DeepRTAlign: toward accurate retention time alignment for large cohort mass spectrometry data analysis

Yi Liu[1,2,#], Yun Yang[3,4,#], Wendong Chen[3,4], Feng Shen[5], Linhai Xie[2,3,4], Yingying Zhang[2,6], Yuanjun Zhai[2], Fuchu He[2,7], Yunping Zhu[2,*], Cheng Chang[2,7,*]

1. Faculty of Environment and Life, Beijing University of Technology, Beijing 100023, China.
2. State Key Laboratory of Proteomics, Beijing Proteome Research Center, National Center for Protein Sciences (Beijing), Beijing Institute of Lifeomics, Beijing 102206, China.
3. International Academy of Phronesis Medicine (Guang Dong), No. 96 Xindao Ring South Road, Guangzhou International Bio Island, Guangzhou 510000, China
4. South China Institute of Biomedicine, No. 83 Ruihe Road, Guangzhou 510535, China
5. Department of Hepatic Surgery IV, the Eastern Hepatobiliary Surgery Hospital, Naval Medical University, Shanghai 200433, China
6. Chongqing Key Laboratory on Big Data for Bio Intelligence, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
7. Research Unit of Proteomics Driven Cancer Precision Medicine, Chinese Academy of Medical Sciences, Beijing 102206, China.

[#]These authors contributed equally: Yi Liu and Yun Yang

* To whom correspondence should be addressed:

Yunping Zhu, Email: zhuyunping@gmail.com

Cheng Chang, Email: changchengbio@163.com

Abstract

Retention time (RT) alignment is one of the crucial steps in liquid chromatography-mass spectrometry (LC-MS)-based proteomic and metabolomic experiments, especially for large cohort studies, and it can be achieved using computational methods; the most popular methods are the warping function method and the direct matching method. However, the existing tools can hardly handle monotonic and non-monotonic RT shifts simultaneously. To overcome this, we developed a deep learning-based RT alignment tool, DeepRTAlign, for large cohort LC-MS data analysis. It first performs a coarse alignment by calculating the average time shift between any two samples and then uses RT and m/z as the main features to train its deep learning-based model. We demonstrate that DeepRTAlign has improved performances, especially when handling complex samples, by benchmarking it against current state-of-the-art approaches on 19 real-world proteomic and metabolomic datasets and the corresponding simulated datasets. Benchmarked on a dataset with known fold changes, the results showed that DeepRTAlign can improve the identification sensitivity of MS data without compromising the quantitative accuracy. Furthermore, using the MS features aligned by DeepRTAlign in a large cohort, we trained a classifier of 15 features to predict the early recurrence of hepatocellular carcinoma. The features were validated on an independent cohort using targeted proteomics with an AUC of 0.833. Being flexible and robust with four different feature extraction tools, DeepRTAlign provides an advanced solution to RT alignment in large cohort LC-MS data, which is currently one of the bottlenecks in proteomics and metabolomics research, especially for clinical applications.

Introduction

Liquid chromatography (LC) is usually coupled with mass spectrometry (MS) in proteomics experiments to separate complex samples. The retention time (RT) of each analyte in MS data usually have shifts for multiple reasons, including matrix effects and instrument performances[1]. Thus, in any experiment involving multiple samples, corresponding analytes must be mapped before quantitative, comparative, or statistical analysis. This process is called correspondence[2]. In other words, this problem can be defined as finding the "same compound" in multiple samples. Generally, in proteomics, correspondence can be done based on peptide identifications. However, taking the test data in this study as an example, only 15%-25% of precursors have the corresponding identifications due to the data-dependent ion selection process in the data-dependent acquisition (DDA) mode. Even for the data-independent acquisition (DIA) data, a number of unidentified precursors (potential peptides) remain, which cannot be considered in the subsequent analysis due to the complex MS/MS spectra[3]. Most existing tools for DDA and DIA data analysis, such as MaxQuant[4], PANDA[5], MSFragger[6,7] and DIA-NN[8], perform RT alignment using the match between runs (MBR) function (also called the cross-align function) to transfer the identified sequences to the unidentified precursors between any two LC-MS runs. Although MBR can increase the total number of identifications to some extent, it is integrated into specific software tools and relies on the identified peptides, which limits its further application in clinical proteomics research to explore new biomarkers from unidentified precursors. In metabolomics, feature alignment is a prerequisite for identification and quantification. In theory, the accuracy of feature alignment depends on the m/z and RT information in MS data. Currently, high-resolution mass spectrometers can limit the m/z shift to less than 10 ppm. Thus, RT alignment is especially important for accurately analyzing large-scale proteomics and metabolomics research data.

There are two types of computational methods for RT alignment. One is called the warping method. Warping models first correct the RT shifts of analytes between

runs by a linear or non-linear warping function[2,9]. There were several existing popular alignment tools based on this method, such as XCMS[10], MZmine 2[11] and OpenMS[12]. However, this warping method is not able to correct the non-monotonic shift because the warping function is monotonic[9]. Another kind of method is the direct matching method, which attempts to perform correspondence solely based on the similarity between specific signals from run to run without a warping function. The representative tools include RTAlign[13], MassUntangler[14] and Peakmatch[15]. The performances of the existing direct matching tools are reported inferior to the tools using warping functions due to the uncertainty of MS signals[2]. Either way, these tools mentioned above can hardly handle both monotonic and non-monotonic RT shifts. Thus, machine learning or deep learning techniques are applied to solve this issue. Li et al. applied a Siamese network for accurate peak alignment in gas chromatography-mass spectrometry data from complex metabolomic samples[16]. But there is no deep learning-based alignment algorithm for LC-MS data analysis currently.

Here, we present a deep learning-based RT alignment tool, named DeepRTAlign, for large cohort LC-MS data analysis. Combining a coarse alignment (pseudo warping function) and a deep learning-based model (direct matching), DeepRTAlign can deal with monotonic shifts as well as non-monotonic shifts. We have demonstrated its high accuracy and sensitivity in several proteomic and metabolomic datasets compared with existing popular tools. Further, DeepRTAlign allows us to apply MS features directly and accurately to downstream biological analysis, such as biomarker discovery or prognosis prediction, which can complement traditional identification (ID)-based methods (**Supplementary Notes**).

Methods and Materials

Datasets

As shown in **Supplementary Table 1**, all the datasets used in training and testing the deep learning model in DeepRTAlign (i.e., one training set and seven test sets) were collected from the following papers: (1) The training set HCC-T is from the tumor samples of 101 early-stage hepatocellular carcinoma (HCC) patients in Jiang et al.'s paper[17]. The corresponding non-tumor dataset (HCC-N) was considered as a test set in this study. (2) Dataset HCC-R was from the tumor samples of 11 HCC patients who had undergone liver transplantations in Bhat et al.[18]. (3) Datasets UPS2-M and UPS2-Y were from Chang et al.[19], and were constructed by spiking 48 UPS2 standard proteins (Proteomics Dynamic Range Standard, Sigma-Aldrich) into mouse cell digestion mixture (UPS2-M) and yeast digestion mixture (UPS2-Y), respectively. The detailed experimental design is shown in **Supplementary Fig. 3**. (4) Dataset EC-H was from Shen et al.[20], which was constructed by spiking *Escherichia coli* cell digestion mixture into human cell digestion mixture. (5) Dataset AT was based on the *Arabidopsis thaliana* seeds (AT) from Ginsawaeng et al.[21]. (6) Dataset SC was a single-cell (SC) proteomic dataset including 18 HT22 cells without the treatment of nocodazole from Li et al.[22].

Further, we tested the generalizability boundary of DeepRTAlign on the seven test sets and seven other datasets (five metabolomic datasets and two proteomic datasets). The five metabolomic datasets include (1) Dataset NCC19, a large-scale plasma analysis of SARS-CoV-2 from Barberis et al.[23]; (2) Dataset SM1100, comprised standard mixtures consisting of 1100 compounds, from Li et al.[24]; (3) Dataset MM, which was from the *Mus musculus* samples in Wase et al.[25]; (4) Dataset SO, which was from soil samples in Swenson et al.[26]; (5) Dataset GUS, which was about global untargeted serum metabolomic samples from Gibson et al.[27]. The two proteomic datasets are (1) Dataset MI, which was about mouse intestinal proteomes from Lichtman et al.[28]; (2) Dataset CD, which was obtained from the gut microbiota of patients with Crohn's disease in Mottawea et al.[29].

