

# Data-Processing Workflow for Quantifying identified and unidentified Features across measured Stool Samples

Dominik Lux, Dr. Svitlana Rozanova, Prof. Dr. Martin Eisenacher, Prof. Dr. Katrin Marcus-Alic  
{dominik.lux, svitlana.rozanova, martin.eisenacher, katrin.marcus}@rub.de

## Abstract

### Introduction:

Mass spectrometry based stool proteomics enables the quantitative analysis of host and microbial proteins, providing a better understanding of host microbe interactions and the mechanisms involved in the development of various diseases. However, the high complexity and heterogeneity of stool make both sample processing and analysis of the obtained MS-data non-trivial. Thus, the complexity of the samples requires large or specially developed databases for spectra identification, which can lead to an overestimation of the FDR and a low identification yield. The high biological heterogeneity also leads to unreliable quantification results.

### Method:

With our work, we describe a data-processing workflow for quantification of stool samples, using already well-established tools. To achieve this, we use Nextflow as the workflow engine, converting measured samples into an open-data-format with ThermoRawFileParser, searching spectra using Comet and applying the feature detection and matching between samples, using identified precursors with OpenMS. To counter the FDR-overestimation and to increase the ID yield, we designed a peptide-FASTA-file containing only unique peptides of around 1000 species which have been found in the human gut.

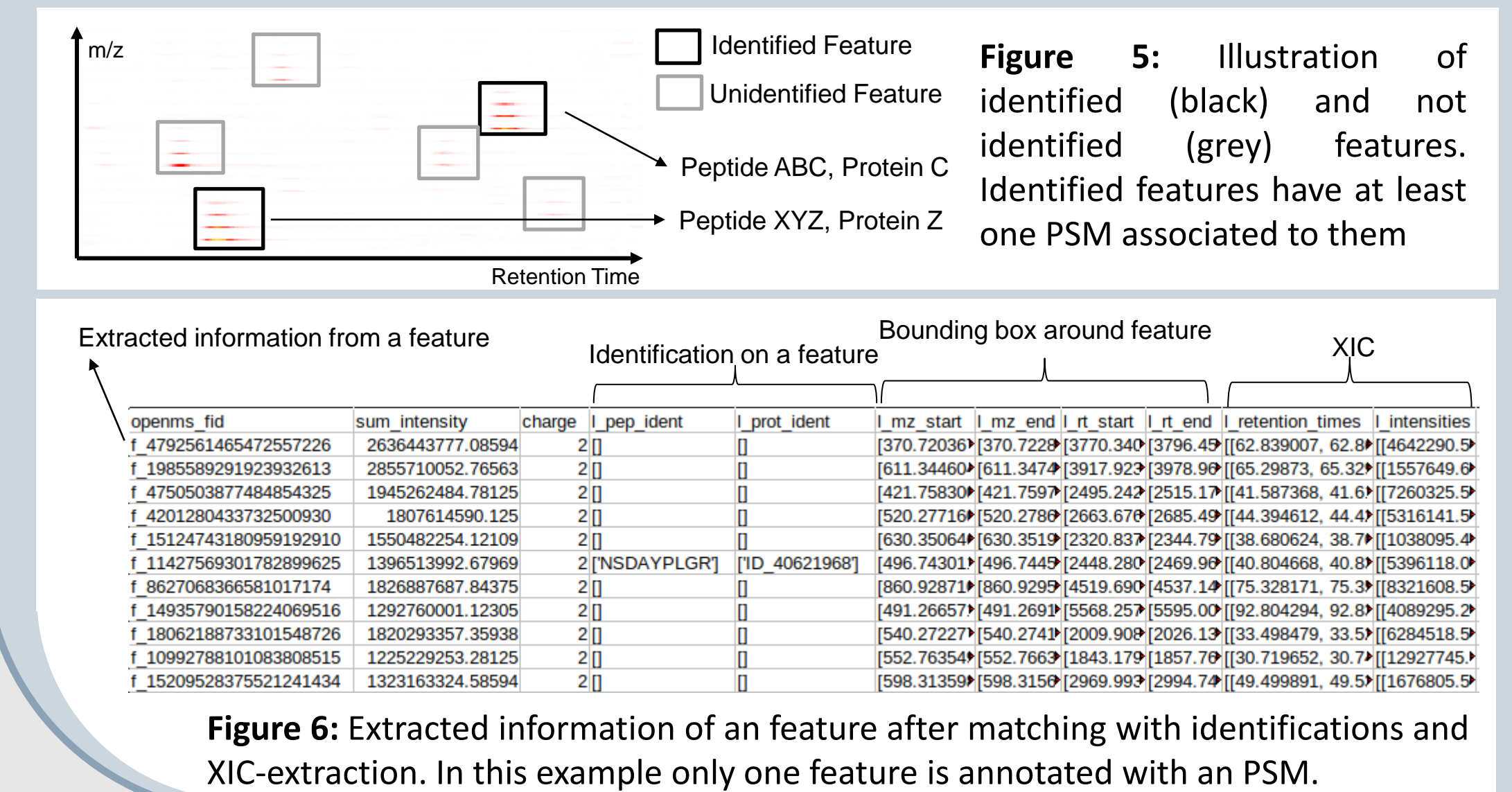
### Results:

