bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

Fast open modification spectral library searching through approximate nearest neighbor indexing

Wout Bittremieux[1,2,3], **Pieter Meysman**[1,2], **William Stafford Noble**[3,4], **Kris Laukens**[1,2]

[1]Department of Mathematics and Computer Science, University of Antwerp, 2020 Antwerp, Belgium; [2]Biomedical Informatics Network Antwerpen (biomina), 2020 Antwerp, Belgium; [3]Department of Genome Sciences, University of Washington, Seattle WA 98195, USA; [4]Department of Computer Science and Engineering, University of Washington, Seattle WA 98195, USA

Corresponding author: wout.bittremieux@uantwerpen.be, kris.laukens@uantwerpen.be

Abstract

Open modification searching (OMS) is a powerful search strategy that identifies peptides carrying any type of modification by allowing a modified spectrum to match against its unmodified variant by using a very wide precursor mass window. A drawback of this strategy, however, is that it leads to a large increase in search time. Although performing an open search can be done using existing spectral library search engines by simply setting a wide precursor mass window, none of these tools have been optimized for OMS, leading to excessive runtimes and suboptimal identification results.
Here we present the ANN-SoLo tool for fast and accurate open spectral library searching. ANN-SoLo uses approximate nearest neighbor indexing to speed up OMS by selecting only a limited number of the most relevant library spectra to compare to an unknown query spectrum. This approach is combined with a cascade search strategy to maximize the number of identified unmodified and modified spectra while strictly controlling the false discovery rate, as well as a shifted dot product score to sensitively match modified spectra to their unmodified counterparts.
ANN-SoLo outperforms the state-of-the-art SpectraST spectral library search engine both in terms of speed and the number of identifications. On a previously published human cell line data set, ANN-SoLo confidently identifies 40 % more spectra than SpectraST while achieving a speedup of an order of magnitude.
ANN-SoLo is implemented in Python and C++. It is freely available under the Apache 2.0 license at `https://github.com/bittremieux/ANN-SoLo`.

1 Introduction

Although mass spectrometry (MS) is a very powerful technique to characterize proteins in complex biological samples, a significant portion of the thousands of spectra that are typically generated during a shotgun proteomics experiment cannot be confidently identified. In many cases a spectrum cannot be identified because it was generated by a peptide that contains one or more post-translational modifications (PTMs) [15]. If a particular modification has not been specified in the search settings, then spectra corresponding to peptides harboring this modification will be assigned an incorrect amino acid sequence. Because such false hits tend to receive higher scores than other false positive matches, these missed modifications have a detrimental effect on the identification performance [8].

On the one hand, protein modifications can be an artifact of the MS process because sometimes they are introduced during sample preparation [7]. A common example is alkylation by using iodoacetamide, which leads to the attachment of a carbamidomethyl group to cysteine residues, preventing the denatured proteins from reforming disulfide bridges. On the other hand, naturally occurring PTMs can be very interesting from a biological perspective as they often play important roles in many cellular processes [60].

Unfortunately, although MS techniques have become quite mature, comprehensive identification of modified proteins in complex samples remains challenging [1, 38]. In the traditional search paradigm, all modifications of interest have to be explicitly specified in the search settings to correctly identify the spectra that include one or more of these modifications. This requirement leads to a significant search space increase because, for each peptide, both its unmodified version and all possible modified variants need to be considered, which results in an increased computational load and reduced sensitivity. Consequently, only a limited selection of the most prevalent modifications are commonly considered.

Alternatively, open modification searching (OMS) is a powerful strategy to identify modified spectra. Whereas traditionally only candidates that fall within a limited mass window around the query spectrum's precursor mass are considered as a potential match, during OMS a very wide precursor mass window exceeding the delta mass induced by a PTM is used. This approach makes it possible to compare a modified query spectrum to its unmodified variant [2, 46]. As such, during an open search all possible protein modifications for which the mass difference falls within the precursor mass window, which is typically on the order of several hundreds of Dalton, are implicitly considered, including PTMs, amino acid substitutions, cleavage variants, etc. Afterwards, the presence and type of each modification can be derived from the difference between the observed precursor mass and the mass of the unmodified peptide.

