pDeep3: Towards More Accurate Spectrum Prediction with Fast Few-Shot Learning

Ching Tarn,[†,‡,¶] Wen-Feng Zeng,[*,†,‡,¶] Zhengcong Fei,[†,‡] and Si-Min He[†,‡]

†*Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology (ICT), CAS, Beijing 100190, China.*

‡*University of Chinese Academy of Sciences, Beijing, China.*

¶*These authors contributed equally to this work.*

E-mail: zengwenfeng@ict.ac.cn

Abstract

Spectrum prediction using deep learning has attracted a lot of attention in recent years, and has been used in the analysis of data-dependent and data-independent acquisition mass spectrometry. Although existing deep learning methods have greatly increased the prediction accuracy, there is still considerable space for improvement, which is presently limited by the difference of instrument types or settings. In this work, we use the few-shot learning method to fit the data online to make up for the shortcoming. We evaluate the method using 9 commonly used datasets, where the instruments include Velos, QE, Lumos, and ABSciex, with NCEs being differently set. Experimental results show that, on nearly all of the datasets, pDeep3 achieves significantly higher prediction accuracy, with extermly low costs. For example, on a trypsin dataset Pandey-Velos of Orbitrap Velos from Human Proteome Map, with only 100 randomly selected peptide-spectrum matches, it increases the accuracy from 49% to 85%, and only takes 7 seconds using CPU. Further, we also show that our design is sufficiently effective to fill the accuracy gap, and not demanding on data quality.

Finally, using OpenSWATH and EncyclopeDIA, we present the potential advantages of pDeep3 for DIA data analysis. The source code of pDeep3 is available at `http://pfind.ict.ac.cn/software/pDeep3`.

Machine learning algorithms currently play important roles in mass spectrometry-based proteomics.[1–4] In recent years, deep learning-based methods have attracted more and more attention, for both data-dependent acquisition (DDA) and data-independent acquisition (DIA) data analysis.[5–14]

For DDA data analysis, the recently published pDeep,[5] Prosit[6] and DeepMass:Prism[7] showed that integrating spectrum prediction into DDA search engines would significantly increase the identification rates. Our recent work, pValid, further demonstrated that the spectra predicted by pDeep could be used to filter out unreliable peptide spectrum matches (PSMs), resulting in more accurate peptide identifications.[15] We also used the predicted spectra to re-rank the *de novo* sequenced peptides and achieved higher recall and precision.[16]

For DIA data analysis, Prosit and DeepMass:Prism showed that the performance of predicted spectral libraries was close to that of experimental ones in spectral library search. DeepDIA built predicted spectral libraries of peptides from nearly the full human proteome for DIA analysis, and detected more proteins than using the pre-built DDA spectral libraries.[11] Searle *et al.* also used a full proteome library predicted by Prosit and corrected it using empirical spectra.[17] A recent work showed that, combining the experimental spectral library with the predicted library for proteins of interest will deepen the coverage of targeted protein families.[18]

It turns out that more accurate predictions result in better identification performance for both DDA and DIA analysis.[6,7] However, in practical uses, not all MS/MS predictions are sufficiently accurate.[9,11,19] Prosit tried to correct the predicted spectra by enumerating all predicting collision energies; DeepDIA aimed at training instrument-specific models for more accurate predictions. Unfortunately, either enumerating all energies or training models for specific instruments is time-consuming, and hence, accelerating devices such as GPUs

are indispensable. Moreover, our previous work, pDeep2, showed that transfer learning is an effective way to improve the prediction of modified peptides from a pre-trained unmodified model requiring only tens of thousands of spectra with modifications for training. Due to the huge computational costs, the training step could be done within minutes using a GPU, but it will take tens of minutes even hours when only CPUs are available.

Pre-trained offline models can hardly achieve the same accuracy as online-trained models, since the latter can automatically learn the characteristics of different MS instrument settings. However, online learning from scratch requires huge computing resources to process spectra in real time and also a lot of data from the specific instrument. Inspired by the few-shot learning technique,[20] we use a few-shot fine-tuning method to train models for specific instruments based on a pre-trained offline model with only 100 training spectra, achieving higher accuracy and higher speed on CPU-level processing. As the name "few-shot" implies, few-shot learning refers to the practice of learning a model with a very small amount of training data, and has been widely used in computer vision and other machine learning tasks. There are two key advantages of few-shot learning: 1) fine-tuning a low accuracy pre-trained model would increase the accuracy of predictions; 2) fine-tuning could be done within only a few seconds on CPU, making training lab-specific, or even sample-specific, models applicable using a personal computer. Performance of the spectral library predicted by pDeep3 for DIA data analysis is also evaluated. The results demonstrate the potential advantages of the predicted libraries with fine-tuning. The source code of pDeep3 including spectral library generation is available at `http://pfind.ict.ac.cn/software/pDeep3`.