In this work, we generated five new datasets **(Supplementary Table 1**): Dataset HCC-R2 was generated from the tumor samples of 23 HCC patients using parallel reaction monitoring (PRM). It was used to validate the 15-feature-based classifier for predicting HCC early recurrence (**Supplementary Methods**). Dataset Benchmark-FC was a mixture of Human embryonic kidney (HEK) 293T samples and *E. coli* samples with known fold changes. In this dataset, 10 ng, 15 ng, 20 ng, and 25 ng of *E. coli* samples were spiked into 200 ng of HEK 293T samples, respectively. Each group contained 3 replicates. It was used for quantification accuracy evaluation. Dataset Benchmark-QC-H consisted of pure HEK 293T with three replicates, and dataset Benchmark-QC-E consisted of pure *E. coli* with three replicates. Both datasets were used to evaluate the QC module in DeepRTAlign. Dataset Benchmark-RT contained two HEK 293T samples with different RT gradients (60 min and 120 min), which were used for evaluating the performance of DeepRTAlign on different LC conditions.

Workflow of DeepRTAlign

<Figure 1>

The whole workflow of DeepRTAlign is shown in **Fig. 1**, and can be divided into two parts, i.e., the training part and the application part. The training part contains the following steps.

(1) Precursor detection and feature extraction. Taking raw MS files as input, precursor detection and its feature extraction were performed using an in-house developed tool XICFinder, an MS feature extraction tool similar to Dinosaur[30]. The algorithm of XICFinder is based on our quantitative tool PANDA[5]. Using the Component Object Model (COM) of MSFileReader, it can handle Thermo raw files directly. XICFinder first detects isotope patterns in each spectrum. Then, the isotope patterns detected in several subsequent spectra are merged into a feature. A mass tolerance of 10 ppm was used in this step.

(2) Coarse alignment. First, the RT in all the samples will be linearly scaled to a certain range (e.g., 80 min in this study, as the RT range of training dataset HCC-T is 80 min). Second, for each m/z, the feature with the highest intensity is selected to build a new list for each sample. Then, all the samples except an anchor sample (we

considered the first sample as the anchor in this study) will be divided into pieces by a user-defined RT window (we used 1 minute in this study). All the features in each piece are compared with the features in the anchor sample (mass tolerance: 0.01 Da). If the same feature does not exist in the anchor sample, this feature is ignored. Then, the RT shift is calculated for each feature pair. For all the features in each piece, the average RT shift is calculated. To align each piece with the anchor sample, the average RT shift between a piece and the anchor sample is directly added to each feature in this piece.

(3) Binning and filtering. All features will be grouped based on m/z, according to the parameters bin_width and bin_precision (0.03 and 2 by default). Bin_width is the m/z window size, and bin_precision is the number of decimal places used in this step. Only the features in the same m/z window will be aligned. After the binning step, there is also an optional filtering step. For each sample in each m/z window, only the feature with the highest intensity will be kept in the user-defined RT range. DeepRTAlign does not perform the filtering step by default.

(4) Input vector construction. Only the RT and m/z of each feature are considered when constructing the input vector. As shown in **Fig. 1a**, we consider two adjacent features (according to RT) before and after the target feature corresponding to a peptide. Part 1 and Part 4 are the original values of RT and m/z. Part 2 and Part 3 are the difference values between the two samples. Then, we used two base vectors (base_1 is [5, 0.03] and base_2 is [80, 1500]) to normalize the features (the difference values will be divided by base_1, the original value will be divided by base_2). Each feature-feature pair will be transferred to a 5×8 vector as the input of the deep neural network (**Supplementary Fig. 1**).

(5) Deep neural network (DNN). The DNN model in DeepRTAlign contains three hidden layers (each has 5000 neurons), which is used as a classifier that distinguishes between two types of feature-feature pairs (i.e., the two features should be or not be aligned). Finally, a total of 400,000 feature-feature pairs were collected from the HCC-T dataset based on the Mascot identification results (mass tolerance: ± 10 ppm, restrict the RT of a peptide to be within the RT range of the corresponding

precursor feature). 200,000 of them are collected from the same peptides, which should be aligned (labeled as positive). The other 200,000 are collected from different peptides, which should not be aligned (labeled as negative). These 400,000 feature-feature pairs were used to train the DNN model. It should be noted that it is not necessary to know the peptide sequences corresponding to the features when performing feature alignment. The identification results of several popular search engines (such as Mascot, MaxQuant and MSFragger) are only used as ground truths when benchmarking DeepRTAlign.

(6) The hyperparameters in DNN. BCELoss function in Pytorch is used as the loss function. The sigmoid function was used as the activation function. We used the default initialization method of pytorch (kaiming_uniform). We used Adam (betas=(0.9, 0.999), eps=1e-08, weight_decay=0, amsgrad=False, foreach=None, maximize=False, capturable=False, differentiable=False, fused=None) as the optimizer. The initial learning rate is set to 0.001, and is multiplied by 0.1 every 100 epochs. Batch size is set to 500. The number of epochs is chosen empirically. We conducted several trial runs and found that the loss tended to be stable at 100-300. So, we set the epoch number to 400 for final training. All other parameters are kept by default in Pytorch v1.8.0.

(7) Parameter evaluation. Network parameters used were examined on the training set (HCC-T) by 10-fold cross validation and the best parameters were selected based on the cross-validation results (**Supplementary Table 2**). The trained model was evaluated on several independent test sets (**Supplementary Table 3**). These results demonstrated that there is no overfitting in the DNN model.

(8) Quality control (QC) of the output. A QC module was implemented in DeepRTAlign to perform QC for the output of DeepRTAlign. As shown in **Supplementary Fig. 2**, in each m/z window, DeepRTAlign will randomly select a sample as a target and build its decoy. In theory, all features in the decoy sample should not be aligned. Based on this, the final false discovery rate (FDR) of the alignment results can be calculated.

In the application part (**Fig. 1b**), DeepRTAlign directly supports the results of

four MS feature extraction tools, i.e., Dinosaur[30], MaxQuant, OpenMS and XICFinder as input. Feature lists from other tools (such as MZmine 2) can be used after format conversion. Refer to the formats of the four tools mentioned above. In this part, feature lists will first go through the coarse alignment and input vector construction steps, as shown in the training part. Then, the constructed input vectors will be fed into the trained DNN model. According to the classification results of the DNN model, DeepRTAlign will output an aligned feature list for further analysis.

Machine learning models for evaluation

To make a systematical evaluation of our DNN model's performance, we compared it with several popular machine learning methods, i.e., random forests (RF), k-nearest neighbors (KNN), support vector machine (SVM) and logistic regression (LR). The parameters for each machine learning method were optimized based on the 10-fold cross validation results of the training set HCC-T (**Supplementary Tables 4-7**). The parameters of the RF model are number of estimators 50, and max depth 50. For KNN, the k is set to 5. For SVM, the kernel function is set to 'poly' (gamma='auto', C=1, and degree=3). For LR, the penalty is L2 and the solver is "lbfgs". All the other parameters are default values in scikit-learn v0.21.3. In total, we trained four machine learning models (named as RF, KNN, SVM and LR) as references in this study.

Tools for alignment comparison

As shown in **Supplementary Table 8**, the existing alignment tools can be classified into three types based on the input information required. The representative tools in each type were compared with DeepRTAlign in this study.

First, two existing popular alignment tools (MZmine 2 and OpenMS) were used for alignment comparison because the two tools showed the best precision and recall in Lange et al.[31] and Pluskal et al. [11]. The recommended parameters in the official user manuals of MZmine 2 and OpenMS were used. For MZmine 2, the ADAP-LC workflow was used. For OpenMS, we used FeatureFinderCentroided for feature extraction and MapAlignerPoseClustering for feature alignment. After alignment, we used FeatureLinkerUnlabeled and TextExporter to obtain the results. When comparing

with MZmine 2 or OpenMS, we considered the Mascot identification results corresponding to MZmine 2 or OpenMS features (mass tolerance: ± 10 ppm, RT tolerance: ± 5 min) as the ground truth, respectively.

The precision formula is:

$$\text{Precision} = \frac{1}{N}\sum_{k=1}^{N}\frac{|A_k \cap G_k|}{|A_k|}$$

The recall formula is:

$$\text{Recall} = \frac{1}{N}\sum_{k=1}^{N}\frac{|A_k \cap G_k|}{|G_k|}$$

N is the sample pair number. A_k is a set containing all the aligned feature pairs in the k_{th} sample pair. G_k is a set containing all the ground truth of the k_{th} sample pair. $|X|$ indicates the number of elements in the set X.