Although OMS makes it possible to identify a wide

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

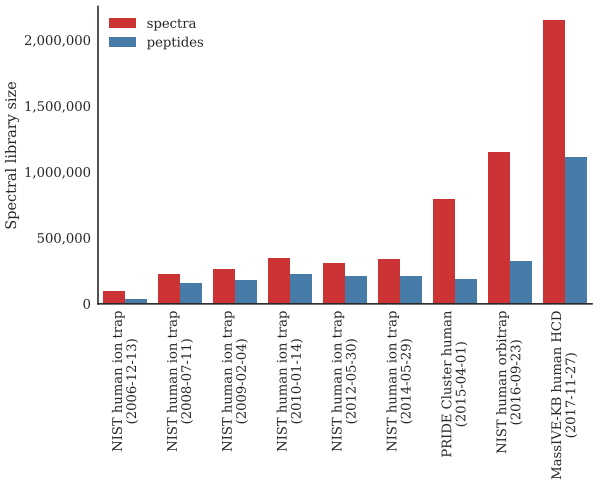


Figure 1: The size of spectral libraries has increased as more high-quality data sets have become available in public data repositories. Whereas traditionally spectral libraries were explicitly curated and compiled, e.g. by the National Institute of Standards and Technology (NIST), a recent alternative has been to automatically generate large spectral libraries on a repository-wide scale, e.g. based on all data sets in the PRoteomics IDEntifications (PRIDE) and MassIVE databases [31, 32].

range of spectra containing diverse modifications, the use of a very wide precursor mass window leads to a drastically increased search space. Consequently, compared to a standard search, the computational cost for an open search is orders of magnitude higher. A popular approach to keep OMS computationally feasible is to use spectral libraries [3, 11, 33, 42, 62]. Because spectral libraries only contain previously observed peptides, the search space is often substantially restricted in comparison to sequence database searching strategies that consider all theoretically possible peptides [30]. On the other hand, with the increasing availability of high quality data sets in public data repositories [50], spectral libraries have grown substantially (figure 1). Indeed, for some well-studied organisms the spectral library size can rival the size of the sequence database.

Here we present the **A**pproximate **N**earest **N**eighbor **S**pectral **L**ibrary (ANN-SoLo) search tool, which has been optimized for fast and accurate open modification spectral library searching. Using a cascade search strategy [37] ANN-SoLo first identifies any unmodified peptides, followed by an open search to identify the modified peptides. During this open search ANN-SoLo uses an approximate nearest neighbor (ANN) index to efficiently find a limited set of the most similar library spectra for each query spectrum. During the open search the shifted dot product is used to accurately match modified query spectra to their unmodified library counterpart by taking into account peak shifts caused by a modification [11, 33].

Although some approaches have previously been proposed to speed up spectral library searching, including parallelizing spectral matching using graphics processing units (GPUs) [6] and candidate filtering based on a shared peak count [59], these approaches did not deal with the specific challenges posed by OMS. In contrast, the Liberator and MzMod software tools are based on the Apache Spark framework for distributed data processing to massively parallelize open spectral library searching [33]. Although these big data tools make it possible to process large amounts of spectral data, they require a specialized cluster or cloud infrastructure.

Additionally, several software tools have recently been developed to speed up OMS when using a sequence database instead of a spectral library. MSFragger uses an index of theoretical fragments to quickly compute the number of shared fragment ions between a query spectrum and theoretical spectra [39]. SpecOMS uses an FP-tree-like data structure [9], called SpecTrees, to encode the number of shared masses between all spectra [17]. Sequence tags are a popular approach to restrict the search space as well. PIPI uses sequence tags of length 3 to perform a fuzzy tag-based filtering [63]. TagGraph uses an FM-index to filter candidates based on substrings of *de novo* derived sequences [21]. Finally, Open-pFind combines tag-based filtering to speed up open searches with a subsequent standard search including highly abundant variable modifications in a two-pass strategy [14].

Generally MS/MS spectrum identification can be considered a nearest neighbor task: for a given query spectrum the most similar database spectrum, i.e. its nearest neighbor, be it a real spectrum during spectral library searching or a theoretical spectrum during sequence database searching, has to be retrieved. Consequently these nearest neighbor queries can be sped up by making use of index structures. For example, a multiple vantage point tree [51] and locality-sensitive hashing (LSH) [23] have previously been proposed to speed up sequence database searching.

These various tools that have recently been developed for efficient OMS on the one hand and the application of multidimensional indexing techniques to speed up spectral identification on the other hand have so far exclusively focused on sequence database searching. Here, we show how ANN indexing can be used to speed up open spectral library searching. Our ANN-SoLo tool is able to efficiently search very large spectral libraries and sensitively identify spectra containing any modification, outperforming other spectral library search engines in both speed and the number of identified spectra.

ANN-SoLo is implemented in Python and C++. It is freely available as open source under the permissive Apache 2.0 license at `https://github.com/bittremieux/ANN-SoLo`.

2 Methods

2.1 Cascade spectral library searching

Spectral library searching works by comparing experimental, unknown query spectra to previously observed, known spectra in the spectral library. To identify a query spectrum, its best matching library spectrum is found and is assigned the corresponding peptide sequence [30,

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

53]. Finding the closest matching library spectrum for a given query spectrum can be divided into two steps: (i) a candidate selection step during which a subset of spectra in the spectral library are selected as candidate matches, and (ii) a candidate ranking step during which, for each candidate spectrum, a spectrum–spectrum match (SSM) score is calculated to quantify the similarity between the two spectra. Subsequently the candidate match with the highest score is used as the identification for the query spectrum.

