experiments.

We used PyTorch and adopted minibatch learning with batch size 200. The number of layers of the prediction NN was set to 10 and their dimensions were 1000 except for the last layer that outputs the scalar CCS value.

It is acceptable to set parameters (a, b, γ) of PE and the architecture of the prediction NN appropriately according to the nature of dataset or computer resource environments. Performance using other architectures is shown in Supplementary Table 3.

We trained the prediction NN with Adam optimizer (learning rate: 0.0003) and the MSE loss function using the training data over 400 epochs, and tested the CCS prediction performance of the trained model on the test data.

Performance metrics

The definitions of Pearson's correlation coefficient r, RMSE, and MAE are given by:

$$r = \frac{\sum_{n=1}^{N}\left(\hat{\Omega}_n - \bar{\hat{\Omega}}\right)\left(\Omega_n - \bar{\Omega}\right)}{\sqrt{\sum_{n=1}^{N}\left(\hat{\Omega}_n - \bar{\hat{\Omega}}\right)^2 \sum_{n=1}^{N}\left(\Omega_n - \bar{\Omega}\right)^2}}, \tag{7}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(\hat{\Omega}_n - \Omega_n\right)^2}, \tag{8}$$

and

$$\mathrm{MAE} = \frac{1}{N}\sum_{n=1}^{N}|\hat{\Omega}_n - \Omega_n|, \tag{9}$$

respectively, where $N = 18{,}335$ is the number of test samples, where Ω_n and $\hat{\Omega}_n$ are the experimental and predicted CCS values, respectively, of nth test sample, and where $\bar{\Omega} := (1/N)\sum_{n=1}^{N}\Omega_n$ and $\bar{\hat{\Omega}} := (1/N)\sum_{n=1}^{N}\hat{\Omega}_n$ are the averages of the experimental and predicted CCS values. Δ95% error is the interval that contains 95% of the peptides in the error distribution.

Data availability

The MS raw data and analysis files have been deposited with the ProteomeXchange Consortium (http://proteomecentral.proteomexchange.org) via the jPOST partner repository[51] (https://jpostdb.org) with the data set identifier PXD046201.

Code availability

The source code for CCS value prediction with the proposed PPLN is available on GitHub (https://github.com/a-nakai-k/PPLN).

Received: 25 September 2024; Accepted: 25 April 2025;
Published online: 06 May 2025