Preliminary results show, that a matching between features in datasets is possible, using only the identified precursors. Further, intensities of identified as well as of unidentified features can be extracted and used for quantification across measured samples. Since this workflow is in an early state, it remains of special interest to verify each step to check for its reliability. But since it provides quantification information of identified and unidentified features, such a workflow could prove itself very useful for complex and heterogeneous samples in metaproteomics like stool where the identification yield is low.

## Match-Finding and XIC-Extraction

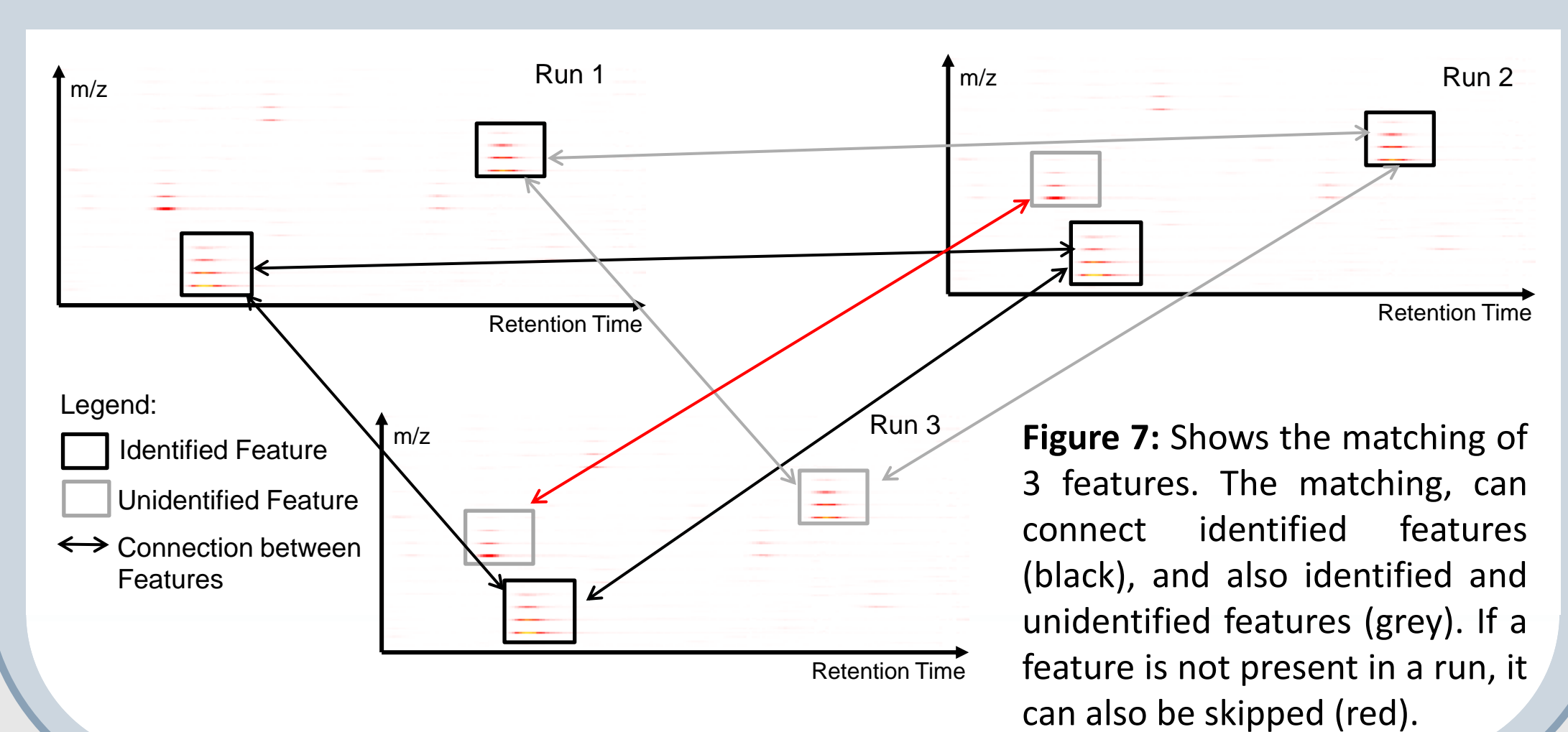
With the help of pyOpenMS [4], we generate a so-called idXML-file of the identification results. The matching between found features at the MS1-level and the idXML-file can be then done with the "IDMapper"-node provided by OpenMS [5], yielding featureXML-files of features, either with or without annotated identifications (Figure 5).

