NTA Analyzer User Guide Version 3

Author: Hussein al Ghoul

Introduction: This software was developed to initialize and execute the NTA workflow developed at EPA. It is based on a SAS version developed by Jon Sobus and a consequent R version developed by Marie Russell and later upgraded by James McCord. This software package was primarily developed in python using scientific packages such as pandas and numpy, and utilizes the Tkinter package to include a user-friendly Graphical User Interface. In what follows is a detailed explanation on how to execute the software and details on what the structure of the input files is expected to be.

Python version 2.7

Spyder Version 3.2.4

Anaconda with Python 2.7

Code Blocks:

1. **Functions_Universal_v3.py**: This is the backend script that performs all the heavy lifting of the software. Functions in this block perform a number of tasks ranging from duplicates elimination of the MPP output csv file, to identifying a set of adducts in the data. The code starts by reading a csv or tsv file, it then selects columns that are deemed important for the analysis, drops those that are irrelevant, and uses a string match sequencing method to identify sample replicates**. It is important to keep in mind that the replicates of the sample should have similar names with a number identifier. An example of a good replicate (triplicate) chain: Sample1_T1, Sample1_T2, Sample1_T3. The code will find that these three strings have similar structure besides the number and assumes that they belong to the same sample. An example of bad chain: Sample1_T1, sample1_T2, Sample1_t2. The code will assume that the previous three strings are different and will therefore assume that they belong to different samples.** Note: Positive mode and negative mode files should have the same sample names. Drop any ionization mode identifiers. Good names: Sample1_T1, Sample1_T2,Sample1_T3 (in both positive and negative mode files). Bad names: Sample1_pos_T1, Sample1_pos_T2,Sample1_pos_T3 and Sample1_neg_T1, Sample1_neg_T2,Sample1_neg_T3.
2. **Batch_search_v3.**py: This script is used to hit the comptox dashboard remotely. It uses a webdriver (chromedriver) to automate the execution of a chrome browser window and automatically copy the masses the user intends to search on the dashboard. During this process, the User is expected to stop using the mouse on the screen where the Chrome window with the dashboard is opened. Until an API is developed for this purpose, this is the most efficient way to search the dashboard and automate the whole workflow process.
3. **Toxpi_v3.**py: This script is designed to perform some post processing of the data pulled from the dashboard and in later versions will perform extensive toxpi calculations.
4. **CFMID_v3.**py: This script uses a list of masses that were searched on the dashboard to perform CFMID matching. It requires MS/MS mgf files for positive and negative modes to search the

CFMID database for matches. Each mass is searched in the database within a user specified window, pulls all the DTXCID matches, then looks for fragment matches between these DTXCIDs and the input mass. A score is then calculated using cosine dot product. Results are then ranked based on either their overall score (sum of multiple energy scores) or a score at a specific collision energy. DTXCIDs with no fragment matches with the input spectrum are automatically assigned a score of zero.

5. **CDP_v3.**py: This script does all the matching and scoring of the DTXCIDs for CFMID.
6. **Gui_v3.**py: This is the script that is executed by the user in a spyder environment. This frontend GUI executes all the functions required by the NTA workflow at the EPA. A description of the different parts of this GUI is provided below.

Graphical User Interface:

1. **Setup**: To begin using this software, open Spyder. Drag the Gui_v3.py file into spyder, it will become a tab in the platform. Before executing this software, a couple packages are needed. To install them got to the Start Menu -> Anaconda2 -> Anaconda prompt. A black prompt window will open.

   In the prompt window type: *conda install selenium*

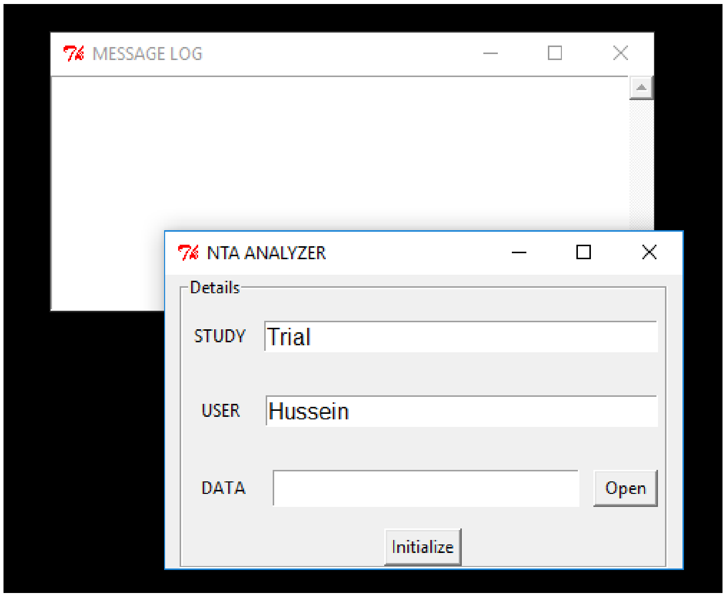
   Hit enter and wait for it to finish

   Once it is done type: *conda install pymysql*

   Hit enter and wait for it to finish

Now that both packages are installed into spyder, you can run the GUI_v3.py file.