References

1. Aebersold, R. & Mann, M. Mass spectrometry-based proteomics. *Nature* **422**, 198–207 (2003).
2. Shuken, S. R. An introduction to mass spectrometry-based proteomics. *J. Proteome Res.* **22**, 2151–2171 (2023).
3. Sinitcyn, P. et al. Global detection of human variants and isoforms by deep proteome sequencing. *Nat. Biotechnol.* **41**, 1776–1786 (2023).
4. Choong, W. K., Chen, C. T., Wang, J. H. & Sung, T. Y. *i*HPDM: in silico human proteome digestion map with proteolytic peptide analysis and graphical visualizations. *J. Proteome Res.* **18**, 4124–4132 (2019).
5. Shishkova, E., Hebert, A. S. & Coon, J. J. Now, more than ever, proteomics needs better chromatography. *Cell Syst.* **3**, 321–324 (2016).
6. Myung, S. et al. Development of high-sensitivity ion trap ion mobility spectrometry time-of-flight techniques: a high-throughput nano-LC-IMS-TOF separation of peptides arising from a Drosophila protein extract. *Anal. Chem.* **75**, 5137–5145 (2003).
7. Bonneil, E., Pfammatter, S. & Thibault, P. Enhancement of mass spectrometry performance for proteomic analyses using high-field asymmetric waveform ion mobility spectrometry (FAIMS). *J. Mass Spectrom.* **50**, 1181–1195 (2015).
8. Bekker-Jensen, D. B. et al. A compact quadrupole-orbitrap mass spectrometer with FAIMS interface improves proteome coverage in short LC gradients. *Mol. Cell. Proteom.* **19**, 716–729 (2020).
9. Meier, F. et al. Parallel accumulation-serial fragmentation (PASEF): multiplying sequencing speed and sensitivity by synchronized scans in a trapped ion mobility device. *J. Proteome Res.* **14**, 5378–5387 (2015).
10. Meier, F. et al. Online parallel accumulation-serial fragmentation (PASEF) with a novel trapped ion mobility mass spectrometer. *Mol. Cell. Proteom.* **17**, 2534–2545 (2018).
11. May, J. C. & McLean, J. A. Ion mobility-mass spectrometry: time-dispersive instrumentation. *Anal. Chem.* **87**, 1422–1436 (2015).
12. Gabelica, V. & Marklund, E. Fundamentals of ion mobility spectrometry. *Curr. Opin. Chem. Biol.* **42**, 51–59 (2018).
13. Valentine, S. J., Counterman, A. E. & Clemmer, D. E. A database of 660 peptide ion cross sections: use of intrinsic size parameters for bona fide predictions of cross sections. *J. Am. Soc. Mass Spectrom.* **10**, 1188–1211 (1999).
14. Ogata, K. & Ishihama, Y. Extending the separation space with trapped ion mobility spectrometry improves the accuracy of isobaric tag-based quantitation in proteomic LC/MS/MS. *Anal. Chem.* **92**, 8037–8040 (2020).
15. Meier, F. et al. Deep learning the collisional cross sections of the peptide universe from a million experimental values. *Nat. Commun.* **12**, 1185 (2021).
16. Chang, C. H. et al. Sequence-specific model for predicting peptide collision cross section values in proteomic ion mobility spectrometry. *J. Proteome Res.* **20**, 3600–3610 (2021).
17. Krokhin, O. V. et al. Use of peptide retention time prediction for protein identification by off-line reversed-phase HPLC-MALDI MS/MS. *Anal. Chem.* **78**, 6265–6269 (2006).
18. Ogata, K., Krokhin, O. V. & Ishihama, Y. Retention order reversal of phosphorylated and unphosphorylated peptides in reversed-phase LC/MS. *Anal. Sci.* **34**, 1037–1041 (2018).
19. Hilderbrand, A. E. & Clemmer, D. E. Determination of sequence-specific intrinsic size parameters from cross sections for 162 tripeptides. *J. Phys. Chem. B* **109**, 11802–11809 (2005).
20. Kaszycki, J. L. & Shvartsburg, A. A. A priori intrinsic PTM size parameters for predicting the ion mobilities of modified peptides. *J. Am. Soc. Mass Spectrom.* **28**, 294–302 (2016).
21. Duch, W. *Artificial Neural Networks: Formal Models and Their Applications-ICANN 2005: 15th International Conference, Warsaw, Poland, September 11-15, 2005, Proceedings* (Springer Science & Business Media, 2005).
22. Manning, C. D., Clark, K., Hewitt, J., Khandelwal, U. & Levy, O. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proc. Natl Acad. Sci. USA* **117**, 30046–30054 (2020).
23. Rao, R., Meier, J., Sercu, T., Ovchinnikov, S. & Rives, A. Transformer protein language models are unsupervised structure learners. In *International Conference on Learning Representations* https://openreview.net/forum?id=fylclEqgvgd (2021).
24. Dens, C., Adams, C., Laukens, K. & Bittremiuex, W. Machine learning strategies to tackle data challenges in mass spectrometry-based proteomics. *J. Am. Soc. Mass Spectrom.* **35**, 2143–2155 (2024).
25. 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).
26. Rao, R. et al. MSA transformer. In *Proceedings of the 38th International Conference on Machine Learning* **139** (eds. Meila, M. & Zhang, T.) Vol. 139, 8844–8856 (PMLR, 2021).
27. Lin, Z. et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* **379**, 1123–1130 (2023).
28. 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).
29. Rao, R. et al. Evaluating protein transfer learning with TAPE. In *Advances in Neural Information Processing Systems* **32** (eds. Wallach, H. et al.), 9689–9701 (Curran Associates, Inc., 2019).
30. Elnaggar, A. et al. Prottrans: toward understanding the language of life through self-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 7112–7127 (2021).

