SAS® GLOBAL FORUM 2018

USERS PROGRAM

Mining Metabolites of Peptides and Antibody-Drug Conjugates from Mass Spectrometry Data using SAS® and Python

Hao Sun and Kristen Cardinal

April 8 – 11 | Denver, CO

#SASGF

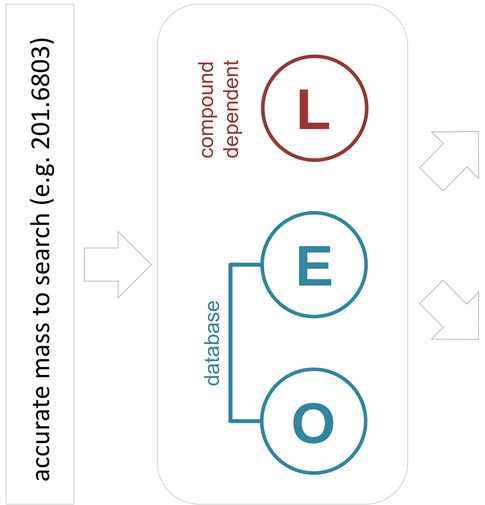
Mining Metabolites of Peptides and Antibody-Drug Conjugates from Mass Spectrometry Data using SAS® and Python

Hao Sun and Kristen Cardinal
Sun Prairie, Wisconsin; Colorado Springs, Colorado; United States

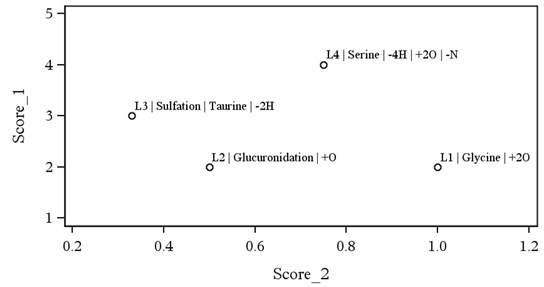
ABSTRACT

Metabolite identification of antibody-drug conjugate (ADC) and peptide drugs using mass spectrometry is challenging due to the complexity of their metabolism and catabolism reactions and lack of computing tools for mining complicated molecular ions and fragmentation patterns from high-resolution mass spectra. Mass spectrometric data are enormous and thus manual interpretation methods routinely used for small molecules are time-consuming and inefficient. Previously, we reported a SAS®-based application, AIR Binder, for dynamic visualization, data analysis and reporting of preclinical and clinical drug metabolism assays (PharmaSUG 2017, MWSUG 2017). We further expanded the application for mining raw accurate mass spectra data. Python scripts were developed for molecular ion and fragment searching and iteration. The algorithm, as integrated in *LEO*, was based on molecule "*L*inearization" (for cleavage reactions), "*E*xtension" (for conjugation reactions), and "*O*xidation" (for oxidation-reduction reactions), for accurate mass prediction. SAS® was used as a platform to analyze searching results and generate novel solutions for molecular structure elucidation. Overall, LEO was successfully applied for mining mass spectra of ADC and peptide metabolites with significantly improved productivity and efficiency.

WORKFLOW OF LEO FOR METABOLITE IDENTIFICATION

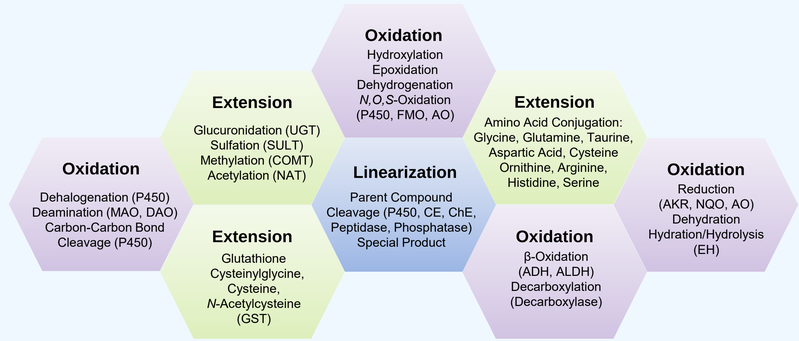


PPM	Score_1	Score_2	Solution
-3.83	2	1	L1 | Glycine | +2O
+1.51	2	0.5	L2 | Glucuronidation | +O
+1.33	3	0.33	L3 | Sulfation | Taurine | -2H
-2.77	4	0.75	L4 | Serine | -4H | +2O | -N



DESIGN OF LEO (LINEARIZATION, EXTENSION, OXIDATION)

LEO was designed to conduct automatic searching and iteration based on the experimental molecular ions observed from mass spectrometry, fragmentation patterns of each metabolite, and theoretical mechanism of reactions from the parent compound. It generates all possible solutions, simple or complicated, for comprehensive, accurate and robust identification and characterization of metabolites. LEO has three main components, "linearization", "extension", and "oxidation", which were used for its searching and iteration algorithm.




```
from itertools import product

ion = 201.6803
ionmass = ion - 1.007825 + 0.00054858 # for protonated ion
upper_ppm = ionmass * (1 + 0.000005)
lower_ppm = ionmass * (1 - 0.000005)

for l,e1,e2,e3,c,h,o in product(xrange(len(linearization)),
                                xrange(len(extension1)),
                                xrange(len(extension2)),
                                xrange(len(extension3)),
                                xrange(len(carbon)),
                                xrange(len(hydrogen)),
                                xrange(len(oxygen))):
       mass = linearization[l] + \
              extension1[e1] + \
              extension2[e2] + \
              extension3[e3] + \
              carbon[c] + \
              hydrogen[h] + \
              oxygen[o]
```