To perform an open search ANN-SoLo employs a cascade search strategy consisting of two levels, which allows it to maximize the number of identified spectra while strictly controlling the false discovery rate (FDR) [37]. In the first level of the cascade search a small precursor mass window is used to identify any unmodified spectra. The resulting SSMs are filtered on FDR, and the confident SSMs below the FDR threshold are retained. Next, the SSMs exceeding the FDR threshold are passed on to the second level of the cascade search, in which a wide precursor mass window is used to identify unmodified spectra. The resulting SSMs are filtered on FDR as well and combined with the accepted SSMs from the first level to form the final set of spectrum identifications.

Below, we describe how ANN-SoLo addresses the candidate selection and candidate ranking steps during both levels of the cascade open search, and describe how it achieves both speed and accuracy.

2.1.1 Spectrum preprocessing

Prior to spectral library searching both the query spectra and library spectra are similarly preprocessed to represent the spectra in a uniform way and discard low-quality spectra [52]. Peaks corresponding to the precursor ion and noise peaks with an intensity below 1 % of the intensity of the most intense peak are removed and, if applicable, the spectrum is further restricted to its 50 most intense peaks [41]. After peak removal, any spectrum that contains fewer than 10 peaks remaining or with a mass range less than 250 Da is discarded. Finally, peak intensities are rank transformed to de-emphasize overly dominant peaks [42].

2.1.2 Candidate selection

Typically, the candidate selection step consists of a precursor mass filter, i.e. only the library spectra whose precursor mass falls within a narrow window around the query spectrum's precursor mass are considered as candidates. Especially for modern high-resolution instruments, which can report masses with a (sub-)ppm accuracy, the number of considered candidate spectra can be very small. However, when a wider precursor mass window is used, as in the case of open searches, the number of candidates that are selected can increase by several orders of magnitude. In this case the precursor mass window will not be an effective filter, and the search time will increase accordingly.

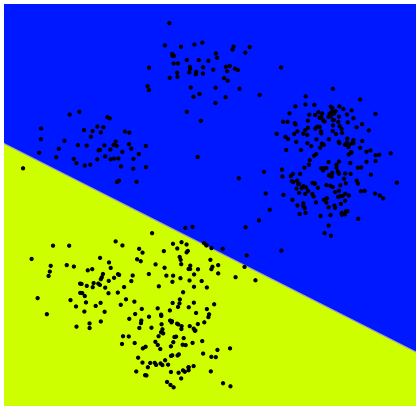
During the first level of its cascade search ANN-SoLo uses a small precursor mass window, and the search proceeds in the standard fashion. In contrast, during the second level of its cascade search ANN-SoLo uses an ANN index consisting of an ensemble of random projection trees [5] to efficiently filter the library spectra based on their similarity to the query spectra.

To construct the ANN index each library spectrum is vectorized to represent it as a point in a multidimensional space. A spectrum is converted into a sparse vector by dividing it into mass bins of 1 Da and assigning its peak intensities to their corresponding mass bins, after which the vector is normalized to have unit length. In case multiple peaks in the mass spectrum are assigned to the same mass bin their intensities are summed. Next, the vectors for all library spectra are used to build a binary index tree (figure 2). This is done by recursively partitioning the data space into two subspaces using random split hyperplanes. Concretely, two points are randomly sampled to construct a split hyperplane equidistant from both points. This hyperplane divides the data points into two subspaces based on their position relative to it (figure 2a). Next, for each of these two subspaces the same procedure can be repeated: by randomly drawing a new split hyperplane into the data subspace it can be further partitioned into two smaller subspaces. This process is recursively repeated to construct a binary index tree (figures 2b and 2c).

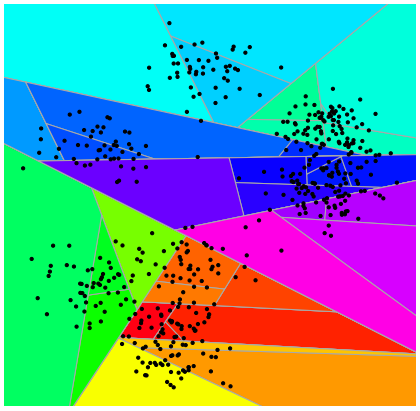
This binary index tree can be used to efficiently find the nearest neighbor for a given query point. Instead of having to compare the query point to all data points the index tree can be traversed to find the data subspace to which the query point would belong and which will likely contain its nearest neighbor (figures 2d and 2e). In this fashion a nearest neighbor query can be performed in logarithmic time in terms of the number of data points, whereas it would require linear time in terms of the number of data points to compare the query point against all data points in a brute-force fashion.

