

User Manual for DL-NPE



1. Overview

DL-NPE (Deep learning-based natural product extractor) is a software for the analysis of LC-MS/MS data of natural products, written in the Python language. DL-NPE is developed based on deep learning technology and is capable of extracting signals of target natural products from LC-MS/MS data of crude plant extracts or enriched fractions and screening for potentially novel skeletal natural products.

System Requirements: Win 10/11, 64-bit; memory 16GB or above; equipped with an NVIDIA GPU with 6GB or more of video memory. The software has been packaged into an .exe file and can be used by simply opening it (Note: The startup time of the software is relatively long and may take up to 30 seconds, so please be patient). It should be noted that the input for the software must not contain any Chinese characters, otherwise the software will generate errors.

2. Usage

DL-NPE mainly consists of three parts: preparation of the training dataset, training of the deep learning model, and analysis of LC-MS/MS data (Figure 1).

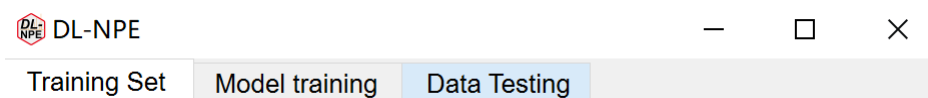


Figure 1

2.1 Preparation of the Dataset

To train the deep learning model, MS/MS spectral data is required. MS/MS spectral data can be obtained from databases such as the GNPS platform (<https://gnps-external.ucsd.edu/gnpslibrary>) and MONA (<https://mona.fiehnlab.ucdavis.edu/>) (Figure 2, Figure 3).

GNPS				
GNPS - Spectral Libraries				
GNPS Library List				
GNPS Library	Library Type	MGF Download	MSP Download	JSON Download
GNPS-LIBRARY	GNPS	Download	Download	Download
GNPS-SELLECKCHEM-FDA-PART1	GNPS	Download	Download	Download
GNPS-SELLECKCHEM-FDA-PART2	GNPS	Download	Download	Download
GNPS-PRESTWICKPHYTOCHEM	GNPS	Download	Download	Download

Figure 2

MoNA - MassBank of North America | Spectra | Downloads | Upload | Help | Search...

Downloads

A set of commonly referenced predefined queries. Clicking the name of the query will display the associated spectra in the query browser. Each query is also available to download in either the MoNA internal show JSON format or as NIST MS Search compatible MSP files.

☐ Display Hidden Downloads

All Spectra (2,042,608 spectra)	Download
In-Silico Spectra (497,555 spectra)	Download
Experimental Spectra (1,545,053 spectra)	Download
GC-MS Spectra (18,902 spectra)	Download

Figure 3

DL-NPE supports reading of .mgf and .json files. On the GNPS platform, datasets in .mgf format can be directly downloaded, while datasets in .json format are available for download from the MONA database. Before constructing the dataset, it is necessary to modify the information in the .mgf file. Open the .mgf file using a text editor such as Notepad (Figure 4, Figure 5). If the .mgf file is larger than 1 GB, other text document readers, such as UltraEdit, should be used to open the file. Replace 'SPECTRUMID' with 'TITLE' and save the file again.