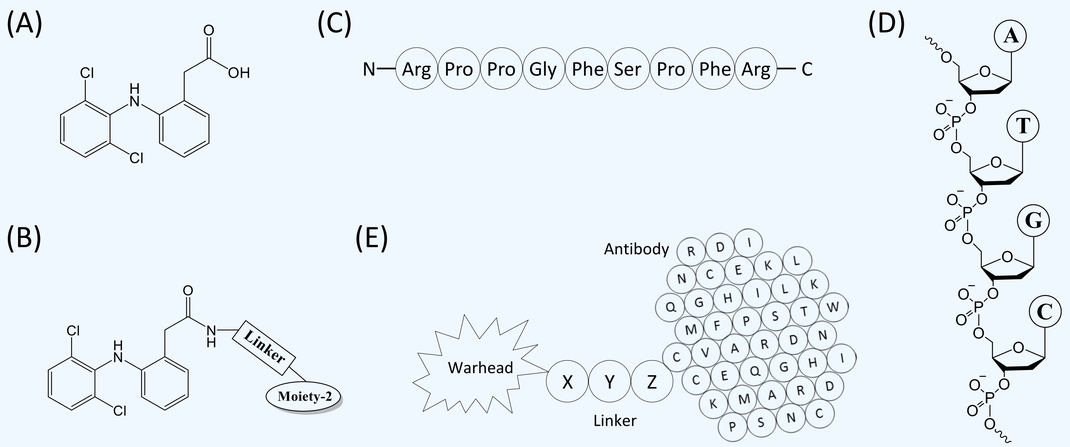
```
ppm = 1000000*(ionmass - mass)/mass

massstr = str(massdict[linearization[l]]) + '\t' + \
          str(massdict[extension1[e1]]) + '\t' + \
          str(massdict[extension2[e2]])  + '\t' + \
          str(massdict[extension3[e3]])  + '\t' + \
          str(int(carbon[c]/12))  + '\t' + \
          str(int(hydrogen[h]/1.007825))  + '\t' + \
          str(int(oxygen[o]/15.994915))  + '\t'

if mass > lower_ppm and mass < upper_ppm:
       print "\n"
       print mass
       print ppm
       print massstr
outfile.write(massstr + str(ppm) + "\n")
```

Mining Metabolites of Peptides and Antibody-Drug Conjugates from Mass Spectrometry Data using SAS® and Python

Hao Sun and Kristen Cardinal
Sun Prairie, Wisconsin; Colorado Springs, Colorado; United States

CHALLENGES OF MET ID WITH MASS SPECTROMETRY

The birth of mass spectrometry was around 1900s with J. J. Thompson's theoretical and experimental investigations on the conduction of electricity by gases. But until 1980s, mass spectrometry was still not extensively used for drug metabolism research. In early 80s, John B. Fenn and Koichi Tanaka developed soft desorption ionization methods for mass spectrometric analysis, and since then, mass spectrometry has gradually become the primary choice for bioanalysis and metabolite identification in drug discovery and development. For metabolite identification and characterization using mass spectrometry, molecular ions, such as protonated or deprotonated molecules, are selected first. High energy is then applied to break the molecule into fragments, and the resulting fragmentation patterns are fingerprints for metabolite identification and characterization. Small molecule drugs have been dominant for the last century. After data acquisition and processing from mass spectrometry, traditional small molecule metabolite identification was mainly accomplished by hand: draw a molecular structure, and match its mass based on theoretical biotransformation mechanisms. Recently, non-small molecule drug development has been a focus for pharmaceutical companies, which include antibodies, peptides, oligonucleotides, and ADCs (below, A-E). The expanded chemical space and more complex mass spectrometric profiles make a manual approach less efficient or not feasible.

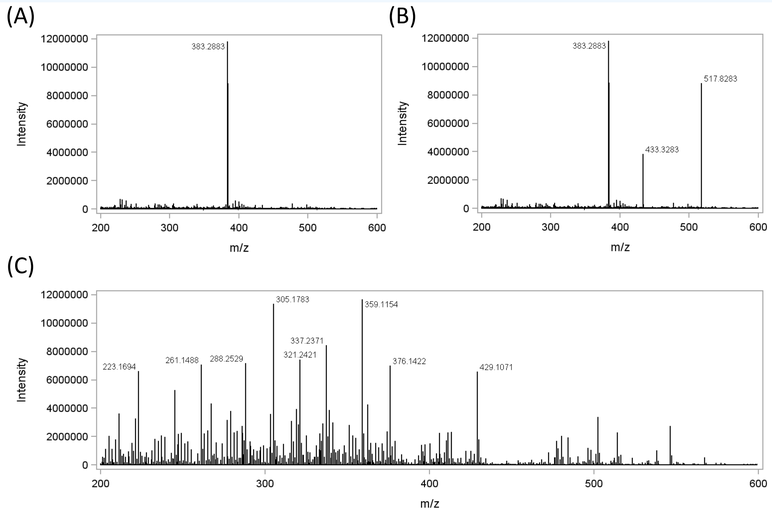


(A) a traditional small molecule, diclofenac; (B) more complicated design with multiple components; (C) peptide, bradykinin; (D) oligonucleotide; and (E) antibody-drug conjugate