Unfortunately, the nearest neighbor for the query point might not be located in the data subspace that has been selected, but instead it might be present in an adjacent data subspace. In this case it is not possible to directly identify the actual nearest neighbor and only an approximate result will be achieved. To reduce the risk of missing the actual nearest neighbor multiple complementary index trees are used. Because random split hyperplanes are used to divide the data space while constructing the initial tree, an alternative index tree can be created by making use of different split hyperplanes. The new random split hyperplanes cause the data space to be subdivided differently, leading to a different binary index tree. As a result, the risk of missing an actual nearest neighbor is minimized by using both index trees simultaneously to answer a query. During querying the data subspaces to which a query point belongs are identified for each tree individually, after which the data points in both of these subspaces are combined to find the actual nearest neighbor. Finally, additional index trees can be constructed in a similar fashion to form an ensemble of index trees, with each tree providing a complementary view on the data. Through the combination of this ensemble of index trees the risk of missing the actual nearest neighbor is further

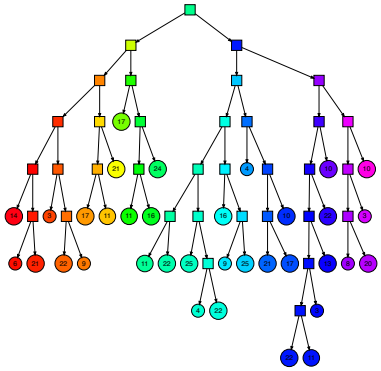
bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.



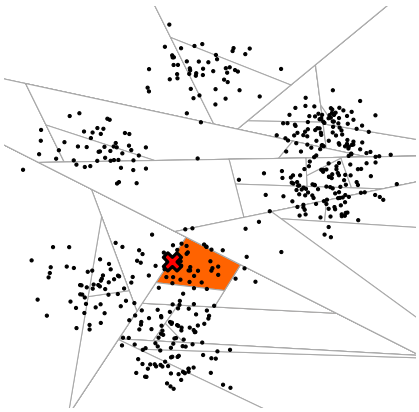
(a) The data space is partitioned into two subspaces using a random split hyperplane.



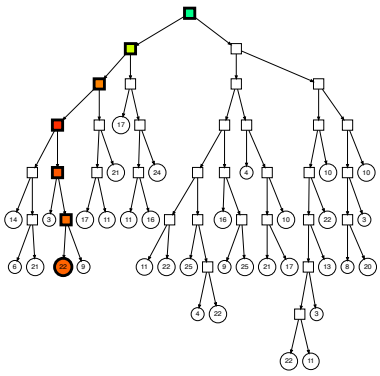
(b) Each subspace is recursively partitioned further.



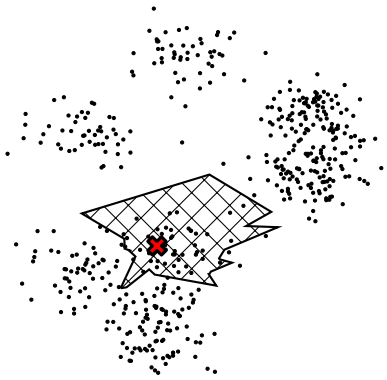
(c) Subspace partitioning results in a binary index tree being constructed.



(d) The data subspace to which a query item belongs is retrieved to find its nearest neighbors.



(e) This subspace is efficiently found using the binary index tree.



(f) A composite data subspace is compiled from multiple index trees to obtain a better approximation.

Figure 2: Approximate nearest neighbor indexing and searching using an ensemble of random split hyperplane index trees.

decreased (figure 2f). As such the number of trees in the ensemble is a hyperparameter that can be used to configure a trade-off between accuracy and speed, as using more trees reduces the risk of missing the actual nearest neighbor at the expense of some increased computational requirements.