Further, we implemented a python-script, parsing through the resulting featureXML-files feature-wise. Per each feature we extract the annotated information and save them in a table-format. Additionally, the XIC-extraction functionality provided by the ThermoRawFileParser [6] is used, extracting the XIC of the found feature. An example can be seen in Figure 6. This step returns tuples of featureXML- and CSV-files.

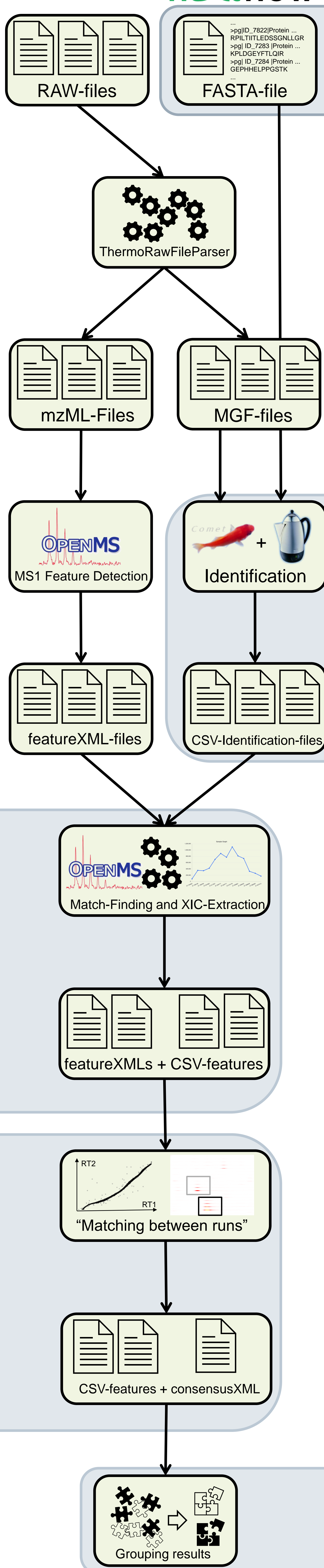


## Map-Alignment and Feature-Matching

Using from the input only the featureXMLs, we use the "MapAlignerTreeGuided"-node from OpenMS [5] to correct shifts and distortions at the retention time. This implementation uses the peptide identifications annotated on the featureXMLs for correction across runs. Afterwards the "FeatureLinkerUnlabeled"-node is used, generating a so-called "consensusXML", describing connected features. The Figure 7 illustrates visually, what the consensusXML contains.