MOLECULE ION SEARCHING AND INTERATION

High-resolution mass spectrometry is a key component for metabolite identification in complex biological samples, such as those from plasma and excreta. These samples contain a significant number of matrix ions in addition to the metabolites of interest. Without matrix interference, we expect to see a clean single molecular ion peak on a total ion chromatogram with a specific “mass over charge ratio” (m/z) for a metabolite (Figure A). With matrix ion interference, we may see multiple molecular ion peaks with similar abundances (Figure B). For non-small molecules, often we would see many molecular ion peaks for a single metabolite (Figure C). In practice, data-dependent scan method (also called information-dependent acquisition) is often used to scan top molecular ions and then acquire their MS/MS fragment spectra automatically. Manual interpretation of these molecular ions and the following fragmentation pattern elucidation may be challenging, especially for uncommon metabolites in vivo. As the first step of the development of LEO (Linearization, Extension, and Oxidation), a computing method was used to match the experimental and theoretical molecular ions individually, followed by a solution to provide definitive identification and elucidation of a molecular structure, based on fragmentation patterns.

ION CHROMATOGRAMS OF A SINGLE RADIOACTIVE PEAK



Mining Metabolites of Peptides and Antibody-Drug Conjugates from Mass Spectrometry Data using SAS® and Python

Hao Sun and Kristen Cardinal
Sun Prairie, Wisconsin; Colorado Springs, Colorado; United States

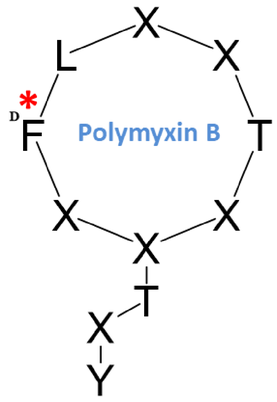
DATABASE IN LEO

```
leodict = {
        14.0157:        'Methylation',
        42.0106:        'Acetylation',
        79.9568:        'Sulfation',
        176.0321:       'Glucuronidation',
        305.0682:       'Glutathione Conjugation',
        176.0256:       'Cysteinylglycine Conjugation',
        119.0041:       'Cysteine S-Conjugation',
        145.0198:       'NAC S-Conjugation',
        103.0092:       'Cysteine',
        57.0215:        'Glycine',
        128.0586:       'Glutamine',
        156.1011:       'Arginine',
        87.0320:        'Serine',
        107.0041:       'Taurine'
        …               …       }
```

SAS POST-ANALYSIS

```
%macro leo (infile=in, outfile=out, n_ex=3);
        data &outfile (drop= i j leo);
           set &infile;
           array es(&n_ex) e1-e&n_ex;
           do i= 1 to &n_ex;
                do j= 1 to &n_ex-i;
                   if es(j) > es(j+1) then do;
                        leo = es(j);
                        es(j)=es(j+1);
                        es(j+1)=leo;
           end; end; end;
        run;
%mend;
```

LINEARIZATION FOR MOLECULE ION SEARCHING



X = α,γ-diaminobutyric acid
Y = methyloctanoic acid

1-mer $^{D}F^{*}$

2-mer $^{D}F^{*}$-L $^{D}F^{*}$-X

3-mer $^{D}F^{*}$-L-X $^{D}F^{*}$-X-X L-$^{D}F^{*}$-X

4-mer $^{D}F^{*}$-L-X-X $^{D}F^{*}$-X-X-X $^{D}F^{*}$-X-X-T L-$^{D}F^{*}$-X-X X-L-$^{D}F^{*}$-X

5-mer $^{D}F^{*}$-L-X-X-T $^{D}F^{*}$-X-X-X-T $^{D}F^{*}$-X-X-T-X *and* 5 *others*

6-mer $^{D}F^{*}$-L-X-X-T-X $^{D}F^{*}$-X-X-X-T-X *and* 10 *others*

7-mer $^{D}F^{*}$-L-X-X-T-X-X $^{D}F^{*}$-X-X-X-T-X-X *and* 14 *others*

8-mer $^{D}F^{*}$-L-X-X-T-X-X-X $^{D}F^{*}$-L-X-X-T-X-X(T) *and* 12 *others*

9-mer $^{D}F^{*}$-L-X-X-T-X-X(T)-X X-T-X-X(T-X-Y)-X-$^{D}F^{*}$ *and* 8 *others*

10-mer $^{D}F^{*}$-L-X-X-T-X-X(T-X-Y) $^{D}F^{*}$-L-X-X-T-X-X(T-X)-X *and* 5 *others*

11-mer $^{D}F^{*}$-L-X-X-T-X-X(T-X-Y)-X

VISULIZATION AND SCORING BY SAS

LEO has two forms of display in the current version: (1) a table display to list ppm, score_1, score_2 and specific mechanisms of the individual solution; (2) a figure display to illustrate two scoring components with each unique solution as data label/annotation using the ODS SGPLOT procedure. The data label placement is adjustable, which depends on the number/complexity of solutions to display.