2.1.3 Candidate ranking

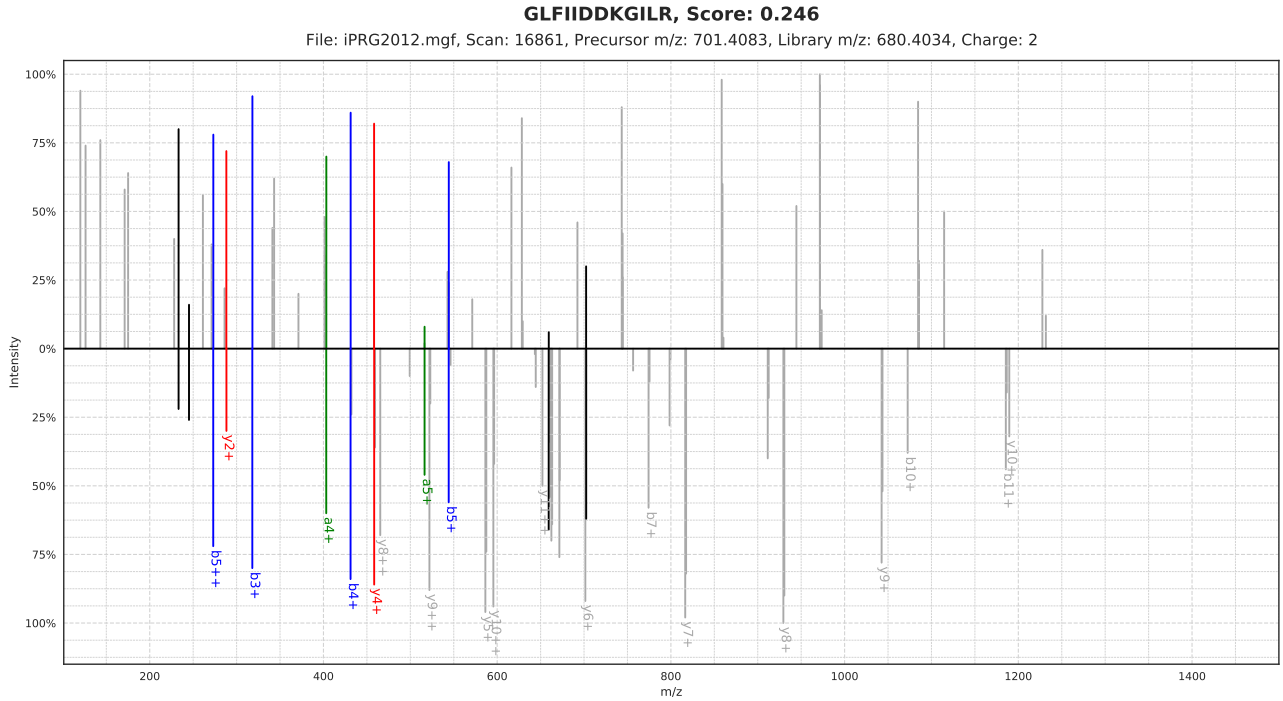
During the candidate ranking step the similarities between the query spectrum and all library spectra that have been selected in the previous step are evaluated to determine the highest-scoring SSM. Even though the ANN index already retrieves the most likely candidate spectra based on their similarity, a subsequent ranking step remains necessary. This is because the vectors employed for ANN indexing only represent the spectra at a coarse, 1 Da bin granularity. In contrast, for high-resolution mass spectra a more accurate score can be computed using a low fragment mass tolerance to obtain the optimal match. Additionally, during the open search the shifted dot product is used as scoring method to take PTMs into account and accurately match modified spectra to their unmodified variant, as described next.

The dot product is a well-established scoring method to rank SSMs. An important advantage of the dot product is that, despite its simplicity, it is able to accurately capture the similarity between two mass spectra [55]. Additionally, it can be computed very efficiently as it has a time complexity of $O(n)$, where n is the total number of peaks in the two spectra being compared. Based on these advantageous properties the dot product has been used by several spectral library search engines [26, 41]. Similarly, ANN-SoLo uses the dot product to identify unmodified spectra during the first level of its cascade search.

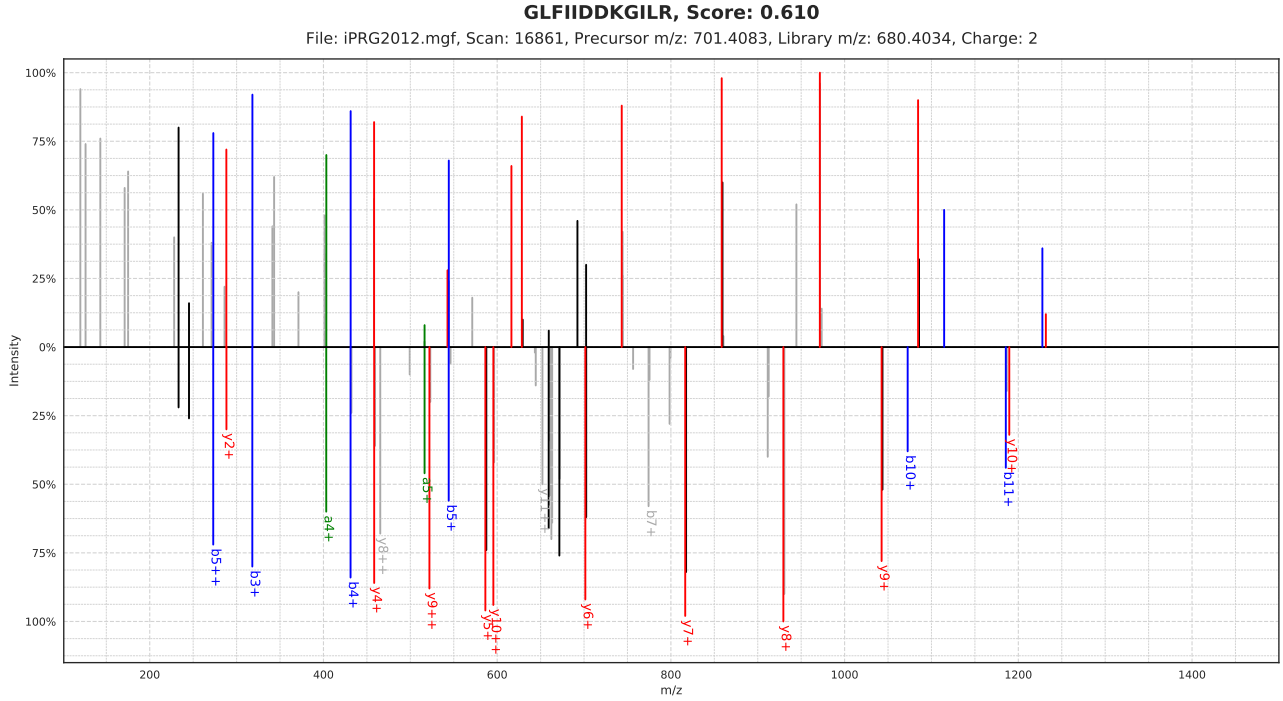
However, because the dot product only considers directly matching peaks with identical masses (while taking the fragment mass tolerance into account) it is less suitable to identify modified spectra. Instead, during the second level of its cascade search ANN-SoLo uses a variation on the dot product, called the shifted dot product [11, 33]. This shifted dot product also considers peaks that are shifted according to the precursor mass difference between the two spectra that are being matched to accurately identify modified spectra (figure 3). We here briefly describe an algorithm to compute the shifted dot product in $O(n \log n)$ time complexity [33].