Methods

The fine-tuning procedure of pDeep3 is illustrated in Figure 1. The pre-trained model is constructed using millions of spectra from different instruments and collision energies, as described in our previous work.[9] The network parameters are then fine-tuned with about

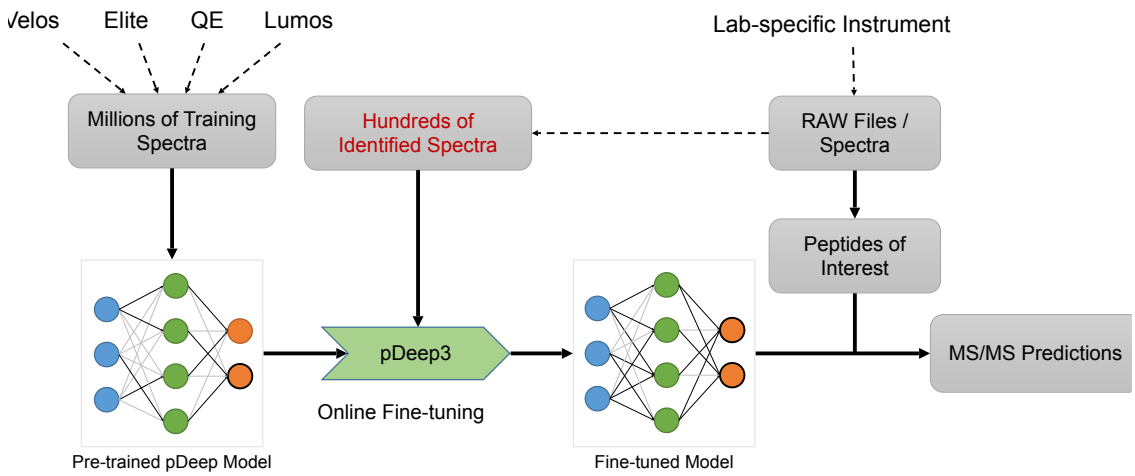

Figure 1: Workflow of the fine-tuning procedure of pDeep3.

100 spectra from lab-specific datasets to achieve better predictions for the lab-specific instrument. The main hyper-parameters of fine-tuning are: epoch number = 2, learning rate = 0.001, mini-batch size = 1024. Since the peptides with different lengths are assigned into different mini-batch respectively, the actual size of each mini-batch depends on peptides' lengths. Different values of the hyper-parameters are also evaluated, as shown in the section of "Results and Discussions".

The quality of pre-trained model is the key to fast fine-tuning. It ensures that the network parameters are already close to the optimal ones for the lab-specific instrument, and hence we can tune the model to fit the lab-specific data with only a few back-propagation updates. As reported in our previous work,[9] the pre-trained pDeep model are tested on Velos, Elite, QE and Lumos datasets, and achieves about 0.98 median Pearson correlation coefficients in most of the test datasets, indicating that the pre-trained model is very suitable for fine-tuning.

Spectral library generation

pDeep3 supports generating spectral libraries for EncyclopeDIA[21] and OpenSWATH.[22] For EncyclopeDIA analysis, pDeep3 generates SQLite-based .dlib library; for OpenSWATH anal-

ysis, pDeep3 generates SQLite-based .pqp library or text-based .tsv library. For the predicted libraries, pDeep3 predicts the retention time (RT) of the peptides using a pDeep3-build-in RT prediction model (pDeepRT, source codes can be found in pDeep3).

pDeep3 supports various formats of input peptide sequences, including the protein fasta files, spectral library files (.tsv, .dlib, etc.), pFind3 search results, MaxQuant search results, etc. When the input is a fasta file, pDeep3 automatically cleaves the protein sequences into peptide sequences, and adds the user-specific fixed and variable modifications to the peptide sequences. The transfer learning technique used in pDeep2[9] is also applicable in pDeep3 for modified peptides.

In this manuscript, we use Pearson correlation coefficient (PCC) as the spectrum-to-spectrum similarity measure, which is also used in pDeep[5] and pDeep2.[9] The median PCC, $PCC_{>0.75}$ and $PCC_{>0.90}$ are used to evaluate the performance of different methods, where $PCC_{>t}$ is defined as the proportion of PCCs greater than t in the test dataset.

Results and Discussions

Data preparation

Diverse datasets are collected from the test datasets of our previous work,[9] which are all publicly available.[23–27] For each dataset, we only consider spectra from five HCD RAW files, which are sufficient for fine-tuning and testing. Dataset information is listed in Table 1. Additionally, we add an ABSciex-6600 DDA dataset from LFQBench-HE,[27] abbreviated as "LH" in Table 1. ABSciex-6600 is a TripleTOF-based instrument, of which the settings are different from those of Orbitrap-based Thermo's instruments. The pre-trained pDeep model is never trained by TripleTOF data before, and hence LFQBench-HE can be used to test the performance of the pre-trained model and the fine-tuned model for the "unknown" instruments.

Table 1: Dataset information

Dataset	Description	Abbr.	Instrument	NCE	#RAWs	#PSMs
Kuster-Velos	Human, Milk	KM	Velos	40	5	35740
Pandey-Velos	Human, Fetal-Gut	PF	Velos	35	5	46114
Olsen-293-QE	HEK293T, Trypsin	O2	QE	28	5	71788
Olsen-HeLa-QE	HeLa, Trypsin	OH	QE	28	5	106006
Olsen-GC-QE	HeLa, GluC	OG	QE	28	5	74826
Olsen-CT-QE	HeLa, Chymotrypsin	OC	QE	28	5	60191
Olsen-LC-QE	HeLa, LysC	OL	QE	28	5	98985
PT-Lumos	ProteomeTools	PL	Lumos	30	5	41772
LFQBench-HE	HeLa and E.coli	LH	AB-6600	?[a]	4	61039

[a] "?": the energy unit of ABSciex-6600 is different from the NCE of Orbitrap-based data.