The first scoring function (score_1) in LEO evaluates simplicity of biotransformation pathways for each solution. Each "extension" component is counted as 1 in the score. In addition, each category of atoms in "oxidation" component such as oxygen, hydrogen and carbon is also counted as 1 in the score. The score_1 is the sum of all scores for each solution. The second scoring function (score_2) evaluates the correlation between a metabolite and its fragments. It calculates the percent of the "extension" component for each solution that appears in pooled fragment solutions, after exactly the same procedure run by LEO. The best solution is expected to have a smaller score_1 (simpler biotransformation) and bigger score_2 (diagnostic fragmentation pattern), as visualized by LEO with PROC SGPLOT.

CONCLUSION

LEO (Linearization, Extension, and Oxidation) was designed and developed for mining metabolites with SAS® and Python, which demonstrated efficiency and improved productivity especially for the identification and characterization of non-small molecule drug metabolites such as ADCs and peptides. SAS® is a powerful and flexible platform for the integration and visualization of mass spectrometry data to develop industrial-specific solutions.

ACKNOWLEDGMENTS

The authors would like to thank Rich Voorman, Xiumin Liu, Bob Greene, and Nag Chemuturi for their discussion on and involvement of testing AIR Binder and LEO. The authors are also grateful to feedback and vision from our clients and friends across pharmaceutical and biotech companies, especially, Steve Alley and Anthony Lee from Seattle Genetics.

REFERENCES

1. Sun, H., Cardinal, K., Kirby, C., and Voorman, R. 2017. "AIR Binder: an automatic reporting and data analysis SAS® application for cytochrome P450 inhibition assay to investigate DDI." PharmaSUG Paper PO09: 1-8.
2. Sun, H., Cardinal, K., and Voorman, R. 2017. "AIR Binder 2.0: A dynamic visualization, data analysis and reporting SAS® application for preclinical and clinical ADME assays, pharmacokinetics, metabolite profiling and identification." MWSUG Paper PH04: 1-26.

Current Address: Hao Sun, PhD | Seattle Genetics | 21823 30th Dr SE, Bothell, WA 98021 | Office: 425-527-4681 | Email: hsun@seagen.com

Paper 2510-2018

Mining Metabolites of Peptides and Antibody-Drug Conjugates from Mass Spectrometry Data using SAS® and Python

Hao Sun, Sun Prairie, Wisconsin, United States
Kristen Cardinal, Colorado Springs, Colorado, United States

ABSTRACT

Metabolite identification of antibody-drug conjugate (ADC) and peptide drugs using mass spectrometry is challenging due to the complexity of their metabolism and catabolism reactions and lack of computing tools for mining complicated molecular ions and fragmentation patterns from high-resolution mass spectra. Mass spectrometric data are enormous and thus manual interpretation methods routinely used for small molecules are time-consuming and inefficient. Previously, we reported a SAS®-based application, AIR Binder, for dynamic visualization, data analysis and reporting of preclinical and clinical drug metabolism assays (PharmaSUG 2017, MWSUG 2017). We further expanded the application for mining raw accurate mass spectra data. Python scripts were developed for molecular ion and fragment searching and iteration. The algorithm, as integrated in *LEO*, was based on molecule "*L*inearization" (for cleavage reactions), "*E*xtension" (for conjugation reactions), and "*O*xidation" (for oxidation-reduction reactions), for accurate mass prediction. SAS® was used as a platform to analyze searching results and generate novel solutions for molecular structure elucidation. Overall, LEO was successfully applied for mining mass spectra of ADC and peptide metabolites with significantly improved productivity and efficiency.

INTRODUCTION

For many people in East Asia, a sip of alcohol will make their cheeks completely "red", which is often called "Asian Flush Syndrome". Scientifically, it is due to "abnormal" metabolism of alcohol. For the first step of alcohol metabolism, the metabolite acetaldehyde is formed by an enzyme called alcohol dehydrogenase. Accumulation of acetaldehyde can irritate facial tissues. The acetaldehyde metabolite can be further biotransformed to a "milder" acetic acid metabolite by aldehyde dehydrogenase in the "normal" populations. But for people with "Asian Flush Syndrome", their aldehyde dehydrogenase enzyme is less efficient, and at the same time, the efficiency of their alcohol dehydrogenase is much higher. As a result, a large amount of irritating aldehyde metabolite accumulates rapidly after alcohol intake, which causes facial flushing. For the science of drug metabolism, we identify metabolites of drugs to understand their effects on human body, which is extremely useful for the discovery and development of new therapeutic drugs with great efficacy and safety. In recent years, mass spectrometry has been predominantly applied for metabolite identification and characterization.