First, the precursor mass difference between the two spectra is calculated and normalized according to the precursor charge. Next, all potential peak pairs, with

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.



(a) The standard dot product only matches the prefix fragments that do not include the acetylation.



(b) The shifted dot product correctly matches both unmodified and modified fragments.

Figure 3: The shifted dot product enables more accurate matching between an unmodified library spectrum (bottom) and a modified query spectrum (top) than the standard dot product. As can be derived from the precursor mass difference the peptide GLFIIDDKGILR has undergone acetylation (mass 42.010 565 Da) on the lysine at position 8. The standard dot product only takes directly matching peaks into account, while the shifted dot product can consider shifted peaks according to the precursor mass difference and charge, correctly assigning a high score to this match.

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

and without a mass shift (taking into account different charges), can be determined in a linear pass through both spectra. Each peak pair is scored by multiplying the intensities of both peaks, as in the standard dot product. Unshifted peak matches and shifted peak matches that include an annotated peak are scored fully while shifted peak matches without an annotation are slightly penalized to minimize the influence of potential spurious matches. Next, to calculate the final shifted dot product score the peak matches are sorted on their intensity product, after which they are summed in a greedy fashion while avoiding to match a single peak in either of the two spectra more than once.

2.2 FDR calculation

False discovery rates are calculated in two phases after both levels of the cascade search using the target–decoy strategy [24]. Specifically, FDRs are estimated based on a concatenated spectral library containing both target and decoy spectra (see section 2.3) using the number of decoys divided by the number of target SSMs.

During the first level of the cascade search the SSMs are directly filtered on the FDR. In contrast, because the score distributions of peptides with different modifications can exhibit distinct properties, during the second level of the cascade search the SSMs are filtered using the subgroup FDR strategy [27]. To combine SSMs with identical modifications they are grouped based on their charge-normalized precursor mass difference between the query spectra and library spectra. The SSMs are split into subgroups by iteratively selecting the SSM with the highest match score alongside all other remaining SSMs whose precursor mass difference falls within a 0.1 Da range. Because FDR estimates become progressively less reliable if only a limited number of observations are considered FDR filtering is only done for subgroups that contain at least five SSMs. Subgroups that contain fewer SSMs are combined into a residual group instead whose FDR is jointly calculated in the end.

2.3 Data sets

The first data set we used was generated in the context of the 2012 study by the Proteome Informatics Research Group of the Association of Biomolecular Resource Facilities. The goal of this study was to assess the community's ability to analyze modified peptides [12]. Towards this end, various participating researchers were asked to identify an unknown data set, after which their proficiency in handling modified peptides was evaluated. The provided data set consisted of a mixture of synthetic peptides with biologically occurring modifications combined with a yeast whole cell lysate as background, and the spectra were measured using a TripleTOF instrument. For full details on the sample preparation and acquisition see the original publication by Chalkley et al. [12]. This high quality data set has been recommended as a reference data set for the evaluation of identification algorithms [28]. All data was downloaded from the MassIVE data repository (accession MSV000078492).

