The Threshold Avoiding Proteomics Pipeline (TAPP) for LC-MS/MS pre-processing

The Threshold Avoiding Proteomics Pipeline (TAPP) is being developed to quantitatively pre-process, explore and annotate LC-MS/MS data, with a special focus on full data traceability and the quantification of low intensity signals. At its most basic level, LC-MS/MS data can be seen as a 2D image, where compounds are separated based on their chemical properties (e.g. hydrophobicity) by liquid chromatography (LC) as well as their weight by mass spectrometry (MS). These compounds appear on the image as 2D Gaussian-like peaks, and it is the pipeline's job to detect and quantify them. Many interesting computational challenges are present in this data, and we are looking for people with a strong background in programming (C++, Python, R), computer science, algorithms, and/or bioinformatics. Previous knowledge of LC-MS instrumentation, biology and chemistry is not required, but it would be appreciated (if not available we will provide an introduction to the level that is necessary to understand the data structure and properties). Ideally the candidate(s) should be familiar with software version control systems (Git) and cross platform development (Windows/Mac/Linux).

We have a number of self-contained projects within the TAPP pipeline that could be interesting candidates for a bachelor or master level project and thesis. Some of which include, but is are limited to:

- Implementation of a faster stream-oriented XML parser to read/write standard mass-spectrometry files (mzXML, mzML, mzIdentML, mzDB, etc.) into the internal TAPP binary data structures.

- Seamless integration of proteomics/metabolomics search engines within the pipeline. Currently we rely on external tools (e.g. SearchGUI, PeptideShaker) to generate mzIdentML files that we use to annotate the detected peaks, but in addition to that, we would like to have the option to perform the search within our own framework.

- Study and implementation of compression algorithms for reading and storing binary data on disk. The current binary serialization functions allow the storage of many of the internal data structures into disk, but some of these files are quite large, which in turn slows down some of the intermediate steps of the pipeline. By using compression, we should be able to reduce the disk space impact of the intermediate data formats, while still maintaining full data traceability for re-analysis and exploration.

- Implementation of an easy to use GUI interface. The pipeline currently provides Python bindings for the C++ high performance functionality, as well as describing a default pipeline in Python that takes a JSON string for the parametrization. The GUI should be cross platform (Windows, Mac, Linux) and use the user inputs to generate the JSON parameters. The GUI can be written in C++ or Python, although the latter is preferred, and we should be able to bundle it with the pipeline in a single executable file for ease of distribution.

- Addition of more unit tests. We are using Doctest for testing a limited set of the C++ functions, but we would like to expand both the number and quality for a wider code coverage.

- Implementation of R bindings following the existing Python bindings. We use Pybind11 for the generation of Python bindings, and would like to use the API exposed from C++ to have the same functionality in the R programming language.

- Exploration of N-dimensional space partitioning techniques for quick data access (e.g. KD-trees).

- Indexing of binary data files to enable reading data only for the sections we are interested in, instead of the entire file.

- Advanced visualization using GPU powered technologies, such as OpenGL or Vulkan to process and/or visualise large LC-MS/MS images with extensive annotation.

We are located at A. Deusinglaan 1, 9713 AV Groningen (ERIBA building, 6th floor). The projects can be started at any time. If you are interested, feel free to contact Prof. Dr. Peter L. Horvatovich <p.l.horvatovich@rug.nl> or Alejandro Sánchez Brotons <a.sanchez.brotons@rug.nl> to discuss the scope and availability of projects.