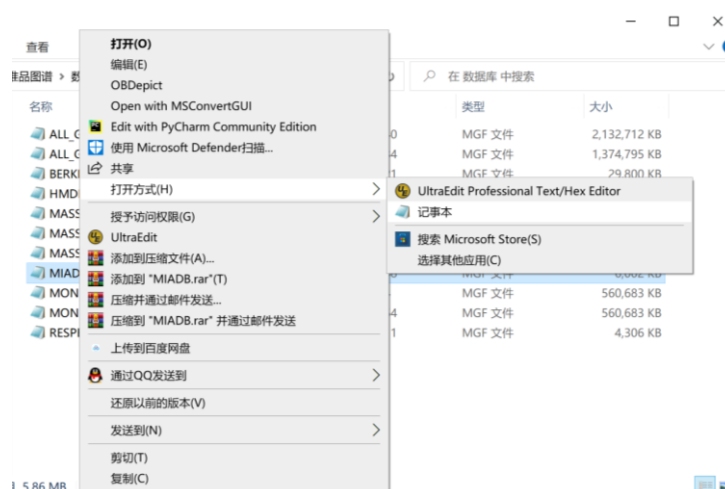


Figure 4

```

BEGIN IONS
PEPMASS= 313.191
CHARGE= 1
MSLEVEL= 2
SOURCE_INSTRUMENT=LC-ESI-qToF
FILENAME=10_hydroxygeissoschizol.mgf
SEQ=.*
IONMODE=Positive
ORGANISM=MIADB
NAME=10-hydroxygeissoschizol M+H
PI=Mehdi Benidzir
DATACOLLECTOR=Alexander Fox Ramos
SMILES=OC1=CC=C(CN2=C3CNC4C2CC(CCO)(C/C4)=C/C/C3=C1
INCHI=InChI=1/C19H24N2O2/C1-2-12-11-21-7-5-15-16-10-14/23(3-4-17)(16-20-19)15(18)21(19-13)(12)6-8-22/H2,4-10,18H
INCHIX=U/N/A
PUBMED=N/A
SUBMITUSER=mwang87
LIBRARYQUALITY=3
TITLE CMLSIB000004679916
SCANS

```

Figure 5

Preparation of the MS/MS dataset for the target compounds (standards). The raw data is first converted into mzML or mzXML files using MSconvert. After importing the files, set the Peak Picking parameters and then start the conversion (Figure 6).

Filters

MS levels:

Scan number:

Scan time (seconds)

Scan event:

Scan polarity

Subset

Charge State Predictor

Demultiplex

ETD Peak Filter

Lockmass Refiner

Peak Picking

Threshold Peak Filter

Scan Summing

Subset

Zero Samples

states:

points:

on type

type

Add

Remove

Filter	Parameters
peakPicking	vendor msLevel=1-2
titleMaker	<RunId>. <ScanNumber>. <ScanNumber>. <ChargeState> File:"<SourcePat...

Save Preset

Start

Filters

Peak Picking

Algorithm:

Vendor (does not work for UNIFI, and it MUST be th

MS Levels:

Min SNR:

Min peak spacing:

1

2

0.1

0.1

Add

Remove

Filter	Parameters
peakPicking	vendor msLevel=1-2
titleMaker	<RunId>. <ScanNumber>. <ScanNumber>. <ChargeState> File:"<SourcePat...

Save Preset

Start

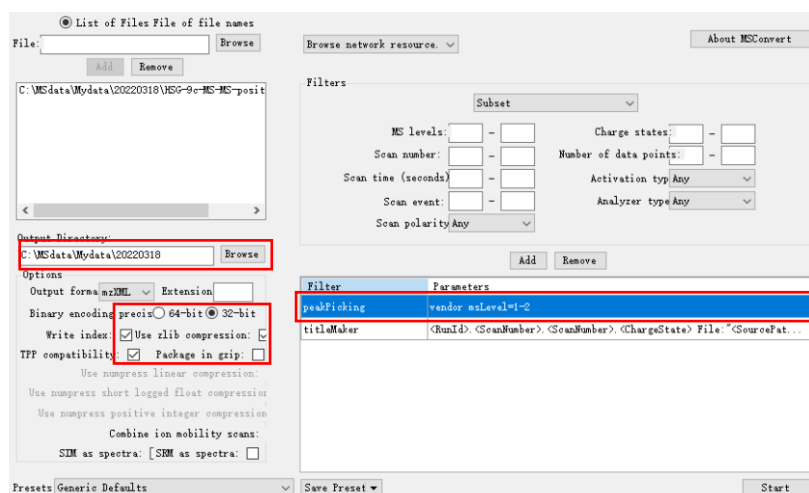
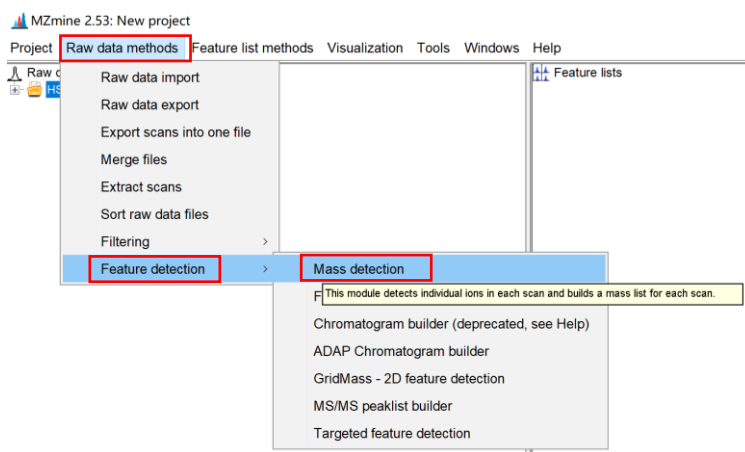


Figure 6

The obtained mzML or mzXML files are imported into MZmine 2 for processing. First, detect the MS/MS data (Figure 7). In the list, locate the target compounds based on the precursor ion mass, and double-click to open the MS/MS spectra (Figure 8). Export the .mgf file. When collecting the MS/MS spectra of reference standards, spectra obtained under different collision energies can be collected simultaneously (Figure 9). Spectra collected at different time points can also be gathered to expand the capacity of the in-house database. DL-NPE is applicable to various types of compounds. Therefore, when collecting data for reference standards, it is necessary to classify the compounds by type and place different types of compounds into their respective folders.