31. Chen, B. et al. xTrimoPGLM: unified 100B-scale pre-trained transformer for deciphering the language of protein. *Nat. Methods* https://doi.org/10.1038/s41592-025-02636-z (2025).
32. Meier, J. et al. Language models enable zero-shot prediction of the effects of mutations on protein function. In *Advances in Neural Information Processing Systems* **34** (eds. Ranzato, M. et al.) 29287–29303 (Curran Associates, Inc., 2021).
33. Madani, A. et al. Large language models generate functional protein sequences across diverse families. *Nat. Biotechnol.* **41**, 1099–1106 (2023).
34. Ogata, K., Chang, C.-H. & Ishihama, Y. Effect of phosphorylation on the collisioncross sections of peptide ions in ion mobility spectrometry. *Mass Spectrom.* **10**, A0093 (2021).
35. Sugiyama, N. et al. Phosphopeptide enrichment by aliphatic hydroxy acid-modified metal oxide chromatography for nano-LC-MS/MS in proteomics applications. *Mol. Cell. Proteom.* **6**, 1103–1109 (2007).
36. Adachi, J. et al. Improved proteome and phosphoproteome analysis on a cation exchanger by a combined acid and salt gradient. *Anal. Chem.* **88**, 7899–7903 (2016).
37. Yu, F. et al. Fast quantitative analysis of timsTOF PASEF data with MSFragger and IonQuant. *Mol. Cell. Proteom.* **19**, 1575–1585 (2020).
38. Teschner, D. et al. Ionmob: a Python package for prediction of peptide collisional cross-section values. *Bioinformatics* **39**, btad486 (2023).
39. Declercq, A. et al. TIMS^{2}Rescore: a data dependent acquisition-parallel accumulation and serial fragmentation-optimized data-driven rescoring pipeline based on MS^{2}Rescore. *J. Proteome Res.* **24**, 1067–1076 (2025).
40. Käll, L., Canterbury, J. D., Weston, J., Noble, W. S. & MacCoss, M. J. Semi-supervised learning for peptide identification from shotgun proteomics datasets. *Nat. Methods* **4**, 923–925 (2007).
41. Kaur, H., Pannu, H. S. & Malhi, A. K. A systematic review on imbalanced data challenges in machine learning: applications and solutions. *ACM Comput. Surv.* **52**, 1–36 (2019).
42. Peng, F. Z. et al. PTM-Mamba: a PTM-aware protein language model with bidirectional gated Mamba blocks. *Nat. Methods* Online ahead of print (2025).
43. Hesslow, D., Zanichelli, N., Notin, P., Poli, I. & Marks, D. RITA: a study on scaling up generative protein sequence models. In The 2022 ICML Workshop on Computational Biology (2022).
44. Masuda, T., Tomita, M. & Ishihama, Y. Phase transfer surfactant-aided trypsin digestion for membrane proteome analysis. *J. Proteome Res.* **7**, 731–740 (2008).
45. Rappsilber, J., Mann, M. & Ishihama, Y. Protocol for micro-purification, enrichment, pre-fractionation and storage of peptides for proteomics using StageTips. *Nat. Protoc.* **2**, 1896–1906 (2007).
46. Ogata, K. & Ishihama, Y. CoolTip: low-temperature solid-phase extraction microcolumn for capturing hydrophilic peptides and phosphopeptides. *Mol. Cell. Proteom.* **20**, 100170 (2021).
47. Kyono, Y., Sugiyama, N., Imami, K., Tomita, M. & Ishihama, Y. Successive and selective release of phosphorylated peptides captured by hydroxy acid-modified metal oxide chromatography. *J. Proteome Res.* **7**, 4585–4593 (2008).
48. Battellino, T., Ogata, K., Spicer, V., Ishihama, Y. & Krokhin, O. Acetic acid ion pairing additive for reversed-phase HPLC improves detection sensitivity in bottom-up proteomics compared to formic acid. *J. Proteome Res.* **22**, 272–278 (2022).
49. Mason, E. A. & Schamp, H. W. Mobility of gaseous ions in weak electric fields. *Ann. Phys.* **4**, 233–270 (1958).
50. Vaswani, A. et al. Attention is all you need. In *Advances in Neural Information Processing Systems* **30** (eds. Guyon, I. et al.) (Curran Associates, Inc., 2017).
51. Okuda, S. et al. jPOSTrepo: an international standard data repository for proteomes. *Nucleic Acids Res.* **45**, D1107–D1111 (2017).

Acknowledgements

This work was supported by JST, CREST Grant Number JPMJCR1862, Japan and MEXT KAKENHI Grant Number 23H04924, 24H00740 and 24H01895. We would like to thank Genki Takahashi, a master's student at the Graduate School of Informatics, Kyoto University, for verifying codes and formulae in this paper, and for conducting preliminary experiments.

Author contributions

A.N.-K., T.T. and Y.I. designed the experiments of the deep learning models, analyzed the data, and interpreted the results. A.N.-K. performed all computational experiments. Y.I. and T.T. designed the main conceptual ideas. K.O. and Y.I. collected all experimental data, and analyzed the results of Fig. 7. All authors wrote the manuscript and discussed the results. Y.I. and T.T. supervised the project and provided resources for the study.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42004-025-01540-z.

Correspondence and requests for materials should be addressed to Yasushi Ishihama or Toshiyuki Tanaka.

Peer review information *Communications Chemistry* thanks Ulises Hernández Guzmán, Ceder Dens, Pavel Sinitcyn, Wout Bittremieux, and the other, anonymous, reviewers for their contribution to the peer review of this work. Peer reviewer reports are available.