To search the iPRG2012 data set the human ion trap (version 2012/05/30) and yeast ion trap (version 2012/04/06) spectral libraries compiled by NIST and obtained from the PeptideAtlas [18] website were used. First, both spectral libraries were concatenated using SpectraST [41], after which additional decoy spectra were added in a 1:1 ratio using the shuffle-and-reposition method [40]. This resulted in a single large spectral library file containing 799 574 spectra. The fragment masses in the NIST ion trap spectral libraries are only specified to the nearest tenth of a Dalton. Due to the limited mass accuracy and resolution of the ion trap instruments from which these spectral libraries are derived, care has to be taken to specify a suitable fragment mass tolerance when using this spectral library.

The second data set consists of spectra measured from the HEK293 human cell line [15]. As per Chick et al. [15], the HEK293 cells were first lysed, trypsinized, and separated into 24 fractions, after which high-resolution and high-mass accuracy MS/MS spectra were obtained on an LTQ Orbitrap Elite mass spectrometer. For full details on the sample preparation and acquisition see the original publication by Chick et al. [15]. Raw files were downloaded from PRIDE [58] (project PXD001468) and converted to MGF files using msconvert [13].

To search the HEK293 data set the MassIVE-KB peptide spectral library (version 2017/11/27) was used. This is a repository-wide human higher-energy collisional dissociation spectral library derived from over 30 TB of human MS/MS proteomics data. The original spectral library contained 2 148 752 MS/MS spectra, from which duplicates were removed using SpectraST [41] by retaining only the best replicate spectrum for each individual peptide ion, resulting in a spectral library containing 1 504 951 spectra. Next, decoy spectra were added in a 1:1 ratio using the shuffle-and-reposition method [40], resulting in a final spectral library containing 3 009 902 spectra. To the best of our knowledge this is the largest spectral library reported in the literature to date.

All MS/MS data, spectral libraries, and identification results have been deposited to the ProteomeXchange Consortium [20] via the PRIDE partner repository [58].

2.4 Search settings

2.4.1 ANN-SoLo

The spectrum preprocessing settings are as described in section 2.1.1. For the iPRG2012 data set a precursor mass window of 20 ppm was used for the standard search and 20 ppm followed by 300 Da for the cascade open searches. Additionally, a fragment mass tolerance of 0.25 Da was used in all cases. For the HEK293 data set a precursor mass window of 5 ppm was used for the standard searches and a precursor mass window of 5 ppm followed by 500 Da was used for the cascade open searches. For all HEK293 searches a fragment mass tolerance of 0.02 Da was used.

Two major hyperparameters influence the performance of the ANN index: the number of trees in the index and the number of nodes to inspect per query. We

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

discuss the effect of these hyperparameters in detail in section 3.2. For the iPRG2012 data set we used ANN indices consisting of 100 to 1000 trees, while the number of nodes inspected during querying was varied between 20 000 and 400 000. To analyze the HEK293 data set an ANN index consisting of 1000 trees was used and 200 000 nodes were inspected during searching.

2.4.2 SpectraST

We mainly compared the performance of ANN-SoLo against the state-of-the-art spectral library search engine SpectraST [41].

We used SpectraST version 5.0 as part of the Trans-Proteomic Pipeline version 5.1.0 [19]. We tried to specify the SpectraST search settings for processing the HEK293 data set as closely as possible to the ANN-SoLo settings to ensure a fair comparison. Spectra were preprocessed to have a minimum mass range of 250 Da and only the 50 most intense peaks were retained. Peak-to-peak matching and rank-based scoring was used to evaluate SSMs. Library caching was enabled, which is an essential requirement in order to be able to complete the open searches. Similar to the ANN-SoLo settings a precursor mass window of 500 Da was used for the open searches. In contrast, for the standard searches a precursor mass window of 0.02 Da was used as SpectraST does not support specification of tolerances in ppm units. A fragment mass tolerance of 0.02 Da was used in all cases. FDRs were estimated using the subgroup FDR strategy described in section 2.2.

2.5 Code availability

The ANN-SoLo software is written in Python, making use of various open-source libraries such as NumPy [57], SciPy, and pandas [44] for scientific computing and Matplotlib [34], Seaborn [61], and Jupyter notebooks [49] for visualization purposes. Pyteomics [29] is used to support some mass spectrometry-specific functionality, such as reading input files and for FDR calculation.

ANN indexing during the candidate selection step is based on the open-source Approximate Nearest Neighbors Oh Yeah (Annoy) library [54], which was originally developed at Spotify to support large-scale music recommendations. Dot product and shifted dot product calculation is implemented as an external C++ module to optimize the candidate ranking step for speed.

All code is released as open source under the permissive Apache 2.0 license and is available at `https://github.com/bittremieux/ANN-SoLo`.