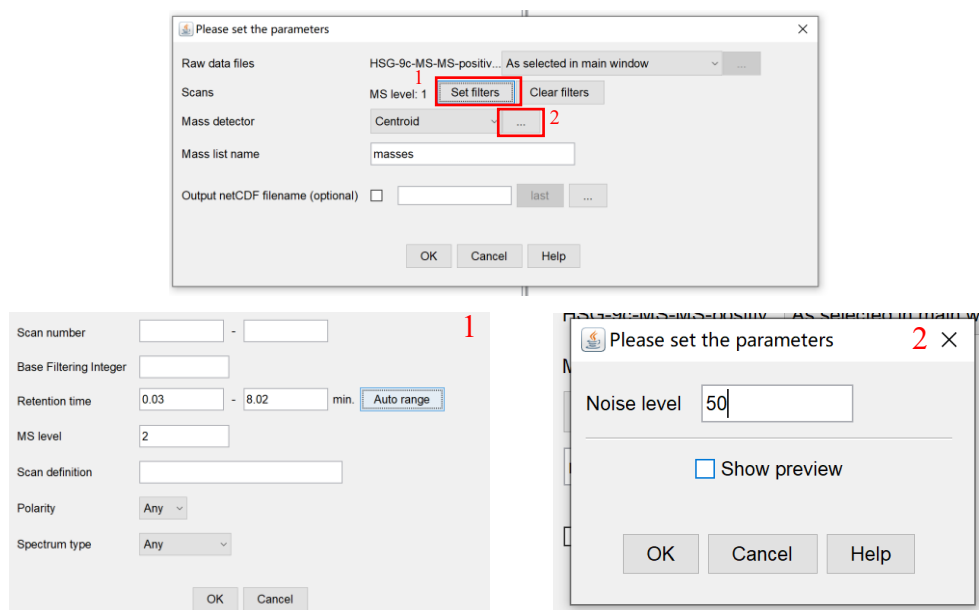


Figure 7

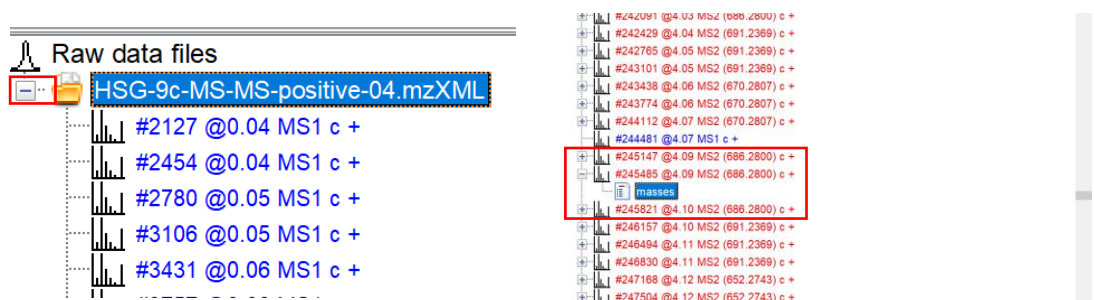


Figure 8

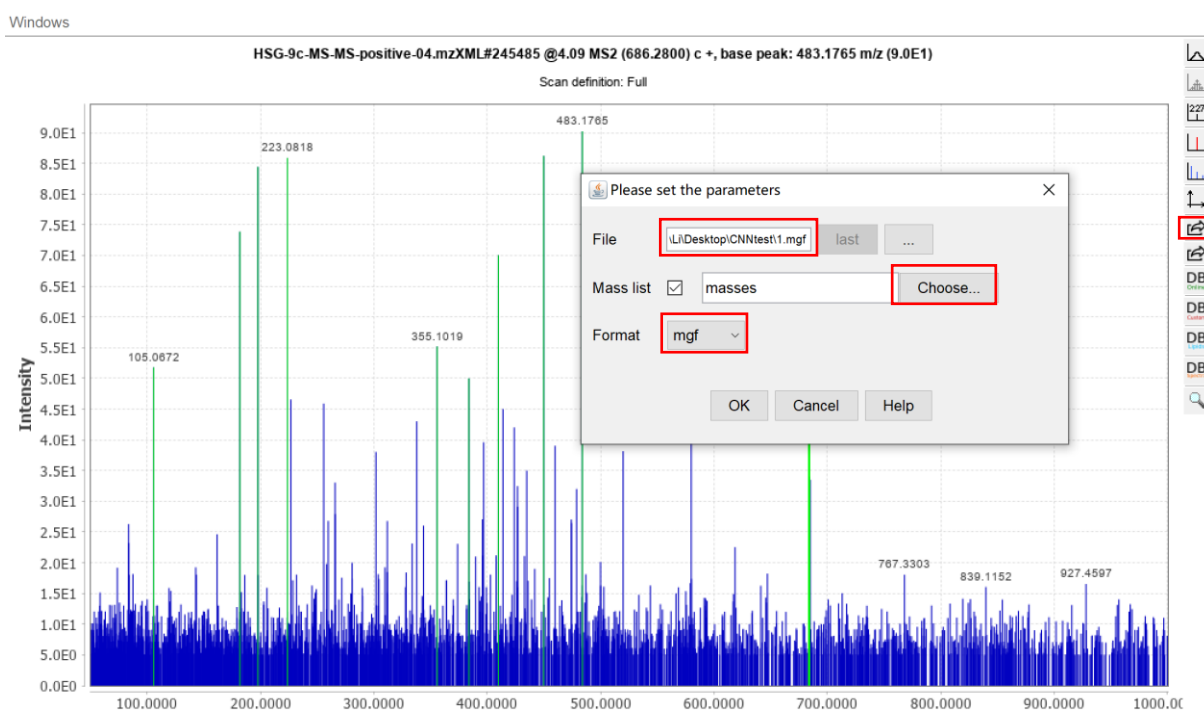


Figure 9

After collecting the MS/MS spectra data of the reference standards, it is necessary to further merge all the .mgf files into a single file. First, place all the .mgf files in the same folder. Open the Command Prompt by pressing Win+R and typing cmd (Figure 10). Use the cd command to navigate to the folder containing the .mgf dataset, and then enter the command to merge the .mgf files (type *.mgf>>C:\Users\Li\Desktop\Reference_Spectra\Various_Collision_Energies\merge.mgf) (Figure 11). After merging, replace 'Title:' with 'TITLE=' in the .mgf file (Figure 12).