2. **Running the Gui_v3.py script**:
   1. Hit the green right pointing triangle on the top tab of the spyder platform, this will open the windows below.



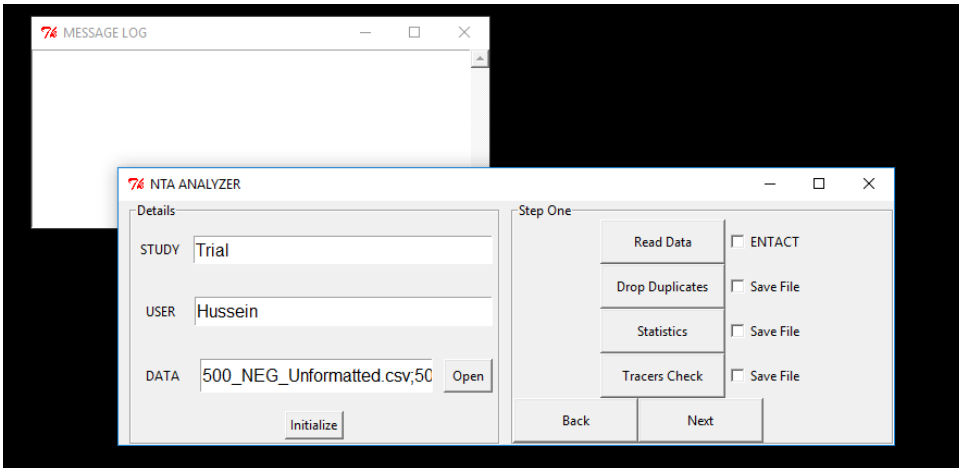
STUDY Trial

USER Hussein

DATA

The inputs **STUDY** and **USER** are used to create a folder where all the outputs from different steps of the analysis are stored. Hit the **Open** button to select the input files. Two files are required, positive mode MPP file and negative mode MPP file. Remember to check the naming structures of the samples as outlined in section one. Select the two files from that open window and hit **Initialize**.

2. A new frame will instantly appear and a set of buttons appears as shown below. The user starts by selecting whether this is an ENTACT EPA analysis. If it is, check the ENTACT checkbox, otherwise leave unchecked. Hit the Read button, it will instantly turn green indicating that the data reading process is done. This means that the CSV, or TSV, files that the user input into the software are of the right format and the code can be further executed. If this step fails, the button will turn red indicating an error. In this case, check the logging console in Spyder (bottom right in the spyder window) for errors. Check the structure of your data that you input into this software.

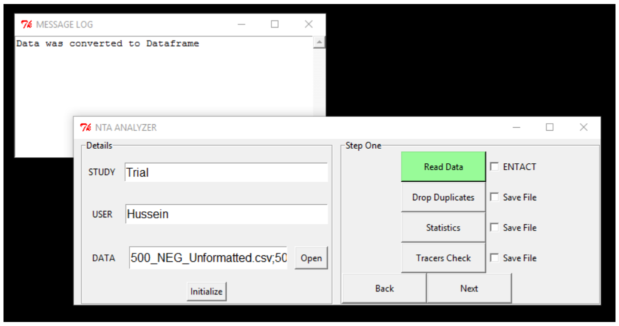


STUDY Trial

USER Hussein

DATA 500_NEG_Unformatted.csv;50

Data properly read gives the following updated GUI:



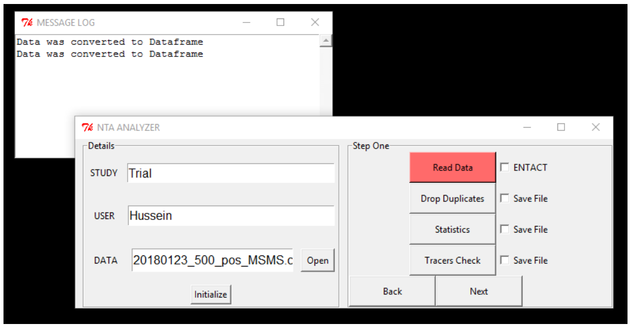
```
Data was converted to Dataframe
```

STUDY Trial

USER Hussein

DATA 500_NEG_Unformatted.csv;50

Data improperly read gives this updated GUI:



```
Data was converted to Dataframe
Data was converted to Dataframe
```