Hyper-parameter selection

In the fine-tuning step, the number of epochs e and the size of training set n are two key hyper-parameters for the trade-off of the training cost and predicting accuracy. Here we use three sets of experiments to select these hyper-parameters. It is worth mentioning that, for each dataset, we only partially select peptide-spectrum matches (PSMs) from a RAW file when fine-tuning, and test the fine-tuned model using all PSMs from one RAW of the rest files.

To select an appropriate e in the fine-tuning step, we try e in $\{0, 1, \ldots, 8\}$, and set n as 100. The test results are shown in Figure 2a. It turns out that, as e increases, the performance on most of the datasets improves as well, and the test $\text{PCC}_{>0.75}$ and $\text{PCC}_{>0.90}$ become stable after about two epochs. In particular, as the pre-trained model has been trained by other peptides of ProteomeTools,[9] the test results on PL are already close to the best accuracy, hence there is no further improvement in the PL dataset as e increases (Figure 2a). However, the performance is not decreased, indicating that fine-tuning may not suffering from the potential risks such as over-fitting.

We also try n in $\{0, 10, 30, 100, 300, 1000\}$, and set e as 2 for fine-tuning. The results are demonstrated in Figure 2b. The performance steadily gets better as n increases. And for PL, the results are not much changed as well. The improvements in most of the datasets

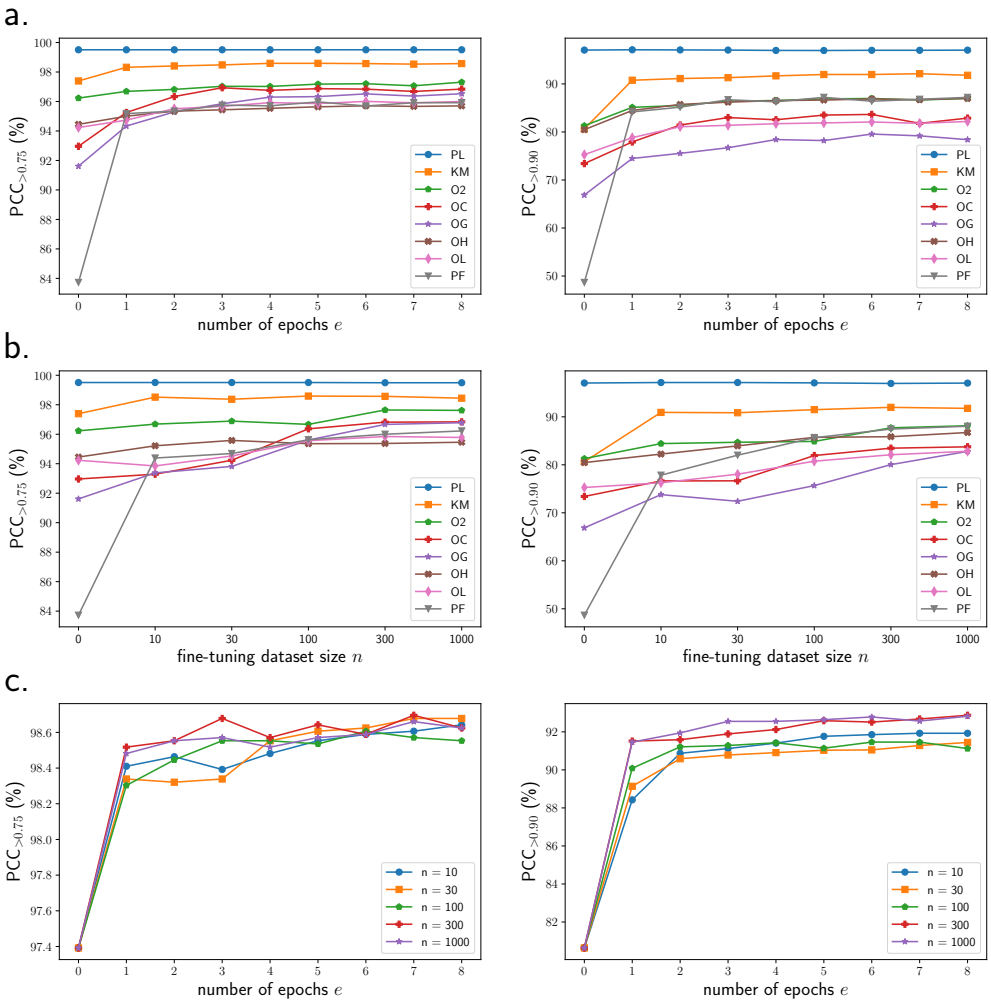

Figure 2: Tests of different hyper-parameters, training epochs e and training set size n, on different datasets. (a) Tests of e with $n = 100$. (b) Tests of n with $e = 2$. (c) Tests of different combinations of e and n on the KM dataset.

gradually slow down as n increases. Specifically, the improvements on KM and PF have achieved the optimal when only about 10 or 30 samples are used; for Olsen's datasets (O2, OC, OG, OH, OL), the increases are roughly linearly related to $\log n$.