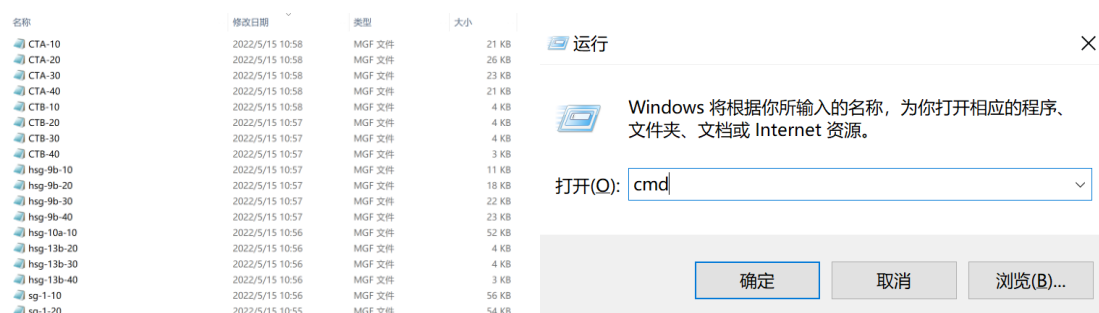


Figure 10

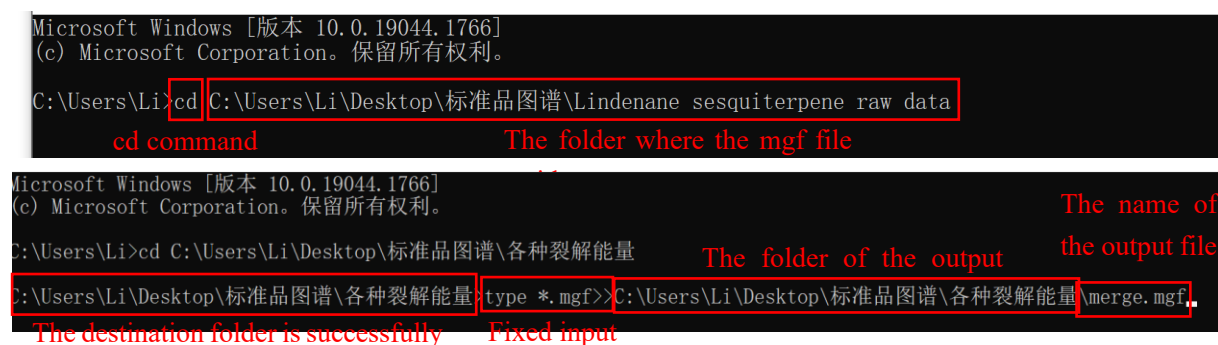


Figure 11

```

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
BEGIN IONS
PEPMASS=686.280029296875
CHARGE=1
MSLEVEL=2
Title: Scan#: 245485, RT: 4.091283333333333 min
105.06717681884766 51.78125
181.09658813476562 73.875
197.05638122558594 84.42045593261719
223.081787109375 85.875
355.1018981933594 55.18560791015625
383.1610107421875 50.0
END IONS

```

Figure 12

2.2 Transformation and Augmentation of the Dataset

Before proceeding with data processing, the first step is to categorize the different types of compounds and label them using numbers such as 0, 1, 2, 3, etc. In our research case, non-

lindenane sesquiterpenoids are labeled as 0 (we suggest labeling non-target compound data as 0), lindenane sesquiterpenoid monomers as 1, dimers as 2, and trimers as 3. Create empty folders according to the predefined labels for the next step of storing MS/MS spectra (Figure 13).

CNNtest > program_test > database

在 database 中搜索

名称	修改日期	类型	大小
0	2022/7/12 17:49	文件夹	
1	2022/7/12 17:49	文件夹	
2	2022/7/12 17:49	文件夹	
3	2022/7/12 17:49	文件夹	

Figure 13. Create empty folders named 0, 1, 2, and 3 within the newly established database folder.

The dataset for deep learning in DL-NPE is in the form of images, so it is necessary to generate corresponding MS/MS images based on the .mgf files. First, use the Build MS/MS figures module to construct the original MS/MS images. The mgf file field requires the full path of the .mgf file (e.g., C:\Users\Li\Desktop\CNNtest\test\CTA.mgf), and the Output pathway should be the location of the folder where the images will be output (e.g., C:\Users\Li\Desktop\CNNtest\program_test\database\0). It should be noted that the Run button should only be clicked once; repeated clicking will cause the program to run a second round after completing the first. Additionally, to determine whether the program has finished its task, observe whether the number of items in the folder continues to increase. (Figure 14, Figure 15)

Build MS/MS figures

mgf file