The birth of mass spectrometry was around 1900s with J. J. Thompson's theoretical and experimental investigations on the conduction of electricity by gases. But until 1980s, mass spectrometry was still not extensively used for drug metabolism research. In early 80s, John B. Fenn and Koichi Tanaka developed soft desorption ionization methods for mass spectrometric analysis, and since then, mass spectrometry has gradually become the primary choice for bioanalysis and metabolite identification in drug discovery and development. For metabolite identification and characterization using mass spectrometry, molecular ions, such as protonated or deprotonated molecules, are selected first. High energy is then applied to break the molecule into fragments, and the resulting fragmentation patterns are fingerprints for metabolite identification and characterization.

Small molecule drugs have been dominant for the last century. After data acquisition and processing from mass spectrometry, traditional small molecule metabolite identification was mainly accomplished by hand: draw a molecular structure, and match its mass based on theoretical biotransformation mechanisms. Recently, non-small molecule drug development has been a focus for pharmaceutical companies, which include antibodies, peptides, oligonucleotides, and ADCs (Figure 1). The expanded chemical space and more complex mass spectrometric profiles make a manual approach less efficient or not feasible.

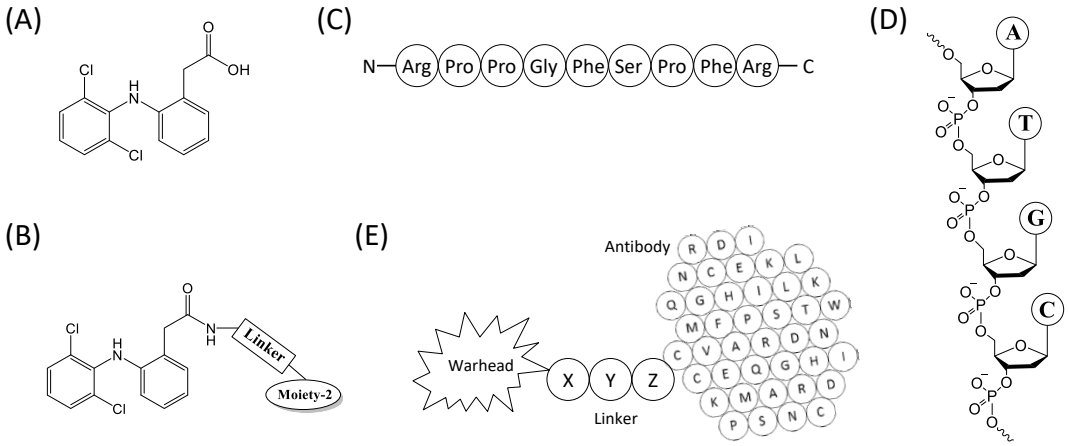


Figure 1 Metabolite identification challenges due to expanded chemical space in drug discovery
(A) a traditional small molecule, diclofenac; (B) more complicated design with multiple components;
(C) peptide, bradykinin; (D) oligonucleotide; and (E) antibody-drug conjugate

High-resolution mass spectrometry is a key component for metabolite identification in complex biological samples, such as those from plasma and excreta. These samples contain a significant number of matrix ions in addition to the metabolites of interest. Without matrix interference, we expect to see a clean single molecular ion peak on a total ion chromatogram with a specific "mass over charge ratio" (*m/z*) for a metabolite (Figure 2A). With matrix ion interference, we may see multiple molecular ion peaks with similar abundances (Figure 2B). For non-small molecules, often we would see many molecular ion peaks for a single metabolite (Figure 2C). In practice, data-dependent scan method (also called information-dependent acquisition) is often used to scan top molecular ions and then acquire their MS/MS fragment spectra automatically. Manual interpretation of these molecular ions and the following fragmentation pattern elucidation may be challenging, especially for uncommon metabolites *in vivo*. As the first step of the development of **LEO** (**L**inearization, **E**xtension, and **O**xidation), a computing method was used to match the experimental and theoretical molecular ions individually, followed by a solution to provide definitive identification and elucidation of a molecular structure, based on fragmentation patterns.