We further try different combinations of e and n for $e \in \{0, 1, \ldots, 8\}$ and $n \in \{10, 30, 100, 300, 1000\}$. The test results on the KM dataset are illustrated in Figure 2c. As shown in Figure 2c, the effects of e and n are quite independent. That is, even if n is different, the method achieves a large increase after about two epochs, and the improvement slows down thereafter; with different numbers of epochs e, the larger the size of the training set is, the better the results

are, and the growth rate gradually slows down as n increases. Results on other datasets can be found in Data S1.

Considering that the time required for fine-tuning is relative to the number of epochs e and the size of the training set n, to ensure that fine-tuning can be completed using CPU within an acceptable time, it is necessary to limit the scale without significantly sacrificing the performance. The test results in Figure 2 demonstrate that using 100 training samples and two epochs, the fine-tuned model can reach good performance in both fine-tuning speed and predicting accuracy. In our tests, the fine-tuning can be finished within only 10 seconds using an Intel CPU (without GPU devices). These results show that, for any dataset to be predicted, fine-tuning can always be used before predicting.

Intra-RAW and inter-RAW tests

For a dataset, we question that whether we need to fine-tune different models for different RAW files of the dataset (run-specific fine-tuning), or we only need to fine-tune a model for all RAW files (sample-/instrument-specific fine-tuning). To answer it, we apply intra-RAW and inter-RAW test experiments.

For each dataset, we split PSMs in each RAW file into a training set and a testing set, where the training set contains 100 PSMs and the testing set contains the rests. The pre-trained model is fine-tuned by 5 training sets, respectively ($e = 2$, $n = 100$), resulting in 5 fine-tuned models. Each model is then tested separately on all the 5 testing sets, and we get a 5×5 $\text{PCC}_{>x}$ matrix $\mathbf{S}$. Each set of results of a model corresponds to a row of the matrix $\mathbf{S}$, where $\mathbf{S}_{ij}$ is the result of fine-tuning with RAW file i and testing with RAW file j. Test results on 5 RAW files of KM and OC datasets are listed in Table 2. We use standard deviation (σ) to measure the similarity between the results. On the same testing RAW file, fine-tuning by the training set of any RAW file achieves the similar $\text{PCC}_{>0.75}$ ($\sigma = 0.6\%$ for KM and $\sigma = 0.5\%$ for OC) and $\text{PCC}_{>0.90}$ ($\sigma = 1.6\%$ for KM and $\sigma = 1.3\%$ for OC), which implies that the model can be fine-tuned only once on a RAW file and applies to others in

Table 2: Intra-RAW and inter-RAW tests. r_i means the model is trained on the training set from RAW i, and s_j means it is tested on the testing set from RAW file j.

(a) KM

	$PCC_{>0.75}$ (%, $\sigma = 0.6\%$)					$PCC_{>0.90}$ (%, $\sigma = 1.6\%$)				
	s_1	s_2	s_3	s_4	s_5	s_1	s_2	s_3	s_4	s_5
r_1	98.6	98.3	98.7	97.5	99.2	93.7	91.8	93.1	93.0	96.0
r_2	98.6	98.4	98.7	97.6	99.2	93.6	92.1	93.9	93.1	95.9
r_3	98.6	98.4	98.7	97.5	99.3	93.1	91.0	92.9	92.1	95.6
r_4	98.5	98.3	98.7	97.5	99.2	93.4	91.5	93.0	93.3	96.0
r_5	98.6	98.4	98.6	97.6	99.2	92.3	90.2	92.4	92.6	96.0

(b) OC

	$PCC_{>0.75}$ (%, $\sigma = 0.5\%$)					$PCC_{>0.90}$ (%, $\sigma = 1.3\%$)				
	s_1	s_2	s_3	s_4	s_5	s_1	s_2	s_3	s_4	s_5
r_1	96.0	95.3	95.5	95.8	96.4	81.6	81.1	80.1	79.9	83.3
r_2	96.8	96.4	96.3	96.8	97.3	82.3	83.1	82.5	82.3	84.7
r_3	96.5	95.9	95.9	96.5	96.8	82.1	81.6	81.4	81.4	84.0
r_4	96.1	95.5	95.4	96.0	96.3	80.6	80.6	80.2	79.9	82.2
r_5	96.2	95.8	95.9	96.3	96.9	81.0	80.8	80.5	80.5	83.8

the same dataset. Results on other datasets can be found at Data S2.

Tests of training set selection bias

One concern of the fine-tuning step is that the 100 randomly selected PSMs hardly cover all characteristics of peptides (different lengths, different charge states, etc.). Once some properties of peptides are not selected for fine-tuning, the fine-tuned model may prefer the selected properties, resulting in poor predictions on peptides with non-selected properties. We call it "training set selection bias" of the fine-tuning model. To test whether the bias exists, we evaluate six training set selection methods, as listed in follows:

1. *low*. The training set only contains peptides with precursor charge ≤ 3;

2. *high*. The training set only contains peptides with precursor charge > 3;

3. *short*. The training set only contains peptides with length ≤ 12;