Output pathway

Run

Figure 14



Figure 15

Dataset Augmentation. MS/MS datasets downloaded from public databases are generally considered to be non-target compound datasets. Since these datasets have a sufficient quantity, further augmentation is not required. In contrast, our reference standard datasets typically consist of only a few hundred spectra, necessitating further expansion. DL-NPE incorporates three methods for augmentation: Data augmentation, Data augmentation (Relative), and Data augmentation (Absolute) (Figure 16). The principle of these augmentation methods is to randomly increase or decrease the intensity of each filtered fragment by 10% to 40%.

Figure 16

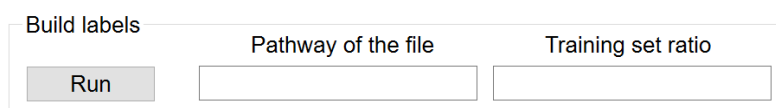
1. Data augmentation: This method does not screen the original MS/MS data and directly adjusts the intensity of each fragment ion. Enter the full path of the .mgf file in the mgf file field (e.g., C:\Users\Li\Desktop\CNNtest\test\CTA.mgf), the preset classification (0, 1, 2, 3, etc.) mentioned above in the Label field, the number of augmentation rounds in the Rounds field, and the location of the output folder for the images in the Output pathway field (e.g., C:\Users\Li\Desktop\CNNtest\program_test\database\0). Regarding the calculation of the

number of MS/MS images after augmentation, the number of images after augmentation = the number of compounds in the .mgf file \times 4 \times the number of Rounds.

2.Data augmentation (Relative): This method first screens out fragment ions with intensities in the top n% and then performs augmentation. A new level option has been added, where $n = 5 (6, 7, 8, 9, 10) / \text{level} \times 100$. If level = 100, then n is 5%, 6%, 7%, 8%, 9%, and 10% respectively, and the program will screen fragment ions with intensities in the top 5%, 6%, 7%, 8%, 9%, and 10% for further augmentation. The inputs for the mgf file, Label, Rounds, and Output pathway fields are the same as above. In this method, the number of images after augmentation = the number of compounds in the .mgf file \times 24 \times the number of Rounds.

