**DeepHalo Documentation**

**DeepHalo: A deep learning-integrated workflow for high-throughput discovery of halogenated metabolites from HRMS data.**

***Version 1.0.0***

**Table of Contents**

1. [**Introduction**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#introduction)
2. **Data Support**
3. [**Core Features**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#core-features)
4. [**Technical Advantages**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#technical-advantages)
5. [**Installation**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#installation)
6. [**Quickstart**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#quickstart)
7. [**Command-Line Usage**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#command-line-usage)
8. **Output Directory Structure**
9. **Instruction**
10. [**Dependencies**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#dependencies)
11. [**License**](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#license)

**Introduction**

DeepHalo is a hierarchically optimized workflow designed for the detection and dereplication of halogenated compounds in **high-resolution** mass spectrometry (HRMS) data. It integrates cutting-edge deep learning models, robust isotope validation, comprehensive scoring, and dual dereplication strategies to deliver high accuracy and efficiency. Key applications include natural product discovery and halogenated metabolite annotation.

**Data Support**

DeepHalo supports **high-resolution** mass spectrometry (HRMS) data, with or without tandem mass spectrometry (MS/MS) information. Note that the MS2 extraction function currently supports only data-dependent acquisition data (DDA). Additionally, DeepHalo exclusively supports the **.mzML** file format for analysis.

**Core Features**

**1. Halogen Prediction**

* **Element Prediction Model (EPM)**
  + Dual-branch Isotope Neural Network (IsoNN) architecture
  + High accuracy Cl/Br detection (>99.9% precision based on benchmark results)
  + Wide mass range coverage (50-2000 Da)
  + Robust interference resistance to B/Se/Fe/dehydro isomers

**2. Isotope Pattern Validation**

* **Dual Validation System**
  + *Mass Dimension*: Statistical rule-based correction.
  + *Intensity Dimension*: Autoencoder-based Anomaly Detection Model (ADM).

**3. Multi-Level Halogen Confidence Scoring (H-score)**

* **- Dual levels**
  + Prediction based on centroid-level isotope patterns
  + Prediction based on Scan-level isotope patterns
  + H-score integration for comprehensive assessment on the above both levels

**4. Dereplication**

* **Dual Strategy Approach**
  + *MS1-based dereplication using Custom Database Matching*: Validates exact mass, halogen patterns, and isotope intensity similarity.
  + *MS2-Based Dereplication by Integrating GNPS*: MS2 molecular networking, halogenated compound annotation, graphML file enhancement.

**Technical Advantages**

* **High Throughput**: automatically process unlimited samples in several to dozens of seconds each on standard hardware (Core i9, 16GB RAM).
* **Accuracy**: >98.3% precision in halogen detection across simulated and experimental LC-MS datasets.
* **Integration with GNPS**: enhance molecular network annotation in the element dimension by embedding DeepHalo results into GNPS output graphML file
* **Efficient Dereplication**: significantly higher dereplication rate compared to molecular networking alone in GNPS

**Installation**

**Prerequisites**

* Python 3.10 (Verify with **python --version**).

**Installation Methods**

**From PyPI**

pip install DeepHalo

**From Local Wheel**

pip install path/to/DeepHalo-xxx.whl

**From Source**

git clone https://github.com/xieyying/DeepHalo.git

cd DeepHalo

pip install -e .

**Quickstart**

**1. Detect Halogenated Compounds in mzML Files**

halo detect -i /path/to/mzml\_files -o /path/to/output\_directory -ms2

**2. Dereplication with GNPS and/or Custom Database**

halo dereplicate -o /path/to/output\_directory -g /path/to/GNPS\_results -ud /path/to/custom\_database.csv

**Command-Line Usage**

**General Help**

halo --help # List all commands

halo [command] --help # Show options for a specific command (e.g., `halo detect --help`)

**Commands**

**1. Analyze mzML Files**

halo detect -i <input\_path> -o <project\_path> [-c <config\_file>] [-b <blank\_samples\_dir>] [-ms2]

**2. Dereplication**

halo dereplicate -o <project\_path> [-g <GNPS\_folder>] [-ud <user\_database.csv>]

**3. Create Training Dataset**

halo create-ds <project\_path> [-c <config\_file>]

**4. Train Model**

halo train <project\_path> [-c <config\_file>] [-m search]