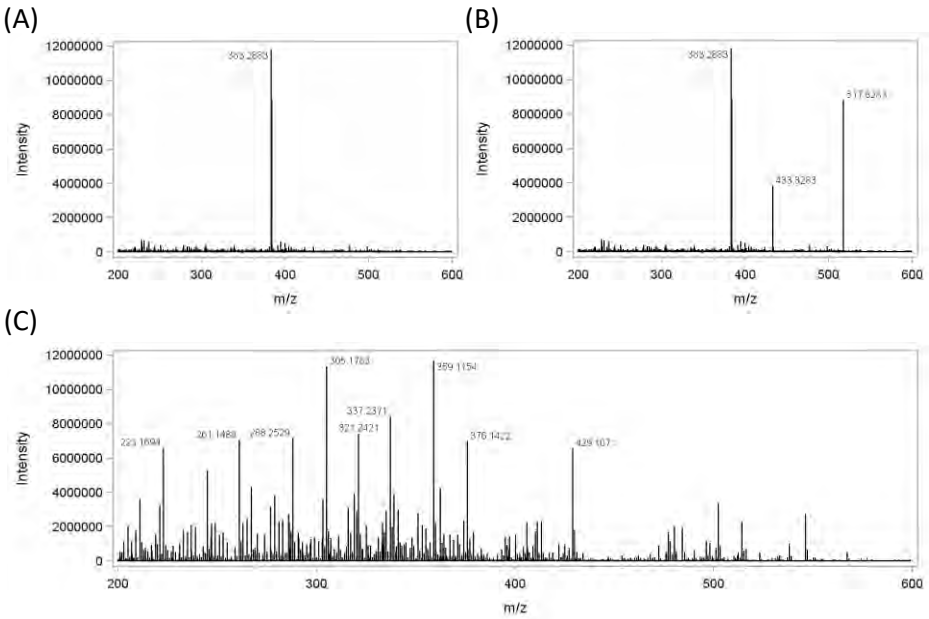


Figure 2 Total ion chromatogram of a selected single radioactive peak
(A) single molecular ion; (B) multiple molecular ions; and (C) many molecular ions

DESIGN OF LEO

LEO was designed to conduct automatic searching and iteration based on the experimental molecular ions observed from mass spectrometry, fragmentation patterns of each metabolite, and theoretical mechanism of reactions from the parent compound. It generates all possible solutions, simple or complicated, for comprehensive, accurate and robust identification and characterization of metabolites. LEO has three main components, “linearization”, “extension”, and “oxidation”, which were used for its searching and iteration algorithm (Figure 3).

“Linearization” refers to breaking the parent molecule into theoretically breakable portions, such as those metabolites via hydrolysis, dealkylation, or special pathways. The linearization process might be guided by the position of the radiolabeled moiety. The accurate mass of each linearized piece of a molecule is stored in Python lists and dictionaries. For example, for polymyxin B molecule, a theoretical 79 linearized parts were produced (Figure 4). “Extension” means various conjugation reactions catalyzed mainly by Phase II drug-metabolizing enzymes, such as glucuronidation, sulfation, methylation, acetylation, glutathione conjugation and downstream hydrolysis products, and amino acid conjugation. A part of Python dictionary in LEO for the conjugation reactions are listed below (*leodict*). “Oxidation” means various oxidation-reduction (redox) reactions, such as hydroxylation, epoxidation, dehydrogenation, *N*,*O*,*S*-oxidation, β-oxidation, decarboxylation, and reduction. The net mass difference of carbon, hydrogen, and oxygen atoms to reflect the redox reaction is stored in Python lists individually. Other reactions involved with mass changes of nitrogen (deamination), fluorine and chlorine (dehalogenation) atoms were treated in LEO similarly.

```
leodict = {
        14.0157:        'Methylation',
        42.0106:        'Acetylation',
        79.9568:        'Sulfation',
        176.0321:       'Glucuronidation',
        305.0682:       'Glutathione Conjugation',
        176.0256:       'Cysteinylglycine Conjugation',
        119.0041:       'Cysteine S-Conjugation',
        145.0198:       'NAC S-Conjugation',
        103.0092:       'Cysteine',
        57.0215:        'Glycine',
        128.0586:       'Glutamine',
        156.1011:       'Arginine',
        87.0320:        'Serine',
        107.0041:       'Taurine'
        …               …       }
```

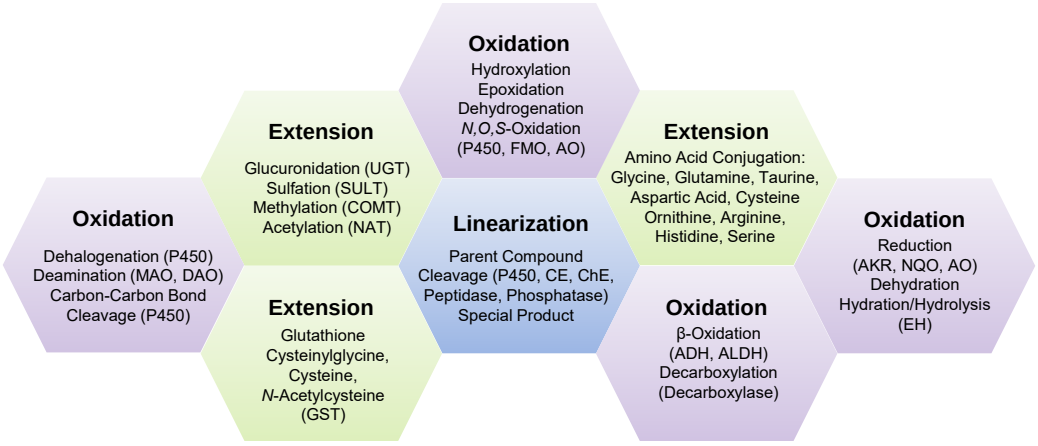


Figure 3 Design of LEO for mining metabolites with Python and SAS®

(UGT, UDP-Glucuronosyltransferase; SULT, Sulfotransferase; COMT, Catechol-*O*-Methyltransferase; NAT, *N*-Acetyltransferase; GST, Glutathione Transferase; P450, Cytochrome P450; FMO, Flavin Monooxygenase; CE, Carboxylesterase; ChE, Cholinesterase; AKR, Aldo-Keto Reductase; AO, Aldehyde Oxidase; EH, Epoxide Hydrolase; NQO, NADPH-Quinone Oxidoreductase; ADH, Alcohol Dehydrogenase; ALDH, Aldehyde Dehydrogenase; MAO, Monoamine Oxidase; DAO, Diamine Oxidase.)