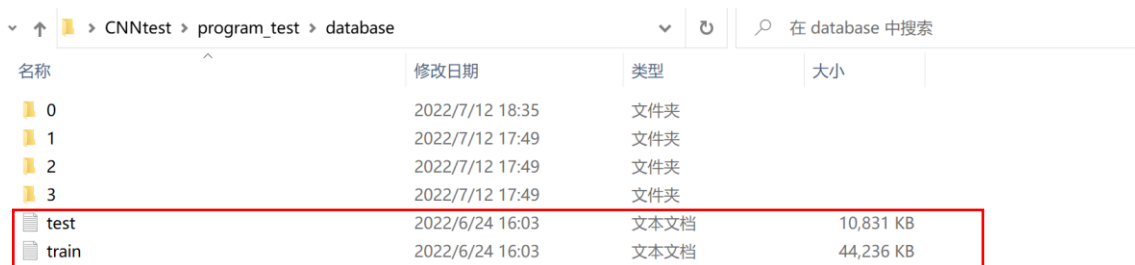
3.Data augmentation (Absolute): This method directly excludes fragments with relative abundances below 10%, 20%, 30%, and 40%, and then completes the augmentation by randomly adjusting the intensity of the remaining fragments. The inputs for the mgf file, Label, Rounds, and Output pathway fields are the same as above. In this method, the number of images after augmentation = the number of compounds in the .mgf file \times 16 \times the number of Rounds.

After preparing the dataset, the next step is to split the dataset and label the data. This can be accomplished using the Build labels module (Figure 17). Enter the total folder path where the data is located in the Pathway of the file field, which is the folder from Figure 13 (C:\Users\Li\Desktop\CNNtest\program_test\database). In the Training set ratio field, input the proportion of the training set, typically set at 0.8, with the remaining 0.2 serving as the validation set for model testing. After running this module, the appearance of two files, test.txt and train.txt (Figure 18), indicates that the dataset has been successfully split and labeled. With this, the preparation of the training set is complete.



The image shows a software window titled "Build labels". Inside the window, there is a "Run" button on the left. To the right of the button are two text input fields. The first field is labeled "Pathway of the file" and the second field is labeled "Training set ratio".

Figure 17



名称	修改日期	类型	大小
0	2022/7/12 18:35	文件夹	
1	2022/7/12 17:49	文件夹	
2	2022/7/12 17:49	文件夹	
3	2022/7/12 17:49	文件夹	
test	2022/6/24 16:03	文本文档	10,831 KB
train	2022/6/24 16:03	文本文档	44,236 KB

Figure 18

2.3 Model Training

The training of deep learning is completed on the Model training page (Figure 19). In DL-NPE, deep learning is implemented based on Pytorch. Parameter Settings. Select the NeutralNetwork 1 from the CNN model selection box. Enter the number of compound classifications in the Classification field. The Epoch is the number of times the entire dataset will be traversed, typically set to 40. For both Training data and Evaluation data, enter the folder path where the train.txt and test.txt files are located. For example, as shown in Figure 18, simply input C:\Users\Li\Desktop\CNNtest\program_test\database in both fields without appending \train.txt and \test.txt. For Batch Size, the model framework can have a maximum of 32 for GPUs with only 6 GB of video memory. If using a GPU with more video memory, this value can be increased. If the limit is exceeded, the program will automatically exit. During training, it is recommended to utilize the GPU memory as much as possible. GPUs with more memory can set a larger Batch Size to speed up model training. For Learning rate, set it above 0.001 in this research case, it is set to 0.001. Enter the folder path where the neural network model will be saved in the model pathway field. A new folder can be created within the training dataset folder to store the neural network model (Figure 20). In this case, input C:\Users\Li\Desktop\CNNtest\program_test\database\Model in this field.

Training Set

Model training

Data Testing

Parameters

CNN model

NeuralNetwork1

Classification

Epoch

Training Data

Evaluation Data

Batch Size

Learning rate

Model Pathway

Run

Processing

0%

Information

Figure 19

The screenshot shows a file explorer window with the path `CNNtest > program_test > database`. The file list contains the following items:

名称	修改日期	类型	大小
0	2022/7/12 18:35	文件夹	
1	2022/7/12 17:49	文件夹	
2	2022/7/12 17:49	文件夹	
3	2022/7/12 17:49	文件夹	
Model	2022/7/12 22:09	文件夹	
test	2022/6/24 16:03	文本文档	10,831 KB
train	2022/6/24 16:03	文本文档	44,236 KB