Second, Quandenser[32], as an alignment method that requires MS/MS information, was also compared with DeepRTAlign on dataset Benchmark-FC. The peptide identification results (MSFragger) were considered as our ground truth.

Third, we compared DeepRTAlign with MaxQuant, MSFragger and DIA-NN for DDA and DIA data analysis, respectively. For DDA data (Benchmark-FC), the peptide identification results (MaxQuant or MSFragger) were considered as the ground truth. For DIA data, we used a single-cell proteomic dataset obtained from 18 HT22 cells without nocodazole treated in Li et al.[22]. All the parameters are kept the same as described in Li et al. We used Dinosaur to extract MS features in each cell and DeepRTAlign to align the features in all 18 cells, compared with the aligned results of DIA-NN with and without MBR function.

Dataset simulation for generalizability evaluation

We further generated 24 simulated datasets with different RT shifts (considered as noise) for each real-world dataset. Here is the dataset simulation procedure. (1) All the features are extracted in each dataset using OpenMS to form an original feature list. Features with charges 2-6 are considered for proteomic data, and features with

charges 1-6 are considered for metabolomic data. (2) An RT shift based on a normal distribution with increasing standard deviations (σ = 0, 0.1, 0.3, 0.5, 0.7, 1, 3, 5) for each mean value (μ = 0, 5 and 10 minutes) are added in each feature to form a new feature list by modifying the featureXML file generated by OpenMS. (3) The new feature list with artificial RT shifts is aligned to the original feature list by DeepRTAlign and OpenMS. In theory, each feature with an RT shift in the new feature list should be aligned with the same feature in the original list. (4) The precision and recall values were calculated to evaluate the generalizability boundary of DeepRTAlign.

Data availability

Datasets HCC-T and HCC-N can be downloaded from the iProX database[33] under accession number IPX0000937000 or PXD006512. Datasets UPS2-M and UPS2-Y can be downloaded from the iProX database under the accession number IPX00075500 or PXD008428. Datasets HCC-R, EC-H, AT, SC, MI and CD can be downloaded from the PRIDE database[34] under the accession numbers PXD022881, PXD003881, PXD027546, PXD025634, PXD002838 and PXD002882, respectively. Datasets NCC19, SM1100, MM, SO and GUS can be downloaded from the MetaboLights database[35] under the accession numbers MTBLS1866, MTBLS733, MTBLS5430, MTBLS492 and MTBLS650, respectively. Datasets HCC-R2, Benchmark-FC, Benchmark-QC-H, Benchmark-QC-E, and Benchmark-RT can be downloaded from the iProX database under the accession numbers IPX0006622000, IPX0006638000, IPX0006819000, IPX0006819000 and IPX0006820000, respectively. All other relevant data supporting the key findings of this study are available within the article or the Supplementary Information files. Source data are provided in this paper.

Code availability

All the codes are programmed in Python v3.7.4. Numpy v1.16.4 is used to preprocess data. SciPy v1.6.2 is used for statistical methods. Pytorch framework v1.8.0 is used to implement the DNN model, and scikit-learn framework v0.21.3 is

used to implement other machine learning algorithms. Mrmr-selection v0.2.3 is used for mRMR algorithm. Other tools used in the study are MSFragger v3.7 (in FragPipe v19.1), Mascot v2.8.1, MaxQuant v1.6, MZmine 2 v2.53, OpenMS v2.6.0, Quandenser v0.03.

Results and Discussions

Model evaluation on the training set and the test sets

To optimize the network parameters in the DNN model, the best parameters were selected based on the 10-fold cross-validation results on the training set (HCC-T) (**Supplementary Table 2**). Then, to benchmark the DNN model, we additionally trained four models using several popular machine learning methods (RF, KNN, SVM and LR) on the same training set (HCC-T). The test results of our DNN model and all the other machine-learning models are shown in **Supplementary Table 3**. We found that our DNN model owned the highest AUC compared with other models. Although the DNN model is trained on the HCC-T dataset, it achieved good generalizability and can be applied to other datasets with different sample types or species.

Ablation analysis

We performed the ablation analysis to evaluate the DNN model in DeepRTAlign on different test sets (HCC-N, HCC-R, UPS2-M, UPS2-Y, EC-H, AT and SC). For the coarse alignment step, we have shown that there are no obvious differences when using different samples as the anchor sample (**Supplementary Table 9**). We also tested the performance of DeepRTAlign with or without the coarse alignment step (**Supplementary Table 10**). We found that our DNN model with the coarse alignment step owned a higher AUC compared with the same DNN model without coarse alignment.

Then, we evaluated the importance of each feature used in the DeepRTAlign DNN model by replacing them with random values. As shown in **Supplementary Table 11**, replacing RT (column “RT”) with random values obtained a smaller AUC than replacing m/z (column “m/z”), indicating that RT is a more important feature than m/z in this task. Meanwhile, we found the differences in RT and m/z in the samples to be aligned (column “Difference”) are more important than the original RT

and m/z values (column “Original”). Considering all the features together (column “All”) had the smallest AUC, showing that the design of our input vector (**Supplementary Fig. 1**) and the DNN model are reasonable.

Comparison with existing alignment tools

According to the information required, all the alignment tools can be divided into three types (**Supplementary Table 8**). DeepRTAlign, MZmine 2 and OpenMS only need MS information. Quandenser requires both MS and MS/MS information. In addition, MaxQuant, MSFragger and DIA-NN with MBR functions require identification results for alignment.

Comparison with MZmine 2, OpenMS

First, DeepRTAlign was compared with two other popular MS-based only alignment tools MZmine 2 and OpenMS, on proteomic datasets. As shown in **Fig. 2**, we found that DeepRTAlign showed an improvement in both precision and recall compared with OpenMS and MZmine 2. The noise threshold in MZmine 2 is the key parameter associated with the number of extracted features. But it is difficult for users to choose. Lowering the noise threshold will further increase its extraction time. The focus of this work is not to compare the pros and cons of different feature extraction tools, so we chose relatively high noise thresholds (1.0E6 on proteomic data, 1.0E5 on metabolomic data) to ensure the feature quality and control of the running time.

<Figure 2>

Then, we also compared DeepRTAlign with MZmine 2 and OpenMS on a public real-world metabolomic dataset SM1100[24]. This dataset was generated from standard mixtures consisting of 1100 compounds with specified concentration ratios. It contains two groups (SA and SB), and each group has 5 replicates. We used DeepRTAlign (combined with 3 different feature extraction tools: MZmine 2, OpenMS and Dinosaur), MZmine 2 and OpenMS to align features across runs. Thus, it resulted in five different algorithm combinations (**Supplementary Table 12**). Here, to demonstrate DeepRTAlign’s capacity to deal with metabolomic data, every adjacent sample pair in each group (i.e., SA1-SA2, SA2-SA3, SA3-SA4, SA4-SA5 and SB1-SB2, SB2-SB3, SB3-SB4, and SB4-SB5) was aligned using the five algorithm

combinations.

For feature extraction, the default parameters in OpenMS and Dinosaur were used. In MZmine 2, the parameter "Noise Level" was set to 1.0E5 to make the extracted feature number similar to those of OpenMS and Dinosaur. Then, the extracted features in each sample were annotated according to the 1100 standard compounds with strict standards (mass tolerance: ± 5 ppm, RT tolerance: ± 0.5 min as suggested in the paper of Li et al.[24]). Based on the annotation results, the precision and recall values for each combination were calculated and shown in **Supplementary Table 12**. We can see that all five combinations performed well due to the simpler composition of this metabolomic standard dataset (compared with the proteomic datasets).

Comparison with Quandenser

DeepRTAlign was then compared with another popular tool Quandenser[32], which uses both MS and MS/MS information. Quandenser applies unsupervised clustering on both MS1 and MS2 levels to summarize all analytes of interest without assigning identities. Using Dinosaur as the feature extraction method, we compared DeepRTAlign with Quandenser on the Benchmark-FC dataset. We mapped all the extracted features to the identification results and only considered the *E. coli* peptides without missing values in all the replicates. As shown in **Supplementary Fig. 4**, we found that DeepRTAlign is comparable to Quandenser in terms of the number of aligned peptides and the quantification accuracy. But DeepRTAlign can align many more features (no matter if there were corresponding identification results) than Quandenser since it is an ID-free alignment approach.