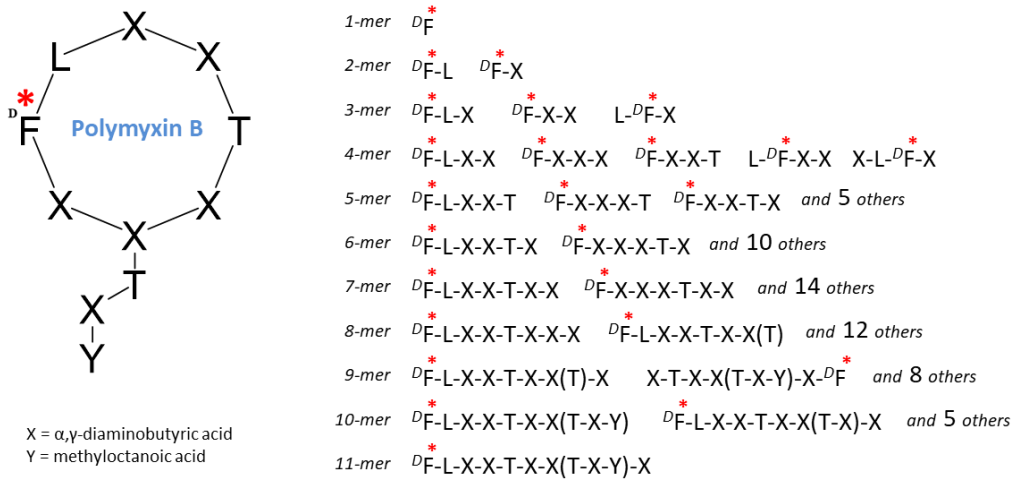
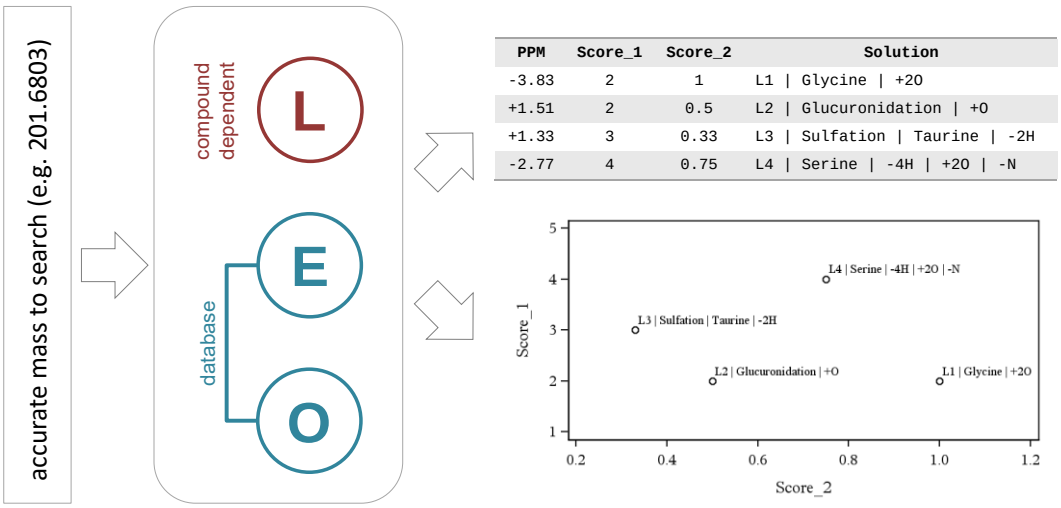


Figure 4 Linearization of polymyxin B for molecular ion searching in LEO
(* radiolabeled D-phenylalanine)

The theoretical accurate mass after the combination of linearization, extension and oxidation steps is calculated with the following nested *for* loops in Python using *itertools*. The mass range of searched solutions, as defined by ppm (parts per million), can be pre-defined (normally ±5 ppm). The reaction mechanisms stored in the Python dictionary are looked up and written as raw solutions ("*massstr" as follows*). The output file is used for further analysis and visualization by SAS®.

```
from itertools import product

ion = 201.6803
ionmass = ion - 1.007825 + 0.00054858 # for protonated ion
upper_ppm = ionmass * (1 + 0.000005)
lower_ppm = ionmass * (1 - 0.000005)

for l,e1,e2,e3,c,h,o in product(xrange(len(linearization)),
                                xrange(len(extension1)),
                                xrange(len(extension2)),
                                xrange(len(extension3)),
                                xrange(len(carbon)),
                                xrange(len(hydrogen)),
                                xrange(len(oxygen))):
        mass = linearization[l] + \
               extension1[e1] + \
               extension2[e2] + \
               extension3[e3] + \
               carbon[c] + \
               hydrogen[h] + \
               oxygen[o]

        ppm = 1000000*(ionmass - mass)/mass

        massstr = str(massdict[linearization[l]]) + '\t' + \
                  str(massdict[extension1[e1]]) + '\t' + \
                  str(massdict[extension2[e2]])  + '\t' + \
                  str(massdict[extension3[e3]])  + '\t' + \
                  str(int(carbon[c]/12))  + '\t' + \
                  str(int(hydrogen[h]/1.007825))  + '\t' + \
                  str(int(oxygen[o]/15.994915))  + '\t'

        if mass > lower_ppm and mass < upper_ppm:
                print "\n"
                print mass
                print ppm
                print massstr
                outfile.write(massstr + str(ppm) + "\n")
```