**Output Directory Structure**

/output\_directory

├─dereplication

│ Demo\_data\_1\_feature.csv

│ Demo\_data\_2\_feature.csv

└─result

│ config.toml

│ error\_files.txt

├─halo

│ Demo\_data\_1\_feature.csv

│ Demo\_data\_1\_scan.csv

│ Demo\_data\_2\_feature.csv

│ Demo\_data\_2\_scan.csv

└─ms2\_output

Demo\_data\_1.mzML

Demo\_data\_2.mzML

**Instruction**

The whole DeepHalo analysis process including 4 steps

**Step 1: Halogenate Mining**

**Run the command:**

halo detect -i <input\_path> -o <project\_path> [-c <config\_file>] [-b <blank\_samples\_dir>] [-ob] [-ms2]

**Parameters:**

* -i INPUT\_PATH (Required)
  + Input .mzML file or directory
  + Example (Linux): -i /home/data/ms\_files
  + Example (Windows): -i D:\data\ms\_files
* -o OUTPUT\_DIR (Required)
  + Output directory path
  + Example (Linux): -o /home/analysis/output
  + Example (Windows): -o D:\analysis\output
* -c <config\_file> (Optional)
  + Parameters for LCMS data analysis
  + Example(Linux): -c /home/deephalo/config.toml
  + Example (Windows): -c D:\deephalo\config.toml
* -b <blank\_samples\_dir> (Optional)
  + Blank .mzML file or directory
  + Example (Linux): -b /home/analysis/blank
  + Example (Windows): -b D:\analysis\blank
  + Run blank exclusion during analysis
  + Default: disabled
* -ob (Optional)
  + Force regenerate blank feature detection results for blank exclusion
  + Default: disabled
* -ms2 (Optional)
  + Enable MS2 data extraction
  + Optional parameter (required for steps 2-4)
  + Default: disabled

**Example Usage:**

halo analyze-mzml -i D:\data\ms\_files -o D:\analysis\output -ms2

**Output Files**

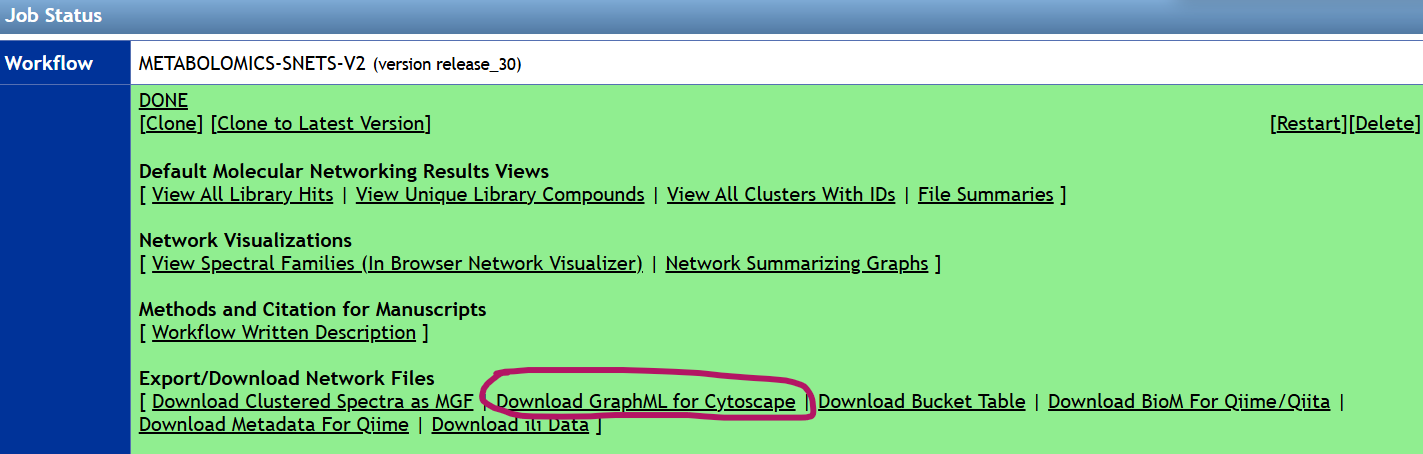
* config.toml: Analysis parameters
* error\_files.txt: Failed mzML files
* halo\: MS1 information
* ms2\_output\: MS2 data

**Notes**

* Results are filtered by H-score
* To disable filtering: Set H-score to 0 in config.toml file
* H-score visible in Cytoscape after dereplication
* For failed files:
  1. Retry analysis
  2. If error persists, reconvert raw data to mzML

**Step 2: Molecular Networking analysis employing GNPS**

1. Submit MS2 data in ms2\_output to GNPS platform
2. Analyze molecular similarity using GNPS platform
3. Download and unzip results