Comparison with MaxQuant, MSFragger and DIA-NN with or without MBR

We compared DeepRTAlign with the alignment methods based on the identification results, which is currently the most used alignment strategy. MBR is an updated version of these kinds of alignment methods, which can transfer identification results to the un-identification features[36]. We further compared DeepRTAlign with MaxQuant's MBR and MSFragger's MBR on the Benchmark-FC dataset (**Fig. 3**). MaxQuant or MSFragger was run twice with and without the MBR function keeping

the other parameters unchanged.

As shown in **Fig. 3**, we found that the combination of using MaxQuant to extract features, DeepRTAlign to align features and MSFragger to identify their peptide sequences could obtain 150% more peptides compared with the original MaxQuant workflow (using MaxQuant to extract features, align them and identify their peptide sequences) without compromising the quantification accuracy. DeepRTAlign also aligned 7% and 14.5% more peptides than OpenMS and MSFragger's MBR, respectively, while its quantification accuracy was comparable with other methods. Overall, we found that OpenMS extracted features represented a slightly better accuracy, regardless of the alignment methods employed. This may be because OpenMS has strict quality controls in feature extraction. DeepRTAlign can still increase the number of aligned peptides based on OpenMS extracted features, showing a higher sensitivity than OpenMS in terms of alignment. These results demonstrated DeepRTAlign has a better performance for alignment than MBR-applied MaxQuant and MSFragger.

It should be noted that since MSFragger did not provide its extracted feature list, we used the features extracted by Dinosaur, OpenMS and MaxQuant for a fair comparison. DeepRTAlign is compatible with multiple feature extraction methods (Dinosaur, OpenMS, MaxQuant and XICFinder), which is convenient for users to choose the feature extraction method suitable for their experiments.

Furthermore, DeepRTAlign also performed better than MBR in DIA-NN on single-cell proteomics DIA data (**Supplementary Fig. 5**). Considering the aligned peptides present in at least two cells, DeepRTAlign can align 39 (6.33%) more peptides on average than the existing popular tool DIA-NN in each cell. Moreover, using DeepRTAlign, the average number of the aligned features present in at least two cells is approximately 42.3 times the average number of the aligned peptides, providing the possibility to identify different cell types using the aligned MS features besides the aligned peptides in the future.

<Figure 3>

Generalizability evaluation on simulated datasets

Based on the 14 real-world datasets (9 proteomic datasets: EC-H, HCC-N, UPS2-M, UPS2-Y, HCC-R, AT, MI, SC and CD; 5 metabolomic datasets: NCC19, SM1100, MM, SO and GUS), we generated a total of 336 (14*24) simulated datasets with different RT shifts to evaluate the generalizability boundary of DeepRTAlign.

As shown in **Fig. 4**, it can be found that in most cases, DeepRTAlign owns higher precision and recall values than OpenMS on the proteomic datasets. Meanwhile, we also find DeepRTAlign and OpenMS perform worse when the standard deviation of the RT shift increases (from 0.1 to 5). Thus, we recommend that the RT shift distribution be controlled to different levels in practical applications for proteomic and metabolomic studies. In most proteomic datasets, when the standard deviation of RT shift is larger than 1 min, the precision and recall drop significantly (**Fig. 4**). In most metabolomic datasets, a similar phenomenon occurs when the standard deviation is larger than 0.3 min, especially in recall (**Supplementary Fig. 6-7**). Under the FDR<1%, DeepRTAlign outperformed OpenMS on two metabolomic datasets (SM1100 and MM) but not on other datasets (NCC19, SO and GUS). We explored the extracted features in datasets NCC19, SO and GUS and found that they contained a large number of features with very close m/z. One possible reason is this type of data is not compatible with our current decoy design method. Thus, we further tested DeepRTAlign on these three datasets by setting FDR threshold to 100%. As shown in **Supplementary Fig. 7**, we found that the performance of DeepRTAlign became comparable with that of OpenMS on the three datasets. However, it should be noted that OpenMS does not perform well on these three datasets either, indicating a potential limitation for MS1-only alignment method. Meanwhile, we found that both proteomic and metabolomic datasets are unaffected when only changing the mean of RT shifts.

<Figure 4>

MS features aligned by DeepRTAlign enable accurate prediction of HCC early recurrence

The accurate detection of HCC early recurrence after surgery is one of the main

challenges in liver cancer research[37]. In our previous work[38], MS features were proved to have a good discriminating power to classify tumor and non-tumor samples.

Here, we applied DeepRTAlign to re-analyze the proteomics data of the tumor samples from 101 HCC patients in Jiang et al.'s paper[17] and used the aligned MS features to train a classifier for predicting HCC early recurrence (**Fig. 5a**). For all the 101 patients in Jiang et al.'s paper, we define the recurrent patients within 24 months after surgery as the early recurrences (N=30), and the non-recurrent patients longer than 60 months after surgery as late recurrences (N=22). Two reference classifiers for recurrence prediction were also built based on the quantified peptides or proteins. We used the same workflow for a fair evaluation of the three classifiers based on MS features, peptides and proteins, respectively: (1) Perform minimum Redundancy Maximum Relevance (mRMR) algorithm[39] between the early recurrence group and late recurrence group. (2) Choose the top 200 MS features, peptides, and proteins with the smallest p-value as input to train an SVM classifier, respectively. All the parameters in SVM were set by default using the scikit-learn package (kernel='rbf', gamma='auto', C=1). As shown in **Fig. 5b**, the five-fold cross validation of the top 200 aligned features-based classifier on the training set (C1, N=52) shows the highest AUC (0.998) compared with those classifiers built on top 200 peptides (AUC: 0.931) and top 200 proteins (AUC: 0.757). Please note that in the 101 patients from dataset HCC-T, the cases of early recurrence (N=30), and the cases of late recurrence (N=22) were considered as the training set (C1) for HCC early recurrence prediction.

Next, we mapped the top 200 MS features, peptides and proteins selected by mRMR to a test set HCC-R (C2, N=11), respectively. There were 15 features, 55 peptides and 56 proteins successfully mapped in HCC-R. Then, the mapped 15 features were used to train a SVM classifier on the HCC-T dataset. To keep the same criteria, the top 15 of the mapped peptides and proteins were used to train the corresponding SVM classifiers in the same way, respectively. Notably, the feature-based classifier achieved better results than peptide- and protein-based classifiers on HCC-R (**Fig. 5c**). We tried to map all the 15 MS features in the feature-based early recurrence classifier to the corresponding identification results. There were eight

features successfully mapped to the identification results of the HCC-T dataset. Seven features left remained unknown (**Supplementary Table 13**).

To further validate the 15 MS features in the feature-based early recurrence classifier, we enrolled a new HCC cohort HCC-R2 (C3, N=23, **Supplementary Table 14**) and performed scheduled PRM (sPRM)[40] experiments for validation. Those 23 HCC patient samples were not fractionated, allowing us to detect all targeted features in a single run. The 15 features and the other 34 features from the top 200 features identified from our training set (HCC-T) were tentatively included in the target list of our sPRM method (**Supplementary Table 15**). Remarkably, all the 15 features were reliably identified in Proteome Discoverer (PD) results and verified in Skyline results (**Supplementary Table 16** and **Supplementary File 1**), including the seven previously unknown features in the HCC-T dataset (**Supplementary Table 13**).

Then, we performed an independent test using the sPRM results of the 15 features on the HCC-R2 cohort (N=23, 12 cases of early recurrence and 11 cases of late recurrence). Peak areas of precursors and transitions were exported from Skyline (**Supplementary Table 17**) and summed as peptide abundance for the 15 features. As shown in **Fig. 5d**, the classifier to predict HCC early recurrence achieved an AUC value of 0.833 in this independent cohort (HCC-R2), indicating the generalizability of the 15-feature-based early recurrence classifier. These results indicate that the aligned MS features may contain more hidden information than peptides and proteins and thus enable more accurate stratification of the patients, showing the advantage of the ID-free alignment.

<Figure 5>

Conclusions

In summary, we present a deep learning-based tool DeepRTAlign for RT alignment in large cohort proteomic and metabolomic data analysis. DeepRTAlign is based on the basic information of MS spectra (m/z, RT and intensity), which can be applied to all the precursor ions in MS data before identification. We have demonstrated that DeepRTAlign outperformed other existing alignment tools by

aligning more corresponding features without compromising the quantification precision and determined its generalizability boundary on multiple proteomic and metabolomic datasets. DeepRTAlign is flexible and robust with different feature extraction tools, which can help researchers achieve accurate and reproducible research data. Finally, we applied DeepRTAlign to HCC early recurrence prediction (**Supplementary Notes**) as a real-world example. The results showed that aligned MS features have effective information compared with peptides and proteins. DeepRTAlign is expected to be useful in finding low-abundant biomarkers, which usually only have low-quality MS/MS spectra, and may play a key role in proteomics-driven precision medicine.