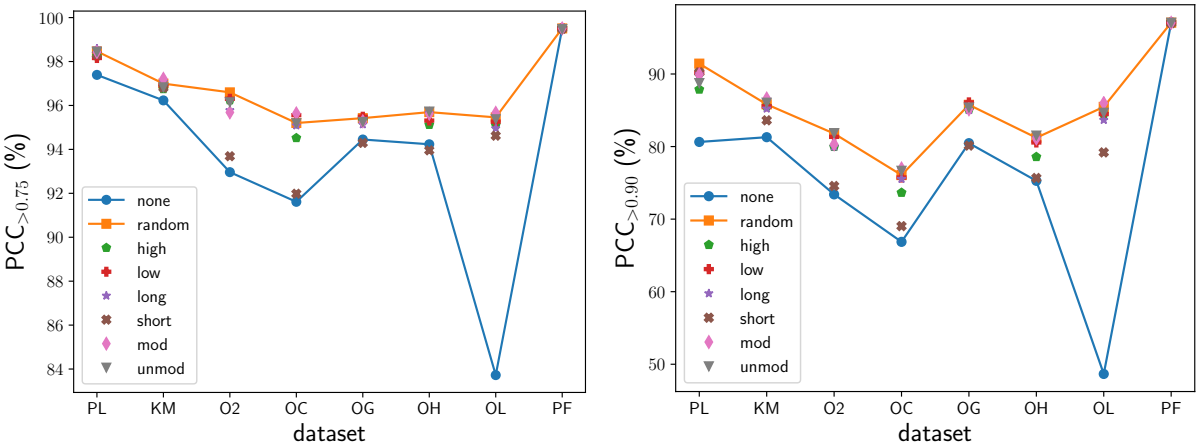

Figure 3: Test of different training set selection methods for fine-tuning.

4. *long*. The training set only contains peptides with length > 12;

5. *mod*. The training set only contains peptides with Oxidation[M];

6. *unmod*. The training set only contains peptides without any variable modification.

We also consider "random selection" (denoted as *random*) and "no fine-tuning" (denoted as *none*) for comparisons. All the models are fine-tuned by selected training sets and tested by another RAW ($e = 2$, $n = 100$). The results, as illustrated in Figure 3, show that a model with fine-tuning is always better than that without fine-tuning, no matter how the training data are selected. And random selection usually achieves the best performance.

Fine-tuning for "unknown" instruments

The pre-trained pDeep model is only trained by Thermo's instruments and NCEs, and all other instruments are set as "unknown" in pDeep. The other "unknown" factor is that the collision energy unit (i.e. NCE) of Thermo's instruments is very different from those of other instruments. The LH dataset is generated from ABSceix-6600, and the collision energy (CE) recorded in the WIFF files is 10. We firstly try to set the instrument as "AB6600" (will be converted to "unknown" in pDeep3) and CE as 10. The $\text{PCC}_{>0.75}$ and $\text{PCC}_{>0.90}$ are very

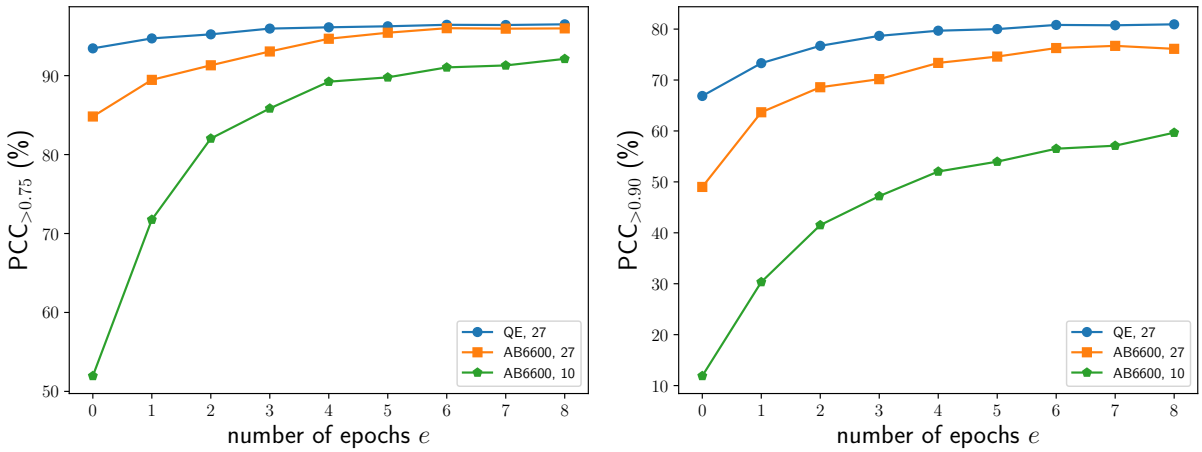

Figure 4: Few-shot learning for DDA data of ABSciex-6600 ("unknown") instrument. The collision energy recorded in the WIFF file is 10.

low even with fine-tuning, as illustrated in Figure 4. Setting the CE as 27 can significantly increase the predicting and fine-tuning performance, and setting the instrument as "QE" and CE as 27 achieves the best performance for both pre-trained and fine-tuned models. Therefore, for the other beam-type instruments, we recommend setting the instrument as "QE" and setting CE as a routine NCE of "QE".

Building spectral libraries for DIA analysis

Test of OpenSWATH

DIA and DDA RAW data are downloaded from Chorus Project with ID 1105.[28] The DDA RAW files (QEHF, NCE=27) are searched by pFind3[29] without the open search, and then the random selected 100 PSMs at 0.1% FDR are submitted to pDeep3 for fine-tuning with 2 epochs.