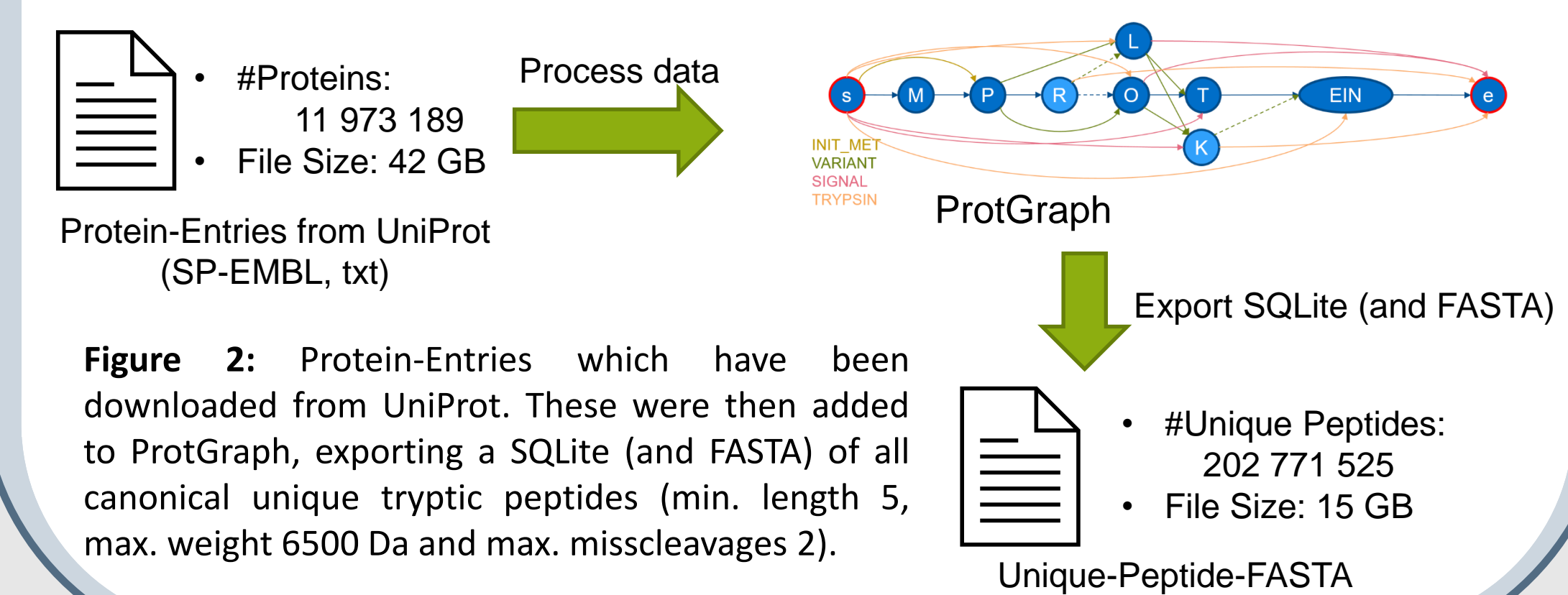
## Data processing Workflow



## FASTA-Generation

In metaproteomics, the species within a sample are not exactly known. Unlike in a common proteomics workflow, where a sample of a specific species is measured and analyzed, selecting a protein-database is a non-trivial task. Selecting a good-fitting protein-database is crucial, since if not well-chosen, all subsequent steps which depend on the identification results can perform worse or yield not reliable results.

The authors from Mirjana and Willem [1] provide a list of ~1000 species, which we set as our protein-database. We downloaded the species from UniProt using the provided API. Additionally we downloaded and added the human proteome into this protein-database. Figure 2 shows how we processed the proteins and provides some characteristics of the FASTA-files, which we used in the final data processing workflow.