Acknowledgment

We thank Dr. Jing Yang, Dr. Mingchao Wang and Dr. Chao Liu for the helpful suggestions. This work has been supported by the National Key Research and Development Program of China (2021YFA1301603 and 2020YFE0202200), the National Natural Science Foundation of China (32088101), and the CAMS Innovation Fund for Medical Sciences (CIFMS) (2019-I2M-5-063).

Author Contributions

Y.Z. and C.C. designed the study and co-supervised the project. Y.L. designed the architecture of DeepRTAlign, performed the experiments, analyzed the results. Y.Y. designed and performed the PRM experiments and analyzed the PRM data. Y.L., Y.Y. and C.C. wrote the manuscript. F.S. enrolled the new HCC cohort (N=23) for PRM validation and provided the necessary clinical information. W.C. generated the benchmark dataset with known fold changes. Yingying.Z. and Yuanjun.Z. collected the public datasets. L.X. helped to revise the manuscript. F.H. coordinated the study. All authors helped to revise the manuscript and approved the final manuscript.

Competing Interests

The authors declare no competing interests.

References

(1) Bouwmeester, R.; Gabriels, R.; Hulstaert, N.; Martens, L.; Degroeve, S. DeepLC Can Predict Retention Times for Peptides That Carry As-yet Unseen Modifications. *Nat Methods* **2021**, *18* (11), 1363–1369. https://doi.org/10.1038/s41592-021-01301-5.

(2) Smith, R.; Ventura, D.; Prince, J. T. LC-MS Alignment in Theory and Practice: A Comprehensive Algorithmic Review. *Briefings in Bioinformatics* **2013**, *16* (1), 104–117. https://doi.org/10.1093/bib/bbt080.

(3) Fernández-Costa, C.; Martínez-Bartolomé, S.; McClatchy, D. B.; Saviola, A. J.; Yu, N.-K.; Yates, J. R. Impact of the Identification Strategy on the Reproducibility of the DDA and DIA Results. *J. Proteome Res.* **2020**, *19* (8), 3153–3161. https://doi.org/10.1021/acs.jproteome.0c00153.

(4) Tyanova, S.; Temu, T.; Cox, J. The MaxQuant Computational Platform for Mass Spectrometry-Based Shotgun Proteomics. *Nat Protoc* **2016**, *11* (12), 2301–2319. https://doi.org/10.1038/nprot.2016.136.

(5) Chang, C.; Li, M.; Guo, C.; Ding, Y.; Xu, K.; Han, M.; He, F.; Zhu, Y. PANDA: A Comprehensive and Flexible Tool for Quantitative Proteomics Data Analysis. *Bioinformatics* **2019**, *35* (5), 898–900. https://doi.org/10.1093/bioinformatics/bty727.

(6) Kong, A. T.; Leprevost, F. V.; Avtonomov, D. M.; Mellacheruvu, D.; Nesvizhskii, A. I. MSFragger: Ultrafast and Comprehensive Peptide Identification in Mass Spectrometry–Based Proteomics. *Nat Methods* **2017**, *14* (5), 513–520. https://doi.org/10.1038/nmeth.4256.

(7) Yu, F.; Haynes, S. E.; Nesvizhskii, A. I. IonQuant Enables Accurate and Sensitive Label-Free Quantification With FDR-Controlled Match-Between-Runs. *Molecular & Cellular Proteomics* **2021**, *20*, 100077. https://doi.org/10.1016/j.mcpro.2021.100077.

(8) Demichev, V.; Messner, C. B.; Vernardis, S. I.; Lilley, K. S.; Ralser, M. DIA-NN: Neural Networks and Interference Correction Enable Deep Proteome Coverage in High Throughput. *Nat Methods* **2020**, *17* (1), 41–44. https://doi.org/10.1038/s41592-019-0638-x.

(9) Mitra, V.; Smilde, A. K.; Bischoff, R.; Horvatovich, P. Tutorial: Correction of Shifts in Single-Stage LC-MS(/MS) Data. *Analytica Chimica Acta* **2018**, *999*, 37–53. https://doi.org/10.1016/j.aca.2017.09.039.

(10) Smith, C. A.; Want, E. J.; O'Maille, G.; Abagyan, R.; Siuzdak, G. XCMS: Processing Mass Spectrometry Data for Metabolite Profiling Using Nonlinear Peak Alignment, Matching, and Identification. *Anal. Chem.* **2006**, *78* (3), 779–787. https://doi.org/10.1021/ac051437y.

(11) Pluskal, T.; Castillo, S.; Villar-Briones, A.; Orešič, M. MZmine 2: Modular Framework for Processing, Visualizing, and Analyzing Mass Spectrometry-Based Molecular Profile Data. *BMC Bioinformatics* **2010**, *11* (1), 395. https://doi.org/10.1186/1471-2105-11-395.

(12) Lange, E.; Gröpl, C.; Schulz-Trieglaff, O.; Leinenbach, A.; Huber, C.; Reinert, K.

A Geometric Approach for the Alignment of Liquid Chromatography—Mass Spectrometry Data. *Bioinformatics* **2007**, *23* (13), i273–i281. https://doi.org/10.1093/bioinformatics/btm209.

(13) Duran, A. L.; Yang, J.; Wang, L.; Sumner, L. W. Metabolomics Spectral Formatting, Alignment and Conversion Tools (MSFACTs). *Bioinformatics* **2003**, *19* (17), 2283–2293. https://doi.org/10.1093/bioinformatics/btg315.

(14) Ballardini, R.; Benevento, M.; Arrigoni, G.; Pattini, L.; Roda, A. MassUntangler: A Novel Alignment Tool for Label-Free Liquid Chromatography–Mass Spectrometry Proteomic Data. *Journal of Chromatography A* **2011**, *1218* (49), 8859–8868. https://doi.org/10.1016/j.chroma.2011.06.062.

(15) Johnson, K. J.; Wright, B. W.; Jarman, K. H.; Synovec, R. E. High-Speed Peak Matching Algorithm for Retention Time Alignment of Gas Chromatographic Data for Chemometric Analysis. *Journal of Chromatography A* **2003**, *996* (1–2), 141–155. https://doi.org/10.1016/S0021-9673(03)00616-2.

(16) Li, M.; Wang, X. R. Peak Alignment of Gas Chromatography–Mass Spectrometry Data with Deep Learning. *Journal of Chromatography A* **2019**, *1604*, 460–476. https://doi.org/10.1016/j.chroma.2019.460476.

(17) Jiang, Y.; Sun, A.; Zhao, Y.; Ying, W.; Sun, H.; Yang, X.; Xing, B.; Sun, W.; Ren, L.; Hu, B.; Li, C.; Zhang, L.; Qin, G.; Zhang, M.; Chen, N.; Zhang, M.; Huang, Y.; Zhou, J.; Zhao, Y.; Liu, M.; Zhu, X.; Qiu, Y.; Sun, Y.; Huang, C.; Yan, M.; Wang, M.; Liu, W.; Tian, F.; Xu, H.; Zhou, J.; Wu, Z.; Shi, T.; Zhu, W.; Qin, J.; Xie, L.; Fan, J.; Qian, X.; He, F.; Chinese Human Proteome Project (CNHPP) Consortium. Proteomics Identifies New Therapeutic Targets of Early-Stage Hepatocellular Carcinoma. *Nature* **2019**, *567* (7747), 257–261. https://doi.org/10.1038/s41586-019-0987-8.

(18) Bhat, M.; Clotet-Freixas, S.; Baciu, C.; Pasini, E.; Hammad, A.; Ivanics, T.; Reid, S.; Azhie, A.; Angeli, M.; Ghanekar, A.; Fischer, S.; Sapisochin, G.; Konvalinka, A. Combined Proteomic/Transcriptomic Signature of Recurrence Post-Liver Transplantation for Hepatocellular Carcinoma beyond Milan. *Clin Proteom* **2021**, *18* (1), 27. https://doi.org/10.1186/s12014-021-09333-x.