For DIA analysis, pah-human-library (PHL, in .tsv format) is downloaded from `http://www.swathatlas.org/`, and is converted to .pqp format by using "TargetedFileConverter" in OpenMS (version 2.4.0). We append decoy transitions by using "OpenSwathDecoyGenerator" with the "reverse" decoy method. Taking PHL as input, we generate the predicted

.pqp PHL without fine-tuning ("pre-trained") and with fine-tuning ("tuned") based on the downloaded PHL, and the decoy method is set as "reverse" in pDeep3. OpenSWATH needs reference peptides (such as iRT peptides) for retention time (RT) alignment between RAW files and libraries, and hence we use peptides from endogenous proteins in human cells (actin and vinculin) in PHL for RT alignment. For the predicted libraries, we predict the RT of the library peptides as well as the reference peptides using pDeepRT. The DIA RAW files (QEHF, NCE=27) are converted to mzML by ProteoWizard[30] with the default parameters. The searching parameters of OpenSWATH are: MS1 mass tolerance = ±10 ppm, MS2 mass tolerance = ±20 ppm, and "RT extraction window" = 600. The false discovery rate (FDR) is then estimated by pyprophet using the global-context estimation.[31] The test results of PHL, "pre-trained" and "tuned" at 1% MS2 level FDR are shown in Figure 5a. The tuned PHL only detects 98 more peptides than those from pre-tained PHL, because the pre-trained model already have very good prediction accuracy in the corresponding DDA RAW file (median PCCs are both 0.98 for pre-trained and tuned models).

Test of EncyclopeDIA

DIA and DDA RAW data are downloaded from PRIDE with ID PXD006722.[32] The DDA RAW files (QEHF, NCE=27) are searched by pFind3[29] without open search, then the random selected 100 PSMs at 0.1% FDR are used for fine-tuning with 2 epochs.

For DIA search, PHL is downloaded from `https://bitbucket.org/searleb/encyclopedia/downloads/pan_human_library.dlib`, then the pre-trained and tuned PHL in .dlib format are generated for EncyclopeDIA. The DIA RAW files (QEHF, NCE=27) are converted to mzML, and then searched by EncyclopeDIA (version 0.9.0) using PHL, pre-trained PHL, and tuned PHL, respectively. The searching parameter "acquisition" is set as overlapping and other searching parameters are left as default. The searched results are demonstrated in Figure 5b. The median PCC is 0.95 for the pre-trained model, and increases to 0.96 after fine-tuning, resulting in 260 more detected peptides.

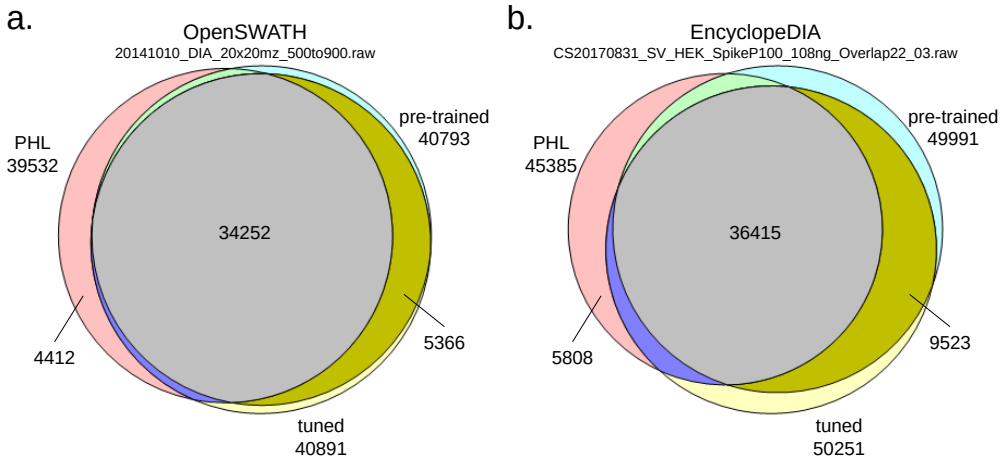

Figure 5: Detected peptides using OpenSWATH (a) and EncyclopeDIA (b) with PHL, predicted PHL without fine-tuning ("pre-trained"), and predicted PHL with fine-tuning ("tuned").

For both OpenSWATH and EncyclopeDIA, predicted (pre-trained and tuned) PHL can detect more peptides than the original library, demonstrating the potential advantages of spectrum prediction of pDeep3 for DIA analysis.

Conclusion

In this work, we show that a good pre-trained pDeep model can be improved using the few-shot learning technique. Few-shot learning aims to fine-tune the pDeep model with a small training set and a small number of training epochs, and is able to increase the predicting accuracy within only seconds. With the few-shot learning technique, a large pre-compiled spectral library, such as pan-human-library can be re-built by fine-tuning with about 100 instrument-specific DDA spectra. We also present that spectral library predicted by fine-tuned pDeep model can increase the number of detected peptides using OpenSWATH and EncyclopeDIA for DIA data analysis.