STUDY Trial

USER Hussein

DATA 20180123_500_pos_MSMS.c

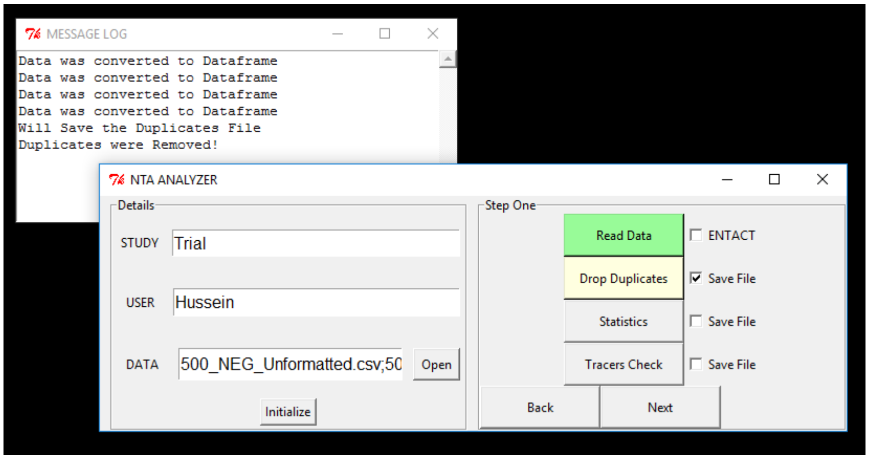
□ ENTACT

□ Save File

□ Save File

□ Save File

3. Before the user executes the **Drop Duplicates** button, they need to make a choice whether they want to save the processed data after performing this action to a csv file. If they decide to, the **Save File** checkbox should be checked. Once clicked, the button will turn yellow indicating that the software is performing the task. This step is quick but greatly depends on the size of the data being processed. Below is an example of what the GUI looks like while a task is being performed.



```
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Will Save the Duplicates File
Duplicates were Removed!
```