(19) Chang, C.; Zhang, J.; Xu, C.; Zhao, Y.; Ma, J.; Chen, T.; He, F.; Xie, H.; Zhu, Y. Quantitative and In-Depth Survey of the Isotopic Abundance Distribution Errors in Shotgun Proteomics. *Anal Chem* **2016**, *88* (13), 6844–6851. https://doi.org/10.1021/acs.analchem.6b01409.

(20) Shen, X.; Shen, S.; Li, J.; Hu, Q.; Nie, L.; Tu, C.; Wang, X.; Poulsen, D. J.; Orsburn, B. C.; Wang, J.; Qu, J. IonStar Enables High-Precision, Low-Missing-Data Proteomics Quantification in Large Biological Cohorts. *Proc Natl Acad Sci USA* **2018**, *115* (21), E4767–E4776. https://doi.org/10.1073/pnas.1800541115.

(21) Ginsawaeng, O.; Gorka, M.; Erban, A.; Heise, C.; Brueckner, F.; Hoefgen, R.; Kopka, J.; Skirycz, A.; Hincha, D. K.; Zuther, E. Characterization of the Heat-Stable Proteome during Seed Germination in Arabidopsis with Special Focus on LEA Proteins. *IJMS* **2021**, *22* (15), 8172. https://doi.org/10.3390/ijms22158172.

(22) Li, Y.; Li, H.; Xie, Y.; Chen, S.; Qin, R.; Dong, H.; Yu, Y.; Wang, J.; Qian, X.; Qin, W. An Integrated Strategy for Mass Spectrometry-Based Multiomics

Analysis of Single Cells. *Anal. Chem.* **2021**, *93* (42), 14059–14067. https://doi.org/10.1021/acs.analchem.0c05209.

(23) Barberis, E.; Timo, S.; Amede, E.; Vanella, V. V.; Puricelli, C.; Cappellano, G.; Raineri, D.; Cittone, M. G.; Rizzi, E.; Pedrinelli, A. R.; Vassia, V.; Casciaro, F. G.; Priora, S.; Nerici, I.; Galbiati, A.; Hayden, E.; Falasca, M.; Vaschetto, R.; Sainaghi, P. P.; Dianzani, U.; Rolla, R.; Chiocchetti, A.; Baldanzi, G.; Marengo, E.; Manfredi, M. Large-Scale Plasma Analysis Revealed New Mechanisms and Molecules Associated with the Host Response to SARS-CoV-2. *IJMS* **2020**, *21* (22), 8623. https://doi.org/10.3390/ijms21228623.

(24) Li, Z.; Lu, Y.; Guo, Y.; Cao, H.; Wang, Q.; Shui, W. Comprehensive Evaluation of Untargeted Metabolomics Data Processing Software in Feature Detection, Quantification and Discriminating Marker Selection. *Analytica Chimica Acta* **2018**, *1029*, 50–57. https://doi.org/10.1016/j.aca.2018.05.001.

(25) Wase, N.; Gutiérrez, J. M.; Rucavado, A.; Fox, J. W. Longitudinal Metabolomics and Lipidomics Analyses Reveal Alterations Associated with Envenoming by Bothrops Asper and Daboia Russelii in an Experimental Murine Model. *Toxins* **2022**, *14* (10), 657. https://doi.org/10.3390/toxins14100657.

(26) Swenson, T. L.; Karaoz, U.; Swenson, J. M.; Bowen, B. P.; Northen, T. R. Linking Soil Biology and Chemistry in Biological Soil Crust Using Isolate Exometabolomics. *Nat Commun* **2018**, *9* (1), 19. https://doi.org/10.1038/s41467-017-02356-9.

(27) Gibson, C. L.; Codreanu, S. G.; Schrimpe-Rutledge, A. C.; Retzlaff, C. L.; Wright, J.; Mortlock, D. P.; Sherrod, S. D.; McLean, J. A.; Blakely, R. D. Global Untargeted Serum Metabolomic Analyses Nominate Metabolic Pathways Responsive to Loss of Expression of the Orphan Metallo β-Lactamase, MBLAC1. *Mol. Omics* **2018**, *14* (3), 142–155. https://doi.org/10.1039/C7MO00022G.

(28) Lichtman, J. S.; Alsentzer, E.; Jaffe, M.; Sprockett, D.; Masutani, E.; Ikwa, E.; Fragiadakis, G. K.; Clifford, D.; Huang, B. E.; Sonnenburg, J. L.; Huang, K. C.; Elias, J. E. The Effect of Microbial Colonization on the Host Proteome Varies by Gastrointestinal Location. *ISME J* **2016**, *10* (5), 1170–1181. https://doi.org/10.1038/ismej.2015.187.

(29) Mottawea, W.; Chiang, C.-K.; Mühlbauer, M.; Starr, A. E.; Butcher, J.; Abujamel, T.; Deeke, S. A.; Brandel, A.; Zhou, H.; Shokralla, S.; Hajibabaei, M.; Singleton, R.; Benchimol, E. I.; Jobin, C.; Mack, D. R.; Figeys, D.; Stintzi, A. Altered Intestinal Microbiota–Host Mitochondria Crosstalk in New Onset Crohn's Disease. *Nat Commun* **2016**, *7* (1), 13419. https://doi.org/10.1038/ncomms13419.

(30) Teleman, J.; Chawade, A.; Sandin, M.; Levander, F.; Malmström, J. Dinosaur: A Refined Open-Source Peptide MS Feature Detector. *J. Proteome Res.* **2016**, *15* (7), 2143–2151. https://doi.org/10.1021/acs.jproteome.6b00016.

(31) Lange, E.; Tautenhahn, R.; Neumann, S.; Gröpl, C. Critical Assessment of Alignment Procedures for LC-MS Proteomics and Metabolomics Measurements. *BMC Bioinformatics* **2008**, *9* (1), 375. https://doi.org/10.1186/1471-2105-9-375.

(32) The, M.; Lukas, K. Focus on the Spectra That Matter by Clustering of

Quantification Data in Shotgun Proteomics. *Nature Communications* **2020**, *11*, 3234.

(33) Ma, J.; Chen, T.; Wu, S.; Yang, C.; Bai, M.; Shu, K.; Li, K.; Zhang, G.; Jin, Z.; He, F.; Hermjakob, H.; Zhu, Y. IProX: An Integrated Proteome Resource. *Nucleic Acids Res*. **2019**, *47* (D1), D1211–D1217. https://doi.org/10.1093/nar/gky869.

(34) Perez-Riverol, Y.; Csordas, A.; Bai, J.; Bernal-Llinares, M.; Hewapathirana, S.; Kundu, D. J.; Inuganti, A.; Griss, J.; Mayer, G.; Eisenacher, M.; Pérez, E.; Uszkoreit, J.; Pfeuffer, J.; Sachsenberg, T.; Yılmaz, Ş.; Tiwary, S.; Cox, J.; Audain, E.; Walzer, M.; Jarnuczak, A. F.; Ternent, T.; Brazma, A.; Vizcaíno, J. A. The PRIDE Database and Related Tools and Resources in 2019: Improving Support for Quantification Data. *Nucleic Acids Research* **2019**, *47* (D1), D442–D450. https://doi.org/10.1093/nar/gky1106.

(35) Haug, K.; Cochrane, K.; Nainala, V. C.; Williams, M.; Chang, J.; Jayaseelan, K. V.; O'Donovan, C. MetaboLights: A Resource Evolving in Response to the Needs of Its Scientific Community. *Nucleic Acids Research* **2019**, gkz1019. https://doi.org/10.1093/nar/gkz1019.

(36) Lim, M. Y.; Paulo, J. A.; Gygi, S. P. Evaluating False Transfer Rates from the Match-between-Runs Algorithm with a Two-Proteome Model. *J. Proteome Res*. **2019**, *18* (11), 4020–4026. https://doi.org/10.1021/acs.jproteome.9b00492.

(37) Hoshida, Y.; Villanueva, A.; Kobayashi, M.; Peix, J.; Chiang, D. Y.; Camargo, A.; Gupta, S.; Moore, J.; Wrobel, M. J.; Lerner, J.; Reich, M.; Chan, J. A.; Glickman, J. N.; Ikeda, K.; Hashimoto, M.; Watanabe, G.; Daidone, M. G.; Roayaie, S.; Schwartz, M.; Thung, S.; Salvesen, H. B.; Gabriel, S.; Mazzaferro, V.; Bruix, J.; Friedman, S. L.; Kumada, H.; Llovet, J. M.; Golub, T. R. Gene Expression in Fixed Tissues and Outcome in Hepatocellular Carcinoma. *N Engl J Med* **2008**, *359* (19), 1995–2004. https://doi.org/10.1056/NEJMoa0804525.