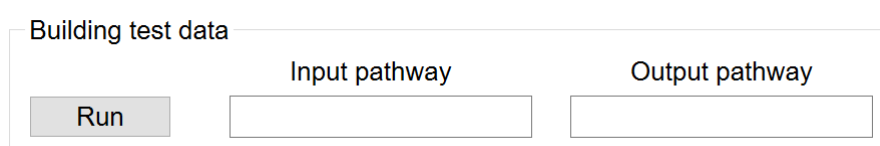
Figure 20

After filling in the parameters, click Run to start the training. Please be patient during the training process. As long as the program does not exit, it indicates that the training is proceeding normally. Once the training begins, the information box will display relevant training information, and the progress bar will show the progress. After one epoch is completed, the Information box will display model information, including the training loss and the model's prediction accuracy on the validation set, accuracy. Generally speaking, if the model training is

normal, the training loss will continuously decrease, and the accuracy will continuously increase. Meanwhile, a traindata.txt file will be generated in the folder specified in Model Pathway, which records the training loss, test loss (which should also show a downward trend), and accuracy for each epoch. In addition to NeutralNetwork 1 in this case, we have integrated NeutralNetwork 2 (GoogleNet) and NeutralNetwork 3 (common CNN) to tackle future research challenges and provide new avenues for exploration.


2.4 LC-MS/MS Data Analysis

After analyzing plant extract samples using LC-MS/MS, the data is converted to mzML or mzXML format using MSConvert and then processed with MZmine 2 following the Feature-based Molecular Networking workflow (refer to FBMN with MZmine - GNPS Documentation) to obtain mgf and csv files. The mgf and csv files are then transformed into MS/MS spectra using the Building test data tool (Figures 21 and 22). In the Input pathway field, enter the full path of the mgf file generated by MZmine 2 (for example, C:\Users\Li\Desktop\CNNtest\test\caoshanhu3.mgf). In the Output pathway field, specify the directory where the transformed results will be stored (for example, C:\Users\Li\Desktop\CNNtest\test\caoshanhu3\result). Upon completion, two folders, eval and test_cropped, will be generated within the example result folder.



The interface for the 'Building test data' tool. It features a 'Run' button on the left. To its right are two input fields: 'Input pathway' and 'Output pathway'.

Figure 21



A screenshot of a file explorer window showing the directory path: CNNtest > test > caoshanhu3 > result. The window displays two folders: 'eval' and 'test_cropped', both created on 2022/7/5 at 10:06.

名称	修改日期	类型	大小
eval	2022/7/5 10:06	文件夹	
test_cropped	2022/7/5 10:06	文件夹	

Figure 22

Data testing

Test

CNN model NeuralNetwork1 Model pathway

Classification

Input mgf

Input csv

Input MS/MS

Output pathway

Characteristic ions file

Selected labels

Filter threshold

Figure 23

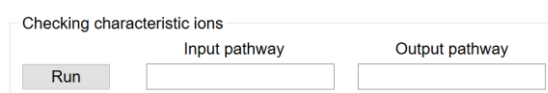
After that, the Data testing section can be run (Figure 23). Through the deep learning model, the target compounds of interest can be identified. First, select the CNN model, which must be consistent with the one used for training. In the Model pathway field, enter the full path of the trained neural network model, which is stored in the Model folder (see Figure 20), for example, C:\Users\Li\Desktop\CNNtest\program_test\database\Model\Model.pth. In the Classification field, enter the number of compound categories. For Input mgf and Input csv, specify the mgf and csv files generated by processing the LC-MS/MS data with MZmine 2. For Input MS/MS, enter the folder containing the eval and test_cropped files generated earlier, such as C:\Users\Li\Desktop\CNNtest\test\caoshanhu3\result. In the Output pathway field, specify the folder where the results will be output. A csv file will be generated, recording the information of the target compounds identified by the deep learning model (Figure 24), along with the mgf and csv files of the target compounds.

	A	B	C	D
1		label	likelihood	
2	0	1	1	
3	1	3	1	
4	2	3	1	
5	3	3	0.993328	
6	4	3	1	
7	5	3	1	
8	6	3	1	
9	7	0	0.999978	
10	8	1	0.979841	
11	9	0	0.995845	
12	10	0	0.95554	
13	11	0	0.999384	