STUDY Trial

USER Hussein

DATA 500_NEG_Unformatted.csv;50

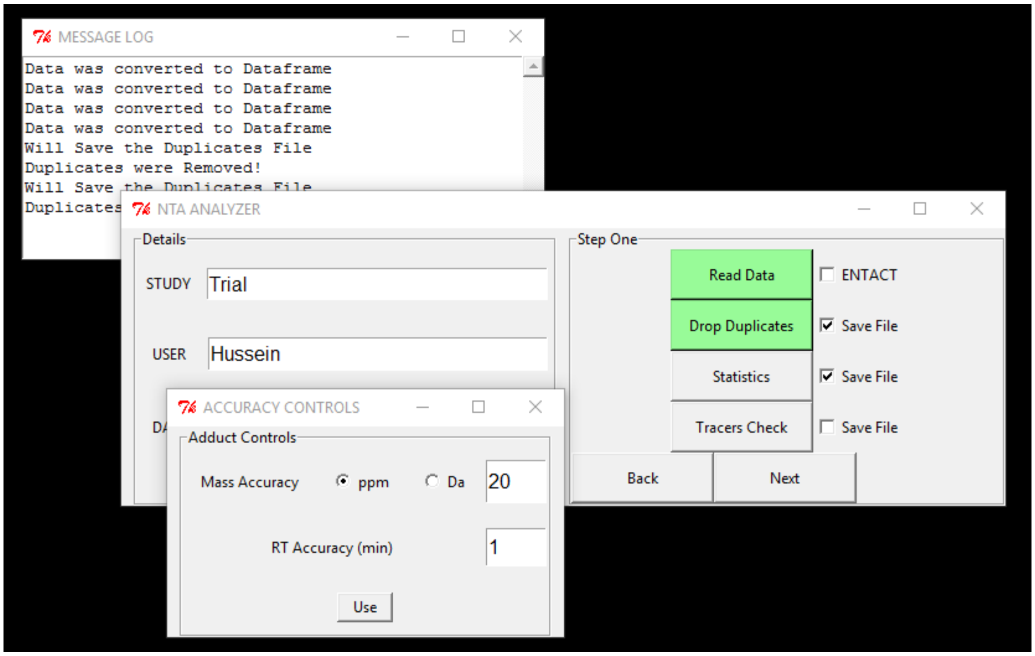
□ ENTACT

☑ Save File

□ Save File

□ Save File

4. Next is the Statistics part. This process calculates a set of Statistical variables of every Sample set (Mean, Median, standard deviation, CV, Number of hits) and appends the results to the processed data. When the user hits the **Statistics** button, an **“ACCURACY CONTROLS”** window appears. This window requires inputs for the mass window and retention time window to identify adducts. Select whether the mass accuracy should be in **ppm** or **Da**, input the numbers you want to use and click on **Use**. The GUI will look like this:



```
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Will Save the Duplicates File
Duplicates were Removed!
Will Save the Duplicates File
Duplicates
```

STUDY: Trial

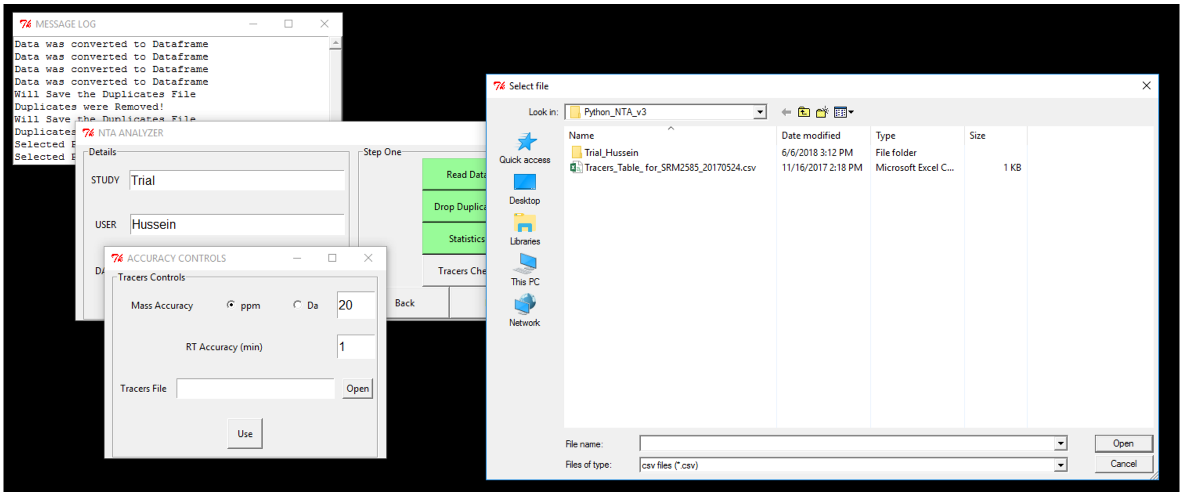
USER: Hussein

Mass Accuracy ● ppm ○ Da 20

RT Accuracy (min) 1

Step One: ☐ ENTACT · ☑ Save File · ☑ Save File · ☐ Save File

5. Next is checking the Tracers. This step requires a predefined tracers CSV file, and example of which is provided with this package. It is fully independent and could be skipped, unlike the previous two steps. When the **Tracers Check** button is clicked, a **Tracers Control** window appears. The values for the **Mass Accuracy** and **RT Accuracy** that the user input in the Adduct Identification step before will be automatically appended to their fields, but they can be changed. A tracers file is needed and should be selected with the **Open** button. Once the Tracers file is selected, the user can click the **Use** button. If the Tracers Check button turns red, consider checking the structure of the tracers file. Note: Even if an error is raised, this step can be skipped and the workflow will not be interrupted. The GUI will look like this when the Open button is clicked.



```
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Will Save the Duplicates File
Duplicates were Removed!
Will Save the Duplicates File
Duplicates
Selected F
Selected F
```