**Example GNPS output structure:**

└─Demo\_GNPS\_output

│ METABOLOMICS-SNETS-V2-xxx-main.graphml

│ params.xml

├─clusterinfo

├─clusterinfosummarygroup\_attributes\_withIDs

├─gnps\_molecular\_network\_graphml

└─result\_specnets\_DB

**Step 3: Dereplication**

**Run the command:**

halo dereplicate -o PROJECT\_PATH [-g GNPS\_DIR] [-ud DATABASE\_FILE]

* -o PROJECT\_PATH (Required)
  + Output directory (same as used in function of ‘detect’)
  + Example: -o D:\analysis\output
* -g GNPS\_DIR (Optional)
  + GNPS results directory containing .GraphML file
  + Example: -g D:\analysis\Demo\_GNPS\_output
* -ud DATABASE\_FILE (Optional)
  + Custom database file (CSV/JSON)
  + Example: -ud D:\databases\compounds.csv

**Output Details**

Upon completion of the analysis, the following files and directories will be generated:

1. **‘dereplication’ Directory**
   * Contains processed MS1 data in CSV format.
   * If a database was provided, compound match information will also be included in the csv file.
2. **GraphML File (generated when GNPS results are provided)**
   * The file name will have the suffix \_adding\_DeepHalo\_results.
   * This file contains the mean H-score and classification predicted from centroid isotope patterns, along with any database match information if a dereplication database was used.
3. **Processed Database File (if a database is provided)**
   * The file name will include the suffix \_DeepHalo\_dereplication\_ready\_database.
   * This file is formatted for future analyses, significantly reducing processing time in subsequent runs.

**Example DATABASE\_FILE:**

|  |  |  |
| --- | --- | --- |
| compound\_name | formula | Smiles |
| Compound1 | C6H5Cl | ClC1=CC=CC=C1 |
| Compound2 | C6H4Cl2 | ClC1=CC=C(Cl)C=C1 |

**Important Notes**

1. **File Format**
   * The input file must be in CSV format.
2. **Required Columns**
   * compound\_name
   * formula
3. **Optional (but Recommended) Column**
   * Smiles (These aid in structure visualization).
4. **Additional Requirements**
   * Ensure that the formula column contains valid molecular formulas.
   * The file must be encoded in UTF-8.

**Step 4: Visualization in Cytoscape**

To visualize the results, open the GraphML file (with the suffix \*\_adding\_DeepHalo\_results.graphml) in Cytoscape. The file includes the following annotated variables:

* **H\_scoreMean:**
  + **Range:** [0, 1]
  + **Default Threshold:** 0.4, above which the presence of halogen is inferred.
  + **Interpretation:** Higher values indicate a greater likelihood of halogen presence.
  + **Other Factors Affecting the Score:**
    - Sample complexity
    - Mass spectrometer resolution
* **classification:**
  + **Isotope pattern classifications (0-7):**
    - *0:* X-type (mixed/polyhalogenated)
    - *1*: Br-type (Br/Cl₃)
    - *2*: Cl-type (Cl/Cl₂)
    - *3*: Se-type (Selenium compounds)
    - *4*: B-type (Boron compounds)
    - *5*: Fe-type (Iron compounds)
    - *6*: C-type (compounds containing only C, H, O, N, F, P, S, Na)
    - *7*: artifact-type (artifact isotope patterns)
* **Inty\_cosine\_score:**  
  Cosine similarity between experimental and theoretical isotope peak intensities. Higher values indicate a greater likelihood of a known compound
* **Compound\_names and Adducts:**These fields provide the known molecular names and adduct information sourced from a user-provided database, offering essential metadata for compound identification.
* **Smiles**

Provides the known molecular structure, sourced from either the GNPS library and a user-provided database.

* **error\_ppm**

Represents the mass error between the measured m/z value of the test molecule and that of its corresponding known compound match.

**Dependencies,**

* pandas == 2.0.3
* numpy == 1.22.0
* tensorflow == 2.10.1
* scikit-learn == 1.3.1
* pyopenms == 3.1.0
* Full list: See [README.md](https://chat.deepseek.com/a/chat/s/d3b7de1c-627f-43db-b551-a75fc99a7c87#dependencies)

**License**

Distributed under the [MIT License](https://github.com/xieyying/DeepHalo/blob/main/LICENSE).

*For methodology details and benchmarks, refer to the* [*GitHub repository*](https://github.com/xieyying/DeepHalo)*.*