Few-shot learning, as well as spectral library generation, has been integrated into pDeep3. For end-users, pDeep3 supports generating spectral libraries for OpenSWATH (.tsv and

.pqp library file), EncyclopeDIA (.dlib file), and Spectronaut (.csv file). And for Python programmers, pDeep3 provides convenient programming interfaces to integrate spectrum prediction and fine-tuning into MS data analysis pipelines.

Acknowledgement

This work was supported by the National Key Research and Development Program of China (No. 2016YFA0501301 to S.-M. H.).

Supporting Information Available

- **Data S1.xlsx**: Test results of different combinations of e and n on all test datasets.
- **Data S2.xlsx**: Intra-RAW and inter-RAW test results on all test datasets.

References

(1) Kall, L.; Canterbury, J. D.; Weston, J.; Noble, W. S.; MacCoss, M. J. Semi-supervised learning for peptide identification from shotgun proteomics datasets. *Nature Methods* **2007**, *4*, 923–925.

(2) Sun, S.; Yang, F.; Yang, Q.; Zhang, H.; Wang, Y.; Bu, D.; Ma, B. MS-Simulator: predicting y-ion intensities for peptides with two charges based on the intensity ratio of neighboring ions. *Journal of Proteome Research* **2012**, *11*, 4509–4516.

(3) Xu, L. L.; Young, A.; Zhou, A.; Rost, H. L. Machine Learning in Mass Spectrometric Analysis of DIA Data. *Proteomics* **2020**, Just Accepted.

(4) Fei, Z. Improving Tandem Mass Spectra Analysis with Hierarchical Learning. *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)* **2020**, Just Accepted.

(5) Zhou, X.-X.; Zeng, W.-F.; Chi, H.; Luo, C.; Liu, C.; Zhan, J.; He, S.-M.; Zhang, Z. pDeep: Predicting MS/MS Spectra of Peptides with Deep Learning. *Analytical Chemistry* **2017**, *89*, 12690–12697.

(6) Gessulat, S.; Schmidt, T.; Zolg, D. P.; Samaras, P.; Schnatbaum, K.; Zerweck, J.; Knaute, T.; Rechenberger, J.; Delanghe, B.; Huhmer, A. et al. Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning. *Nature Methods* **2019**, *16*, 509–518.

(7) Tiwary, S.; Levy, R.; Gutenbrunner, P.; Soto, F. S.; Palaniappan, K. K.; Deming, L.; Berndl, M.; Brant, A.; Cimermancic, P.; Cox, J. High-quality MS/MS spectrum prediction for data-dependent and data-independent acquisition data analysis. *Nature Methods* **2019**, *16*, 519–525.

(8) Zhou, X.-X.; Ren, R.; Gao, W.-L.; Huang, Y.-Y.; Zeng, W.-F.; Kong, D.-F.; Hao, T.-S.; Zhang, Z.-F.; Zhan, J.-F. Trends on Methods for Prediction of Tandem Mass Spectra of Peptides. *Progress in Biochemistry and Biophysics* **2019**, *46*, 169–180.

(9) Zeng, W.-F.; Zhou, X.-X.; Zhou, W.-J.; Chi, H.; Zhan, J.; He, S.-M. MS/MS Spectrum Prediction for Modified Peptides Using pDeep2 Trained by Transfer Learning. *Analytical Chemistry* **2019**, *91*, 9724–9731.

(10) Guan, S.; Moran, M. F.; Ma, B. Prediction of LC-MS/MS Properties of Peptides from Sequence by Deep Learning. *Molecular and Cellular Proteomics* **2019**, *18*, 2099–2107.

(11) Yang, Y.; Liu, X.; Shen, C.; Lin, Y.; Yang, P.; Qiao, L. In silico spectral libraries by deep learning facilitate data-independent acquisition proteomics. *Nature Communications* **2020**, *11*, 146.

(12) Liu, K.; Li, S.; Wang, L.; Ye, Y.; Tang, H. Full-Spectrum Prediction of Peptides Tandem Mass Spectra using Deep Neural Network. *Analytical Chemistry* **2020**, Just Accepted.

(13) Tran, N. H.; Zhang, X.; Xin, L.; Shan, B.; Li, M. De novo peptide sequencing by deep learning. *Proceedings of the National Academic of Sciences* **2017**, *114*, 8247–8252.

(14) Lin, Y.-M.; Chen, C.-T.; Chang, J.-M. MS2CNN: predicting MS/MS spectrum based on protein sequence using deep convolutional neural networks. *BMC Genomics* **2019**, *20*, 1–10.

(15) Zhou, W.-J.; Yang, H.; Zeng, W.-F.; Zhang, K.; Chi, H.; He, S.-M. pValid: Validation Beyond the Target-Decoy Approach for Peptide Identification in Shotgun Proteomics. *Journal of Proteome Research* **2019**, *18*, 2747–2758.

(16) Yang, H.; Chi, H.; Zeng, W.-F.; Zhou, W.-J.; He, S.-M. pNovo 3: precise de novo peptide sequencing using a learning-to-rank framework. *ISMB/Bioinformatics* **2019**, *35*, i183–i190.

(17) Searle, B. C.; Swearingen, K. E.; Barnes, C. A.; Schmidt, T.; Gessulat, S.; Kuster, B.; Wilhelm, M. Generating high quality libraries for DIA MS with empirically corrected peptide predictions. *Nature Communications* **2019**, *11*, 1548.