STUDY: Trial

USER Hussein

Mass Accuracy ● ppm ○ Da 20

RT Accuracy (min) 1

Tracers File

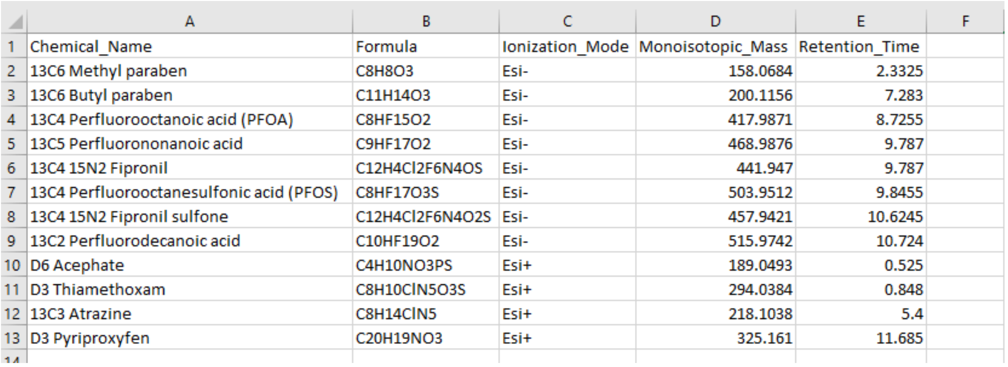
Look in: Python_NTA_v3

Name	Date modified	Type	Size
Trial_Hussein	6/6/2018 3:12 PM	File folder	
Tracers_Table_ for_SRM2585_20170524.csv	11/16/2017 2:18 PM	Microsoft Excel C...	1 KB

File name:

Files of type: csv files (*.csv)

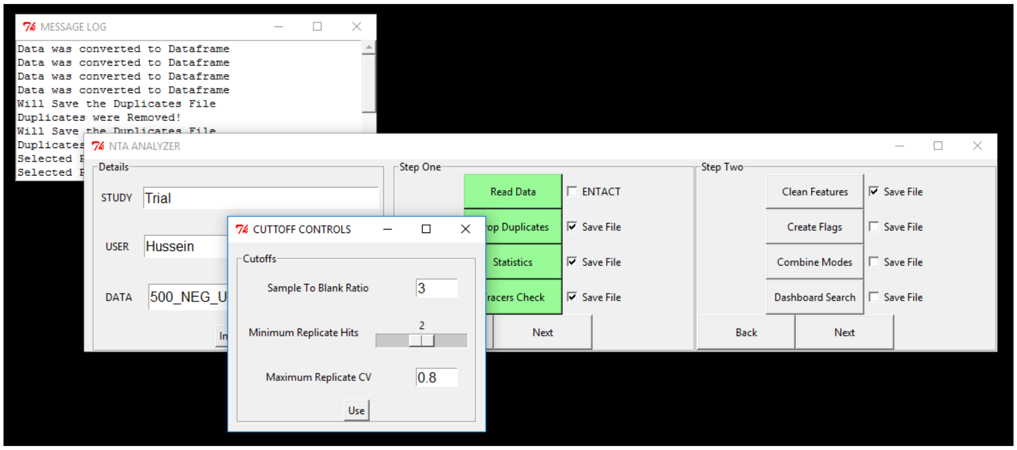
The Tracers file has the following structure:



	A	B	C	D	E	F
1	Chemical_Name	Formula	Ionization_Mode	Monoisotopic_Mass	Retention_Time	
2	13C6 Methyl paraben	C8H8O3	Esi-	158.0684	2.3325	
3	13C6 Butyl paraben	C11H14O3	Esi-	200.1156	7.283	
4	13C4 Perfluorooctanoic acid (PFOA)	C8HF15O2	Esi-	417.9871	8.7255	
5	13C5 Perfluorononanoic acid	C9HF17O2	Esi-	468.9876	9.787	
6	13C4 15N2 Fipronil	C12H4Cl2F6N4OS	Esi-	441.947	9.787	
7	13C4 Perfluorooctanesulfonic acid (PFOS)	C8HF17O3S	Esi-	503.9512	9.8455	
8	13C4 15N2 Fipronil sulfone	C12H4Cl2F6N4O2S	Esi-	457.9421	10.6245	
9	13C2 Perfluorodecanoic acid	C10HF19O2	Esi-	515.9742	10.724	
10	D6 Acephate	C4H10NO3PS	Esi+	189.0493	0.525	
11	D3 Thiamethoxam	C8H10ClN5O3S	Esi+	294.0384	0.848	
12	13C3 Atrazine	C8H14ClN5	Esi+	218.1038	5.4	
13	D3 Pyriproxyfen	C20H19NO3	Esi+	325.161	11.685	

Press **Next** once the **Tracers Check** Button turns green.