Figure 24

Based on the deep learning model in DL-NPE, which can accurately identify the target compounds of interest, we have additionally designed a method to screen for compounds with

potentially unique skeletons among the target compounds. The first part involves collecting characteristic fragment ions for each category of target compounds. In DL-NPE (Figure 25), ions with a relative abundance higher than 0.6 are defined as characteristic fragment ions, which are then extracted and placed into a csv file. In the Input pathway field, enter the full path of the merged mgf file for each category of target compounds (Figure 11, 12), for example, C:\Users\Li\Desktop\CNNtest\test\merge.mgf. The Output pathway should specify the folder where the csv file recording the characteristic fragment ions of this category of compounds will be saved, such as C:\Users\Li\Desktop\CNNtest\test. After running the program, a file named Characteristic_ion.csv will be generated (Figure 26). Different types of compounds need to be processed separately. If the Output pathway is not changed, the csv file name should be modified after generating the results for one category of compounds; otherwise, the csv file will be overwritten when processing the next category of compounds. After obtaining the characteristic fragment ions for each category of target compounds, these data need to be manually integrated. First, create a new Excel worksheet, open the Characteristic_ion.csv file, copy the entire column of data, and paste it into Excel. Change the header from characteristic_ion to the classification label for that category of compounds, such as 1, 2, etc. (Category 0 represents the non-target compound dataset and does not require characteristic ion detection; see Figure 27). Finally, click on the top-left corner: File > Save As > Save as type: CSV UTF-8 (comma-delimited), and save the file as a csv.



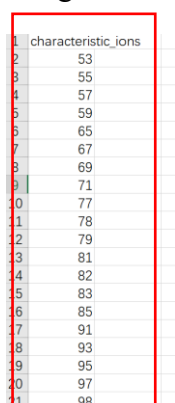
Checking characteristic ions

Input pathway

Output pathway

Run

Figure 25



characteristic_ions	
53	
55	
57	
59	
65	
67	
69	
71	
77	
78	
79	
81	
82	
83	
85	
91	
93	
95	
97	
98	

Figure 26

	A	B	C	D
1	1	2	3	
2	55	53	197	
3	57	55	199	
4	59	57	215	
5	67	59	225	
6	73	65	226	
7	77	67	243	
8	79	69	245	
9	80	71	257	
10	81	77	258	
11	82	78	266	
12	83	79	295	
13	91	81	296	
14	93	82	499	
15	95	83	539	
16	97	85	571	
17	103	91	663	
18	105	93	713	
19	106	95	741	
20	107	97	759	
21	109	98	773	

Figure 27

The software further extracts the top 20 most abundant fragment ions from the target compounds and compares them with the characteristic fragment ions of the corresponding compound category (information in Figure 27). It calculates how many of these fragment ions are present in the list and then determines the proportion of these ions among the top 20 fragment ions. Specifically, in addition to the information required for identifying target compounds, the Characteristic ions file field is added to input the complete path of the integrated csv file containing the characteristic fragment ions of each target compound (e.g., C:\Users\Li\Desktop\CNNtest\test\caoshanhu3\merge.csv). The Filter threshold is set to determine the threshold for potential new skeleton compounds. For example, entering 0.3 means that among these compounds, no more than 6 of the top 20 fragment ions can match the characteristic fragment ions of the standard ($6/20 = 0.3$). The smaller this value is set, the stricter the screening will be. After clicking test and running the process, two files, target.mgf and target.csv, will be generated. These files can be directly uploaded to the GNPS platform for molecular networking analysis.

Additionally, there is another method for running the Data testing section, which is used to validate the performance of the deep learning model. This requires MS/MS data of compounds not included in the training set (including both target and non-target compounds). First, prepare the datasets for these compounds according to the method described in Section 2.1 (the mgf files need to be processed as shown in Figure 28). Then, construct the test data using the Building test data method described in this section, and proceed with the Data testing module. During testing, after selecting the model, only the Model pathway, Classification, and Input MS/MS fields need to be filled in. After clicking test and completing the run, a testdata.csv

file will be generated in the folder containing the eval and test_cropped files (Figure 22). Open this file for inspection. Since standard compounds are used, the compound classification labels are known. The number of non-target labels in the label column (Figure 24) represents the number of compounds incorrectly predicted by the model. By calculating the accuracy of the model's predictions, the optimal deep learning model can be selected for analyzing actual samples.

MSLEVEL=2 Title: Scan#: 679, RT: 4.02665 min 50.22727966308594 11.03125	MSLEVEL=2 TITLE=679_4.02665_ 50.22727966308594 11.03125
1. Replace ' , RT: ' with ' _ '	
MSLEVEL=2 Title: Scan#: 679, RT: 4.02665 min 50.22727966308594 11.03125	MSLEVEL=2 TITLE=679_4.02665_ 50.22727966308594 11.03125
2. Replace ' min' with ' _ '	
MSLEVEL=2 Title: Scan#: 679, RT: 4.02665 min 50.22727966308594 11.03125	MSLEVEL=2 TITLE=679_4.02665_ 50.22727966308594 11.03125
3. Replace ' Title: Scan#: ' with ' TITLE= '	

Figure 28