PPM	Score_1	Score_2	Solution
-3.83	2	1	L1 | Glycine | +2O
+1.51	2	0.5	L2 | Glucuronidation | +O
+1.33	3	0.33	L3 | Sulfation | Taurine | -2H
-2.77	4	0.75	L4 | Serine | -4H | +2O | -N

Figure 5 Workflow of LEO for mining metabolites with SAS and Python

The overall workflow of LEO is illustrated in Figure 5. It starts with an input of either a molecular ion of the metabolite or a fragment of that metabolite. The algorithm via summation of linearization, extension and oxidation pieces generates a list of solutions. However, the nested *for* loops in Python as described above could not differentiate the order of all "extension" components. SAS® is used to identify unique solutions during post-processing. The following SAS® macro LEO is used to re-order a list of *x* "extension" components (*n_ex*) for each solution. The dataset is further sorted by "*e1, e2, … ex*", and the unique solution is selected with the statement "*if last.ex*".

```
%macro leo (infile=in, outfile=out, n_ex=3);
       data &outfile (drop= i j leo);
          set &infile;
          array es(&n_ex) e1-e&n_ex;
          do i= 1 to &n_ex;
               do j= 1 to &n_ex-i;
                  if es(j) > es(j+1) then do;
                       leo = es(j);
                       es(j)=es(j+1);
                       es(j+1)=leo;
          end; end; end;
       run;
%mend;
```

LEO has two forms of display in the current version: (1) a table display to list ppm, score_1, score_2 and specific mechanisms of the individual solution; (2) a figure display to illustrate two scoring components with each unique solution as data label/annotation using the ODS SGPLOT procedure. The data label placement is adjustable, which depends on the number/complexity of solutions to display.

The first scoring function (score_1) in LEO evaluates simplicity of biotransformation pathways for each solution. Each "extension" component is counted as 1 in the score. In addition, each category of atoms in "oxidation" component such as oxygen, hydrogen and carbon is also counted as 1 in the score. The score_1 is the sum of all scores for each solution. The second scoring function (score_2) evaluates the correlation between a metabolite and its fragments. It calculates the percent of the "extension" component for each solution that appears in pooled fragment solutions, after exactly the same procedure run by LEO. The best solution is expected to have a smaller score_1 (simpler biotransformation) and bigger score_2 (diagnostic fragmentation pattern), as visualized by LEO with PROC SGPLOT.

Future work will include the development of "smarter" scoring functions for mapping fragments from mass spectrometry, parent and metabolite fragmentation pattern correlation analysis, and an automatic chemical structure "linearization".

CONCLUSION

LEO (Linearization, Extension, and Oxidation) was designed and developed for mining metabolites with SAS® and Python, which demonstrated efficiency and improved productivity especially for the identification and characterization of non-small molecule drug metabolites such as ADCs and peptides. SAS® is a powerful and flexible platform for the integration and visualization of mass spectrometry data to develop industrial-specific solutions.

ACKNOWLEDGMENTS

The authors would like to thank Rich Voorman, Xiumin Liu, Bob Greene, and Nag Chemuturi for their discussion on and involvement of testing AIR Binder and LEO. The authors are also grateful to feedback and vision from our clients and friends across pharmaceutical and biotech companies, especially, Steve Alley and Anthony Lee from Seattle Genetics.

REFERENCES

Sun, H., Cardinal, K., Kirby, C., and Voorman, R. 2017. “AIR Binder: an automatic reporting and data analysis SAS® application for cytochrome P450 inhibition assay to investigate DDI.” *PharmaSUG* Paper PO09: 1-8.

Sun, H., Cardinal, K., and Voorman, R. 2017. “AIR Binder 2.0: A dynamic visualization, data analysis and reporting SAS® application for preclinical and clinical ADME assays, pharmacokinetics, metabolite profiling and identification.” MWSUG Paper PH04: 1-26.

CONTACT INFORMATION

Hao Sun, PhD
118 Cobham Ln
Sun Prairie, Wisconsin 53590
United States

Email: CYP2F1@gmail.com
Phone: 801.541.0884

Current Address: Hao Sun, PhD | Seattle Genetics | 21823 30th Dr SE, Bothell, WA 98021 | Office: 425-527-4681 | Email: hsun@seagen.com