6. Next is the **Clean Features** Step. This step uses predefined variables to eliminate unnecessary parts of the data. **Sample to Blank Ratio** is used to select only Features where the Sample Median divided by the Blank Median is greater than or equal a user defined ratio. Any number greater than zero could be input in this field. The **Minimum Replicates Hits** slider is used to drop features that do not appear in some number of replicates in all samples. In Other words, if the user selects 2 in the slider, a feature that appears in less than two replicates in **all** samples will be dropped. The slider is automatically adjusted based on the replicates in the data. In this example, the data comes in triplicates so the slider has a maximum of 3. **Maximum Replicate CV** is used to drop features where the CV in **all** samples is greater than the input value. Once the variables are assigned values, click the **Use** button.



```
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Will Save the Duplicates File
Duplicates were Removed!
Will Save the Duplicates File
Duplicates
Selected E
Selected E
```

STUDY Trial

USER Hussein

DATA 500_NEG_U

Sample To Blank Ratio 3

Minimum Replicate Hits 2

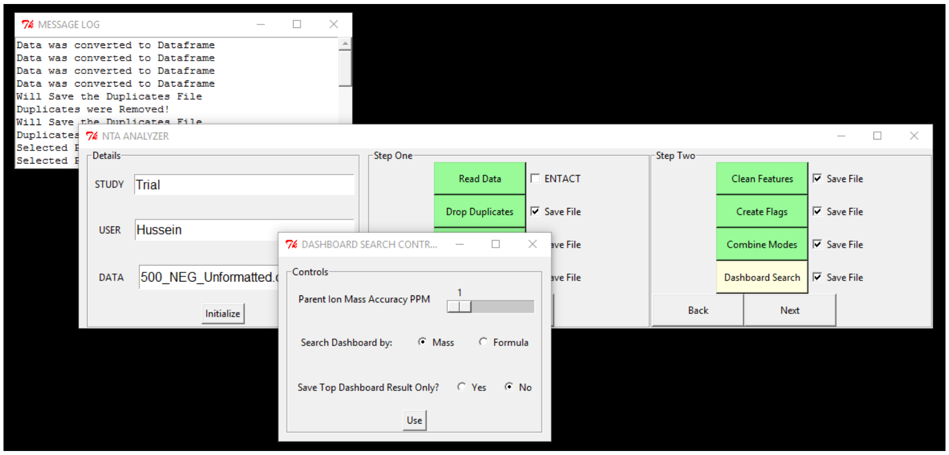
Maximum Replicate CV 0.8

7. Next step is Create Flags. In this step a set of predefined flags used for further post processing are created. This step is quick and does not require any inputs.

8. Next is the **Combine Modes** step. This process combines the separately yet dependently processed positive and negative mode data into one file. This step takes some time depending on the size of the data. It should be noted that due to limitations in multithreading a tkinter, the Combine Modes button will not turn yellow while the data is

processed. It will be “stuck” in the clicked position. However, data is being processed in the background. Once the button is “released”, the next step can be initiated.

9. Next is the **Dashboard Search** step. A set of masses that are flagged for dashboard search during the Create Flag step (See the **For_Dashboard_Search** column in the previous step’s output). These masses are selected based on their formula match score (above 90), or if the feature has a score below 90, a negative mass defect, and contains a Halogen. These masses are then appended into a list and the dashboard is searched with a user specified mass accuracy (ppm). To automate this step an API is needed to remotely query the dashboard. Since such a service is still under development, a webdriver and python’s Selenium wrapper are used to automate the process of opening a browser, and making selections on the comptox dashboard. When the **Dashboard Search** button is clicked, a controls window appears. The user selects the mass accuracy (1-10 based on the accuracy from the dashboard), whether the search is done by mass or formula, and if the top dashboard hit is only kept (if the user selects No, all hits will be returned). After making all these selections, click the Use button. In a few seconds, a chrome window will automatically popup. The user is expected to cease using the mouse in the current screen as it will interrupt the websdriver’s automation. The webdriver will control the window, making selections, copying the mass list, and downloading the dashboard file automatically. Once the download is complete, the software will look through the structure of the downloaded file and if all works well, the **Dashboard Search** button will turn green.



```
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Data was converted to Dataframe
Will Save the Duplicates File
Duplicates were Removed!
Will Save the Duplicates File
Duplicates
Selected E
Selected E
```