(38) Dong, H.; Liu, Y.; Zeng, W.; Shu, K.; Zhu, Y.; Chang, C. A Deep Learning□Based Tumor Classifier Directly Using MS Raw Data. *Proteomics* **2020**, *20* (21–22), 1900344. https://doi.org/10.1002/pmic.201900344.

(39) Ding, C.; Peng, H. Minimum Redundancy Feature Selection from Microarray Gene Expression Data. In *Computational Systems Bioinformatics*. *CSB2003. Proceedings of the 2003 IEEE Bioinformatics Conference. CSB2003*; IEEE Comput. Soc: Stanford, CA, USA, 2003; pp 523–528. https://doi.org/10.1109/CSB.2003.1227396.

(40) Van Bentum, M.; Selbach, M. An Introduction to Advanced Targeted Acquisition Methods. *Molecular & Cellular Proteomics* **2021**, *20*, 100165. https://doi.org/10.1016/j.mcpro.2021.100165.

Figure Legends

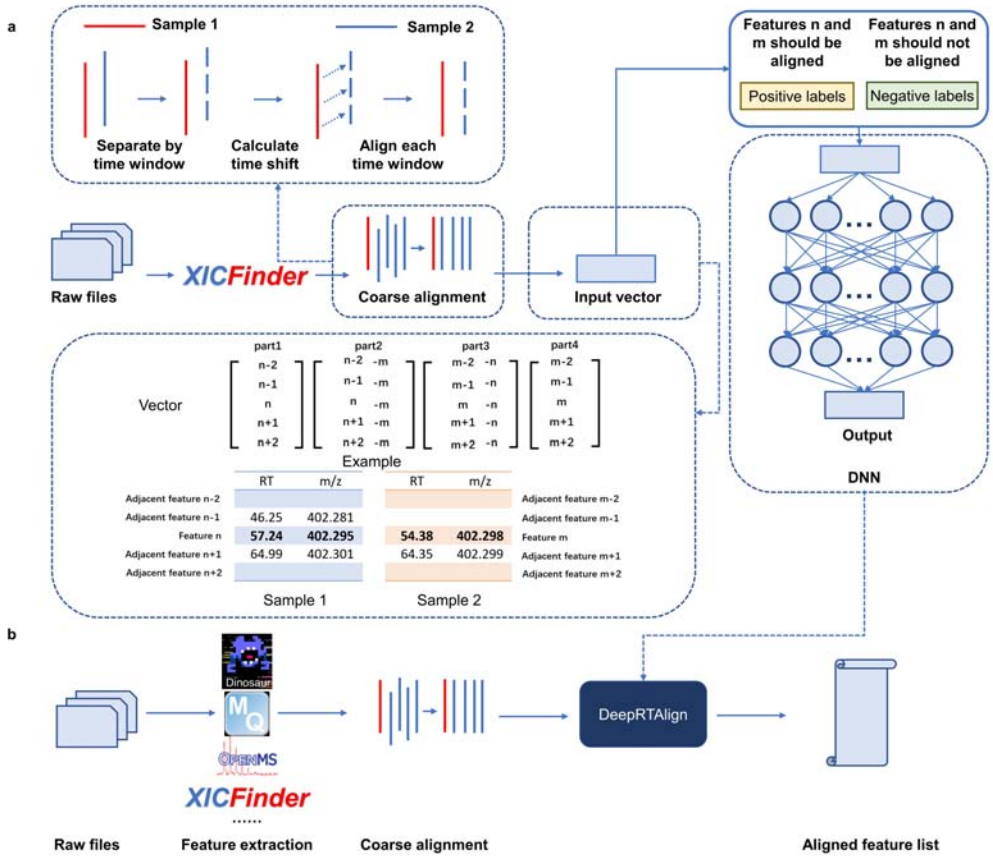

Fig. 1. The illustration of DeepRTAlign algorithm. (a) The training procedures of DeepRTAlign. The n is the feature rank number (sorted by RT) in one m/z window of sample 1. The m is the feature rank number (sorted by RT) in one m/z window of sample 2. Please note that “n-1”, “n-2”, “n+1” and “n+2” represent four adjacent features of the feature n in one m/z window. (b) The workflow for RT alignment using DeepRTAlign. DeepRTAlign supports four feature extraction methods (Dinosaur, MaxQuant, OpenMS and XICFinder). The features extracted will be aligned using the trained model shown in (a), with the aligned feature list as the output.

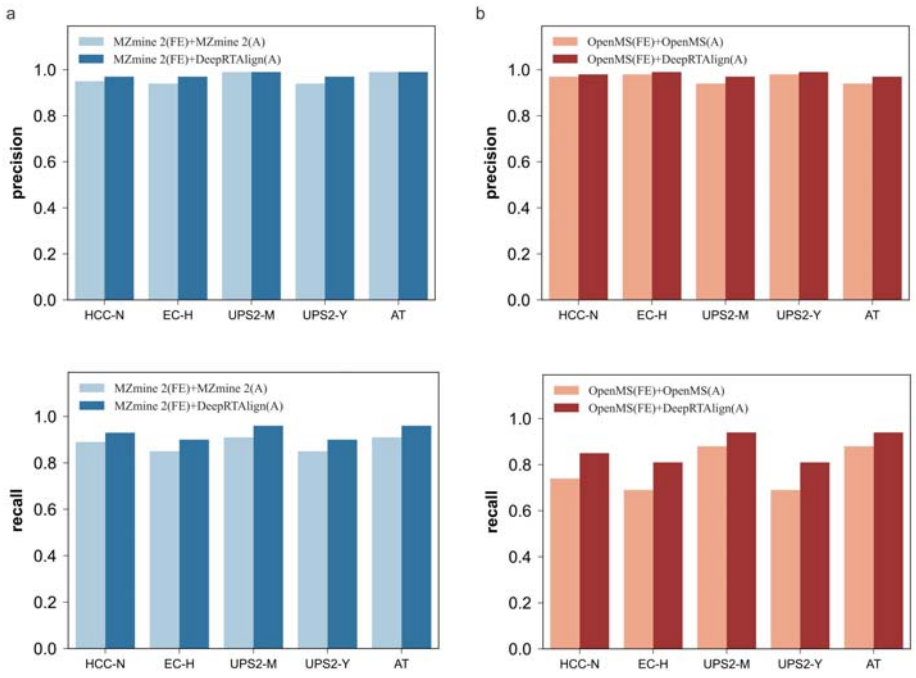

Fig. 2. Performance evaluation of DeepRTAlign compared with MZmine 2 and OpenMS. Panel a shows the precisions and recalls of MZmine 2 and DeepRTAlign on different test sets. Panel b shows the precisions and recalls of OpenMS and DeepRTAlign on different test datasets. “FE” means the feature extraction method. “A” means the RT alignment method. We took the Mascot identification results with FDR<1% as the ground truth in datasets HCC-N, UPS2-M and UPS2-Y and took the MaxQuant identification results with FDR<1% as the ground truth in datasets EC-H and AT. In dataset EC-H, we only considered the *E. coli* peptides for evaluation. In datasets UPS2-M and UPS2-Y, we only considered the UPS2 peptides for evaluation.