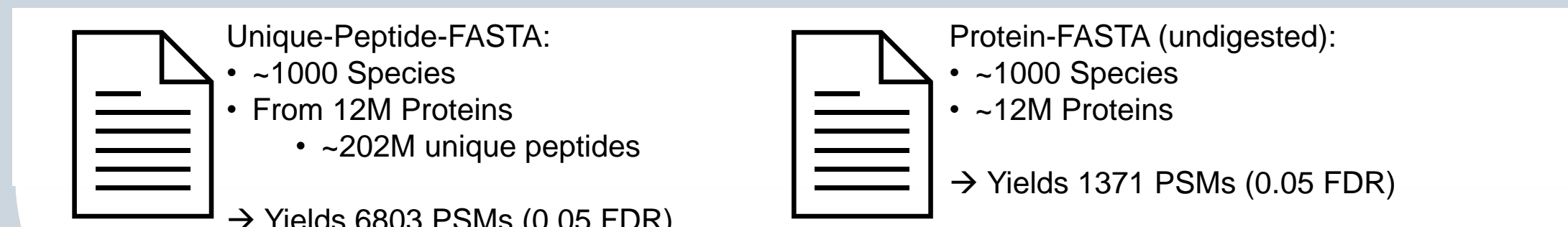
## Identification

The identification was done by Comet [2], with the "decoy\_search"-setting set to 1, which generates the reverse sequence for each entry in the FASTA-file as a decoy. Additionally to improve the number of identifications and to counter the FDR-Overestimation, we included Percolator [3], which rescores the identification results. The q-value was then calculated and we selected a cutoff of 5%. The identification step returns a CSV-file, containing specific column for subsequent steps. An overview of the columns can be seen in Figure 3.

This step was tested on two FASTA-files. The first FASTA-file was generated as in Figure 2. The second FASTA-file was generated by downloading the canonical FASTA-entries from UniProt directly (using ~1000 species as above, but also including the human proteome). Figure 4 summarizes the identification results.