STUDY Trial

USER Hussein

DATA 500_NEG_Unformatted.c

□ ENTACT

☑ Save File

☑ Save File

☑ Save File

☑ Save File

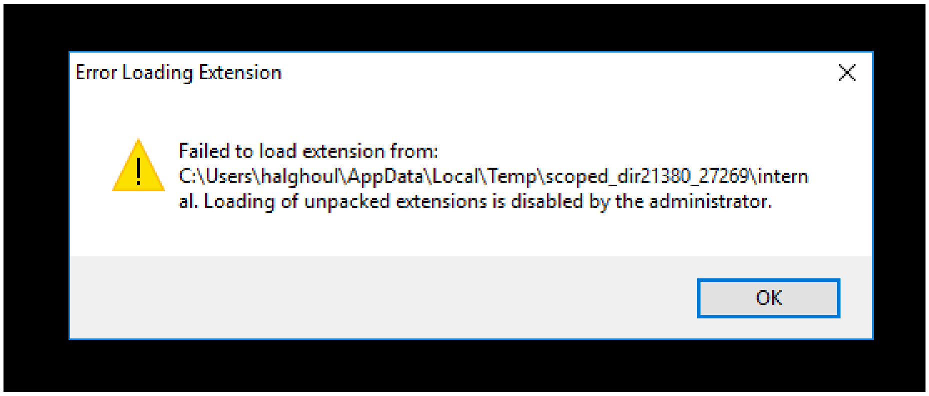
☑ Save File

Parent Ion Mass Accuracy PPM 1

Search Dashboard by: ● Mass ○ Formula

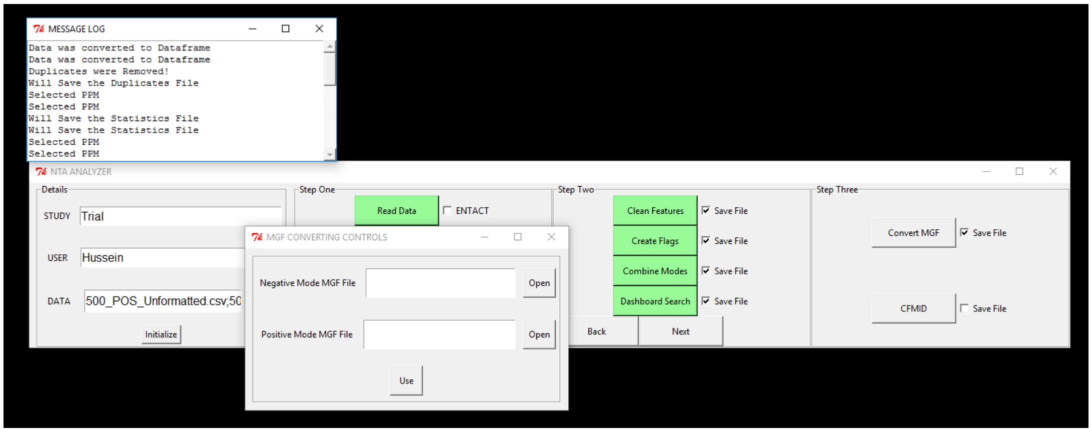
Save Top Dashboard Result Only? ○ Yes ● No

Note: For users with computers under administrative restriction (such as the **EPA**), the following window will popup, click ok and cease using the mouse on the current screen immediately.


Error Loading Extension

Failed to load extension from: C:\Users\halghoul\AppData\Local\Temp\scoped_dir21380_27269\internal. Loading of unpacked extensions is disabled by the administrator.

10. Next is the **Convert MGF** Step. This step and the following one, are specific to users interested in integrating CFMID predictions into their workflow. In this step, an MGF file of MS/MS data from an Agilent QTOF is parsed into a format that would allow linking MS data with MSMS data and subsequently extracting a spectrum to perform matches with the CFMID predicted data. Two files are to be supplied here, negative and positive mode data files. The software checks if these files have been already parsed, in which case this step is quick. If parsing is needed, this process can take some time. Once the parsing is finished, the **Convert MGF** button turns green.


MESSAGE LOG

```
Data was converted to Dataframe
Data was converted to Dataframe
Duplicates were Removed!
Will Save the Duplicates File
Selected PPM
Selected PPM
Will Save the Statistics File
Will Save the Statistics File
Selected PPM
Selected PPM
```

NTA ANALYZER

Details

STUDY Trial

USER Hussein

DATA 500_POS_Unformatted.csv;50

Step One: ☐ ENTACT

Step Two: Clean Features ☑ Save File; Create Flags ☑ Save File; Combine Modes ☑ Save File; Dashboard Search ☑ Save File