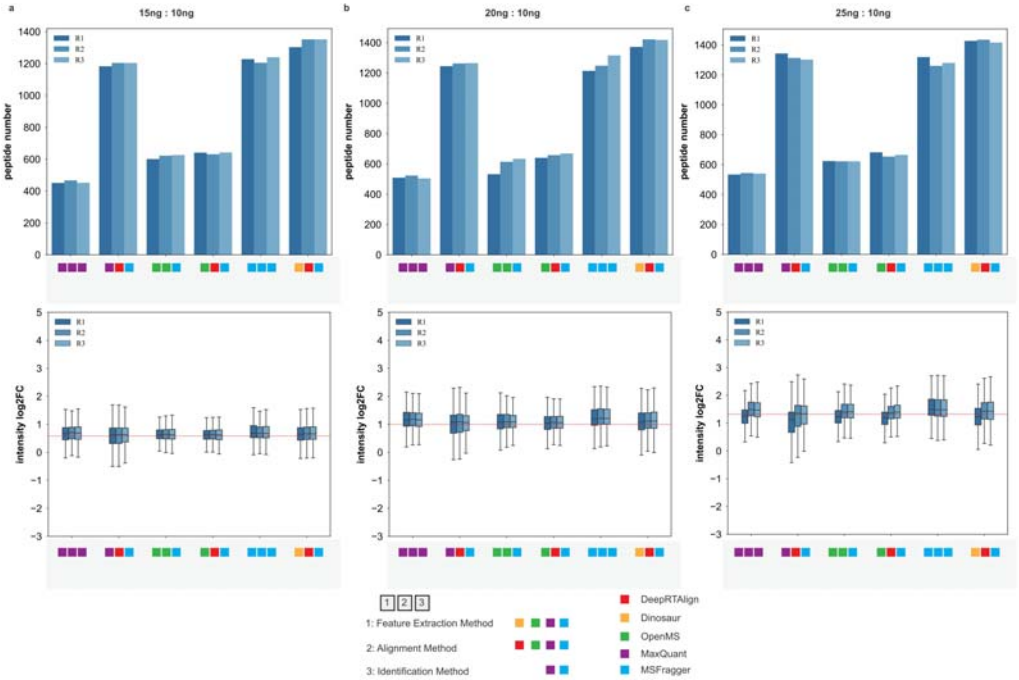

Fig. 3. The number and ratio distributions of all *E. coli* peptides between specific samples (15ng/10ng, 20ng/10ng, and 25ng/10ng) in each replicate (R1, R2 and R3) after alignment. In all boxplots, the center black line is the median of log2 of the peptide fold changes. The box limits are the upper and lower quartiles. The whiskers extend to 1.5 times the size of the interquartile range.

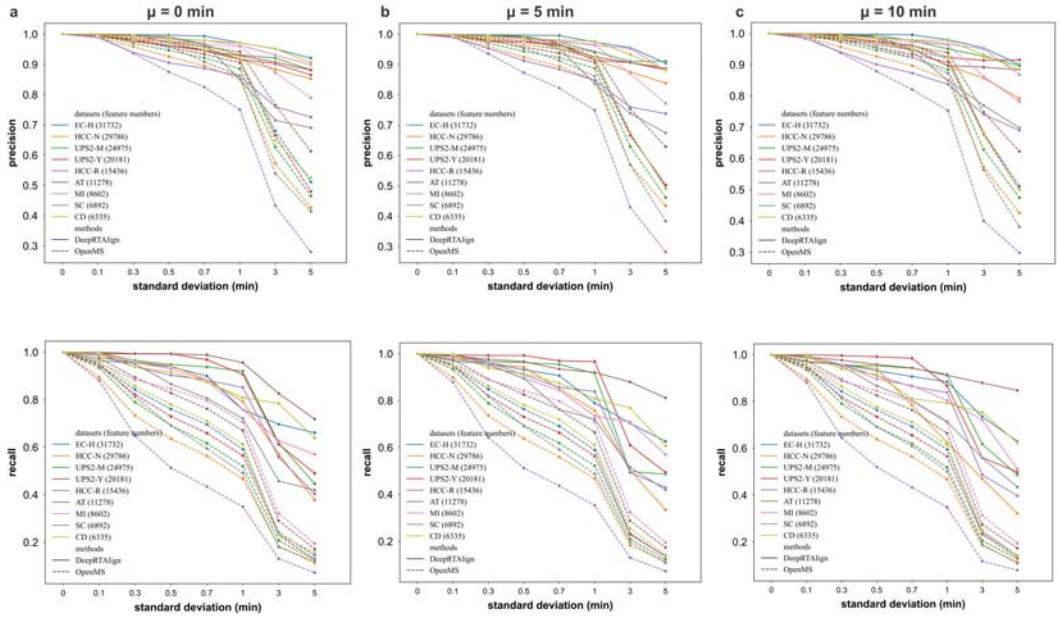

Fig. 4. Comparison of DeepRTAlign and OpenMS on multiple simulated datasets

generated from 9 real-world proteomic datasets. The simulated datasets were constructed by adding normally distributed RT shifts to the corresponding real-world dataset. (a) μ=0 min. (b) μ=5 min. (c) μ=10 min. The normal distribution has an increasing σ, i.e., σ=0, 0.1, 0.3, 0.5, 0.7, 1, 3, 5 for different μ (0, 5 and 10 minutes), respectively.

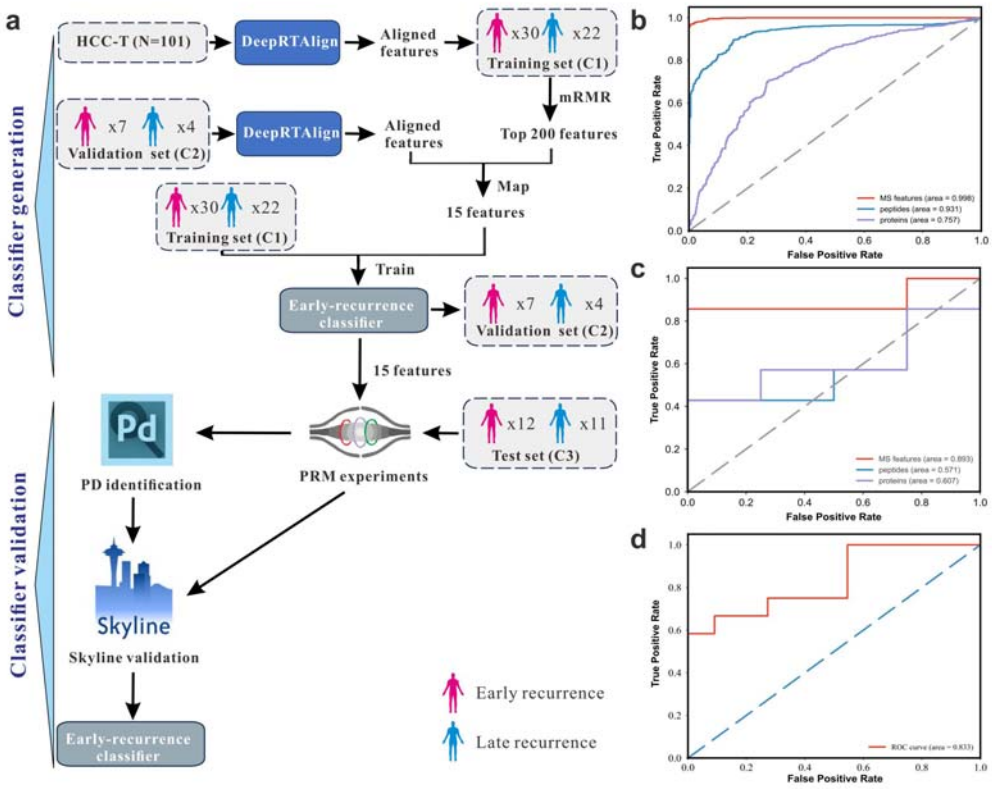

Fig. 5. Construction and validation of an HCC early recurrence classifier based on the MS features aligned by DeepRTAlign. (a) Study design for construction and validation of this 15-feature-based classifier. First, all the features in dataset HCC-T (N=101) were aligned by DeepRTAlign. Top 200 features were selected by mMRM. We mapped the top 200 MS features to an independent test set HCC-R (C2). There were 15 features successfully mapped to HCC-R. In the 101 patients from dataset HCC-T, the cases of early recurrence (N=30), and the cases of late recurrence (N=22) were considered as the training set (C1) for HCC early recurrence prediction. Then, the mapped 15 features were used to train a SVM classifier based on the C1 dataset, and tested on the HCC-R dataset (C2, N=11). Peptide- and protein-based classifiers were generated in the same way, but they were not marked in **Fig.5a** for clarity. These 15 features in the early recurrence classifier were then targeted in PRM experiments and analyzed by PD and Skyline based on an independent dataset HCC-R2 (C3,

N=23). The 15-feature-based classifier was further tested on dataset HCC-R2 (C3) using the Skyline quantification results. (b) The five-fold cross validation AUCs on C1 of the SVM models for HCC recurrence prediction trained by the top 200 aligned MS features, top 200 peptides, and top 200 proteins, respectively. Top 200 peptides and top 200 proteins were also selected by mRMR. (c) The test AUCs on C2 of the SVM models trained on HCC-T dataset by the aligned 15 MS features, top 15 peptides, and top 15 proteins, respectively. (d) The test AUC on C3 of the 15-feature-based SVM model.