psm_id	protein	peptide	quality	fasta_id	fasta_desc	used_score	charge	retention_time	exp_mass_to_charge
CEX83844_E1_R1_9519_3	1	INCEITITGGAGHQAALGK	0.0	201925901	ADA1C3H9B3(10511068.misschg.0)	3.060209	3	2152.2	605.41730701667
CEX83844_E1_R1_32043_2	1	ECQAEVFTVGAQEPFAEK	0.0	89417601	ADA1H8DM4(1129.misschg.0)	2.938	2	5932.2	1106.48248535
CEX83844_E1_R1_21985_2	1	SPEQFEQEEISTW	0.0	428479531	ADA173S12(1294296.misschg.0)	2.868	2	4228.2	862.897521035
CEX83844_E1_R1_45311_2	1	KVQAGSTGATGK	0.0	158479531	ADA1ZNUHWT(103118.misschg.1)	2.856	2	3111.4	686.92536535
CEX83844_E1_R1_8185_3	1	GGSDSDVTPSTNNVLDLSNAK	0.0	67388578	ADA174E5GX(167188.misschg.1)	2.835	3	1263.3	762.02848701667
CEX83844_E1_R1_23353_2	1	VGEPTGAGVQVGEQK	0.0	71457365	ADA174E5GX(167188.misschg.1)	2.835	2	4287.1	928.90235535
CEX83844_E1_R1_35388_3	1	ANTIEGDRVPLGQADQGSFTVK	0.0	138474420	ADA1ZNUHWT(103118.misschg.1)	2.682	3	6504.1	872.78948701667
CEX83844_E1_R1_23841_3	1	VLDGVHNSQVETGTTTLCSTSVLR	0.0	13828655	ADA1H8DM4(1129.misschg.0)	2.546	3	6255.2	1024.8734701667
CEX83844_E1_R1_34082_3	1	DLVDTGAVGNGEQLWAK	0.0	13828655	ADA1ZNUHWT(103118.misschg.0)	2.552	3	6279.4	760.061748701667
CEX83844_E1_R1_24325_2	1	LDLDTGQVAENQDPLK	0.0	132403185	ADA1ZNUHWT(103118.misschg.0)	2.528	2	4622.6	522.459899935
CEX83844_E1_R1_32585_3	1	VVNDIEEWSGQTEPFATFNK	0.0	138287119	ADA1ZNUHWT(103118.misschg.1)	2.517	3	6025.2	953.451548701667
CEX83844_E1_R1_22259_3	1	EGADGAGVCDVYNEEAATNAVK	0.0	71240794	ADA174YKMS(5579.misschg.0)	2.509	3	4274.4	897.746448701667
CEX83844_E1_R1_130531_3	1	TAACEKASKEKLEKPLATATGDSYAAAR	0.0	138492149	ADA1ZNUHWT(103118.misschg.0)	2.381	3	6182.2	1103.86648701667
CEX83844_E1_R1_21393_2	1	MTMGANITFAGDTLLK	0.0	84172724	ADA1G8VB(2646661.misschg.0)	2.374	2	4220.5	838.40496535
CEX83844_E1_R1_26125_2	1	VWYEDDPAFLVDVNR	0.0	71342239	ADA174YKMS(5579.misschg.0)	2.355	2	6625.4	1124.48959535
CEX83844_E1_R1_12021_2	1	VRENTSDQAEYK	0.0	22872889	ADA1ZNUHWT(103118.misschg.0)	2.348	2	2568.2	862.924848535
CEX83844_E1_R1_35519_3	1	VGLVDLITFTDEAVPTK	0.0	11266380	ADA1Y3V12(76.93.misschg.0)	2.347	2	6526.2	1013.51007035
CEX83844_E1_R1_124111_4	1	VDEAVQDTAEYK	0.0	13241114	ADA1ZNUHWT(103118.misschg.1)	2.262	3	3902.2	632.314248701667
CEX83844_E1_R1_12284_3	1	NAHSAQSVQVSTNDNFATPK	0.0	23026354	ADA1ZNUHWT(103118.misschg.0)	2.252	3	2609.6	828.70348701667
CEX83844_E1_R1_23730_2	1	YISYTVSSDYSEK	0.0	138280755	ADA1ZNUHWT(103118.misschg.0)	2.25	2	4522.2	869.512414035

**Figure 3:** PSMs from 1 RAW-file, showing all the columns, which are generated along this step. The columns: "retention\_time", "charge" and "exp\_mass\_to\_charge" are needed to determine if and when a PSM is inside a feature.



## Grouping Results

After features have been matched, we use the CSV-files and group the entries according to the generated matches described in the consensusXML. The final table contains per column, data about a run and an attribute and continues with the next runs, then with the next attribute. Essentially all columns from previous steps are saved here. Figure 8 shows a part of the final result of the pipeline.

Intensities of each individual run	Identifications (if any) of each individual run
<p>Run 1</p> <p>Run 2</p> <p>Run 3</p>	<p>Run 1</p> <p>Run 2</p> <p>Run 3</p>

**Figure 8:** Final result of the Pipeline. Only intensities and peptide identifications are illustrated.

**References:**  
[1] Mirjana Rajčić-Stojanović, Willem M. de Vos, FEMS Microbiology Reviews, 2014, <https://doi.org/10.1111/1574-6976.12075>  
[2] Jimmy K. Eng, Talmima A. Jahan, Michael R. Hoopmann, Proteomics, 2013, <https://doi.org/10.1002/pmic.201200439>  
[3] Matthew The, et al., J. Am. Soc. Mass Spectrom., 2016, <https://doi.org/10.1007/s13361-016-1460-7>  
[4] Röst HL, Schmitt U, Aebersold R, Malmström L, Proteomics, 2014, <https://doi.org/10.1002/pmic.201300246>  
[5] Röst HL, Sachsenberg T, Alche S, Bielow C et al., Nat Meth., 2016, <https://doi.org/10.1038/nmeth.3959>  
[6] Niels Hulstert, et al., Journal of Proteome Research, 2020, <https://doi.org/10.1021/acs.jproteome.9b00328>