Step Three: Convert MGF ☑ Save File; CFMID ☐ Save File

MGF CONVERTING CONTROLS

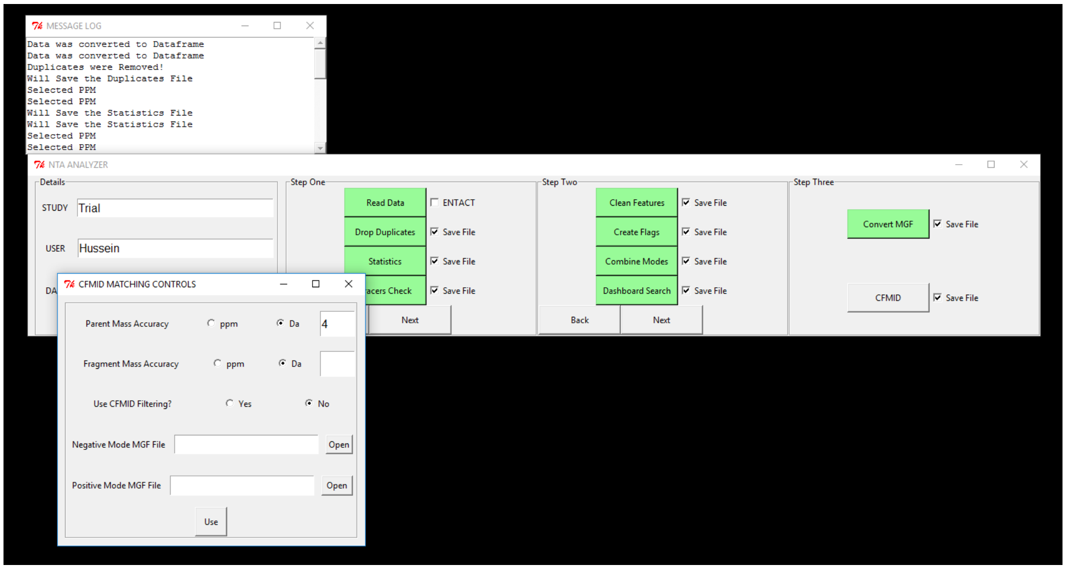
Negative Mode MGF File

Positive Mode MGF File

11. Next is the **CFMID** step. This step is currently specific to EPA users. A database with predicted CFMID spectra for dsstox data is currently restricted to internal access. Each mass searched on the dashboard is looked up in the MGF file for its corresponding spectrum. These masses are then search in the CFMID data based within a mass accuracy window (same as the accuracy window in the **Dashboard Search** step). The spectrum of each mass hit in the CFMID database is then pulled down and compared to the spectrum of the searched mass. If at least one fragment in the predicted spectrum matches one of the searched spectrum, a score is assigned. The scoring in this case is a Cosine Dot Product. Masses with no matching fragments are automatically assigned a score of zero. All these matched masses are the ranked by their scores (either as the sum of three collision energies score, or as an individual energy score).

When the user clicks the **CFMID** button, a controls window pops up. A mass accuracy window is preset (this is the same mass accuracy window the user input into the dashboard search step). A fragment accuracy is also expected. CFMID filtering can be enabled if the user wants to only keep the 30 highest intensity fragments in the predicted spectrum. Next the converted (the ouputs from the previous step, where the mgf files are converted to csv files) are selected for positive and negative modes. Once the user is done setting these controls, they can click the Use button. The CFMID matching process will start. This step could take a long time, depending on the number of searched masses and the population is each spectrum. The **CFMID** button will not turn yellow indicating ongoing processing in this case, but the progress can be checked in the Spyder window. Once the process is done, the **CFMID** button turns green. The final product of this workflow will have the names:

- **Data_MultiScores_Both_Modes.csv**: scores for matches for all three collision energies are used, and ranking is done based on the lowest collision energy.
- **Data_OneScore_Both_Modes.csv**: scores for matches are ranked based on the sum of all energy scores.



MESSAGE LOG

```
Data was converted to Dataframe
Data was converted to Dataframe
Duplicates were Removed!
Will Save the Duplicates File
Selected PPM
Selected PPM
Will Save the Statistics File
Will Save the Statistics File
Selected PPM
Selected PPM
```

NTA ANALYZER

Details

STUDY Trial

USER Hussein

Step One: ☐ ENTACT; Drop Duplicates ☑ Save File; Statistics ☑ Save File; Tracers Check ☑ Save File

Step Two: Clean Features ☑ Save File; Create Flags ☑ Save File; Combine Modes ☑ Save File; Dashboard Search ☑ Save File

Step Three: Convert MGF ☑ Save File; CFMID ☑ Save File

CFMID MATCHING CONTROLS

Parent Mass Accuracy ○ ppm ● Da 4

Fragment Mass Accuracy ○ ppm ● Da

Use CFMID Filtering? ○ Yes ● No

Negative Mode MGF File

Positive Mode MGF File