(18) Lou, R.; Tang, P.; Ding, K.; Li, S.; Tian, C.; Li, Y.; Zhao, S.; Zhang, Y.; Shui, W. Hybrid Spectral Library Combining DIA-MS Data and a Targeted Virtual Library Substantially Deepens the Proteome Coverage. *iScience* **2020**, *23*, 100903.

(19) Xu, R.; Sheng, J.; Bai, M.; Shu, K.; Zhu, Y.; Chang, C. A Comprehensive Evaluation of MS/MS Spectrum Prediction Tools for Shotgun Proteomics. *Proteomics* **2020**, 1900345.

(20) Wang, Y.; Yao, Q.; Kwok, J.; Ni, L. M. Generalizing from a Few Examples: A Survey on Few-Shot Learning. *arXiv* **2020**, *preprint at arXiv*.

(21) Searle, B. C.; Pino, L. K.; Egertson, J. D.; Ting, Y. S.; Lawrence, R. T.; MacLean, B. X.; Villen, J.; MacCoss, M. J. Chromatogram libraries improve peptide detection and quan-

tification by data independent acquisition mass spectrometry. *Nature Communications* **2018**, *9*, 5128.

(22) Röst, H. L.; Rosenberger, G.; Navarro, P.; Gillet, L.; Miladinović, S. M.; Schubert, O. T.; Wolski, W.; Collins, B. C.; Malmström, J.; Malmström, L. et al. OpenSWATH enables automated, targeted analysis of data-independent acquisition MS data. *Nature Biotechnology* **2014**, *32*, 219–223.

(23) Wilhelm, M.; Schlegl, J.; Hahne, H.; Gholami, A. M.; Lieberenz, M.; Savitski, M. M.; Ziegler, E.; Butzmann, L.; Gessulat, S.; Marx, H. et al. Mass-spectrometry-based draft of the human proteome. *Nature* **2014**, *509*, 582–587.

(24) Kim, M.-S.; Pinto, S. M.; Getnet, D.; Nirujogi, R. S.; Manda, S. S.; Chaerkady, R.; Madugundu, A. K.; Kelkar, D. S.; Isserlin, R.; Jain, S. et al. A draft map of the human proteome. *Nature* **2014**, *509*, 575–581.

(25) Bekker-Jensen, D. B.; Kelstrup, C. D.; Batth, T. S.; Larsen, S. C.; Haldrup, C.; Bramsen, J. B.; Sørensen, K. D.; Høyer, S.; Ørntoft, T. F.; Andersen, C. L. et al. An Optimized Shotgun Strategy for the Rapid Generation of Comprehensive Human Proteomes. *Cell systems* **2017**, *4*, 587–599.e4.

(26) Zolg, D. P.; Wilhelm, M.; Schnatbaum, K.; Zerweck, J.; Knaute, T.; Delanghe, B.; Bailey, D. J.; Gessulat, S.; Ehrlich, H.-C.; Weininger, M. et al. Building ProteomeTools based on a complete synthetic human proteome. *Nature Methods* **2017**, *14*, 259–262.

(27) Navarro, P.; Kuharev, J.; Gillet, L. C.; Bernhardt, O. M.; MacLean, B.; Röst, H. L.; Tate, S. A.; Tsou, C.-C.; Reiter, L.; Distler, U. et al. A multicenter study benchmarks software tools for label-free proteome quantification. *Nature Biotechnology* **2016**, *34*, 1130–1136.

(28) Ting, Y. S.; Egertson, J. D.; Bollinger, J. G.; Searle, B. C.; Payne, S. H.; Noble, W. S.;

MacCoss, M. J. PECAN: library-free peptide detection for data-independent acquisition tandem mass spectrometry data. *Nature Methods* **2017**, *14*, 903–908.

(29) Chi, H.; Liu, C.; Yang, H.; Zeng, W.-F.; Wu, L.; Zhou, W.-J.; Wang, R.-M.; Niu, X.-N.; Ding, Y.-H.; Zhang, Y. et al. Comprehensive identification of peptides in tandem mass spectra using an efficient open search engine. *Nature Biotechnology* **2018**, *36*, 1059–1061.

(30) Chambers, M. C.; Maclean, B.; Burke, R.; Amodei, D.; Ruderman, D. L.; Neumann, S.; Gatto, L.; Fischer, B.; Pratt, B.; Egertson, J. et al. A cross-platform toolkit for mass spectrometry and proteomics. *Nature Biotechnology* **2012**, *30*, 918–920.

(31) Rosenberger, G.; Bludau, I.; Schmitt, U.; Heusel, M.; Hunter, C. L.; Liu, Y.; MacCoss, M. J.; MacLean, B. X.; Nesvizhskii, A. I.; Pedrioli, P. G. A. et al. Statistical control of peptide and protein error rates in large-scale targeted data-independent acquisition analyses. *Nature Methods* **2017**, *14*, 921–927.

(32) Peckner, R.; Myers, S. A.; Jacome, A. S. V.; Egertson, J. D.; Abelin, J. G.; MacCoss, M. J.; Carr, S. A.; Jaffe, J. D. Specter: linear deconvolution for targeted analysis of data-independent acquisition mass spectrometry proteomics. *Nature Methods* **2018**, *15*, 371–378.