3 Results

3.1 Cascade open search maximally identifies unmodified and modified peptides

The advantage of an open search compared to a standard search is that modified peptides can be identified without having to specify the expected modifications. However, using a wide precursor mass window might lead to a loss in identifications of spectra that would have previously been identified during a standard search. Whereas a specific query spectrum might be identified differently with a slightly higher score when using a wide precursor mass window than when using a small precursor mass window, as the search space is significantly larger in the former case these scores cannot be directly compared. Instead, through its cascade search strategy ANN-SoLo maximally identifies both unmodified and modified peptides. Based on the iPRG2012 data set ANN-SoLo achieves a 38 % increase in identifications when performing an open search compared to a standard search (table 1), with the additional identifications corresponding to modified peptides.

Search mode	Similarity measure	# SSMs
Standard search	dot product	3610
Open search	dot product	4970
Open search	shifted dot product	6051

Table 1: The number of accepted SSMs at a 1 % FDR threshold for various searches of the iPRG2012 data set. The open search identifies all unmodified spectra previously identified in the standard search as well as additional modified spectra. A further increase in identifications is achieved by using the shifted dot product to score modified SSMs.

Moreover, a further 30 % increase in identifications is achieved by using the shifted dot product instead of the standard dot product to score the spectral similarity of modified spectra (table 1). Because the shifted dot product explicitly accounts for modification-induced shifts in the fragment ion peaks it more accurately identifies modified spectra. As a result, by employing a cascade open search strategy and using an optimized scoring function modified spectra can be accurately matched to the spectral library, resulting in a total increase in identifications of 68 % when comparing a standard search against an open search on the iPRG2012 data set.

3.2 Approximate nearest neighbor indexing speeds up open search

By using an ANN index only a limited number of the most relevant library spectra are retrieved during the candidate selection step, which speeds up the subsequent candidate ranking step because far fewer SSM scores have to be computed. Two major hyperparameters influence the performance of the ANN index: the number of trees in the index ensemble, which controls the index construction during an initialization phase, and the number of tree nodes to inspect when evaluating a query, which controls the performance during searching.

Using more trees has a positive effect on the accuracy of the ANN index by providing multiple complementary views on the data subspaces, as detailed previously. However, this improvement comes at the expense of some increased computational requirements because more index trees need to be traversed. Additionally, the number

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

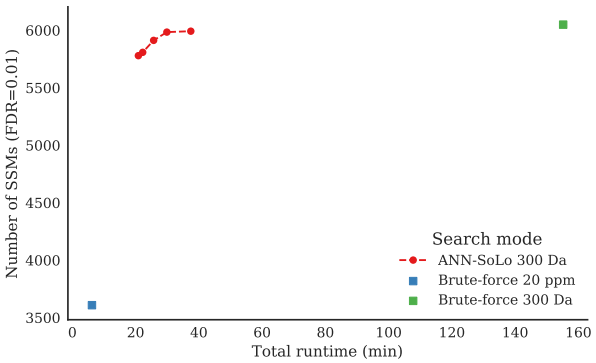


Figure 4: Runtime versus number of identifications for various searches of the iPRG2012 data set. Timing results in this figure and in table 2 were obtained on a single-core Intel Xeon E5-2680 v2 processor. An open search identifies a significantly higher number of spectra than a standard search at the expense of a large increase in runtime. The ANN-SoLo results show that the ANN index significantly reduces the time required for open searches. The multiple ANN-SoLo results correspond to different configurations of the ANN index hyperparameters, with the settings that lie on the Pareto frontier shown. Even when maximizing accuracy to achieve the same number of identifications as the brute-force approach ANN-SoLo considerably speeds up the search.

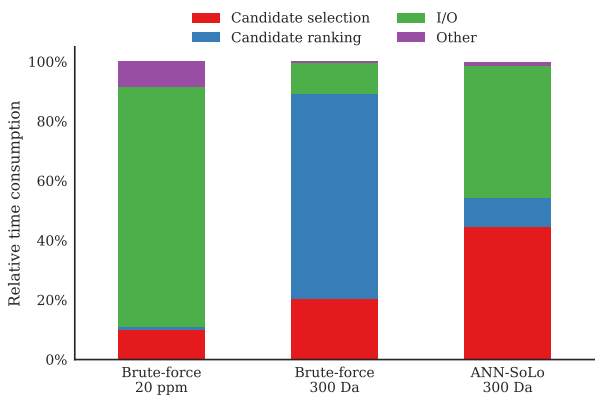


Figure 5: Profiling shows how much time was spent in each part of the code for various searches of the iPRG2012 data set. During a brute-force open search the majority of time is spent during the candidate ranking step, while ANN indexing helps to select only a limited number of candidates and minimize the time required to rank these candidates. Note that relative time consumptions are reported: using an ANN index results in a significant speedup, as shown previously. Correspondingly, for example, although the relative I/O time consumption is higher when using an ANN index than in the brute-force case, the absolute I/O time consumption is lower.

of tree nodes to inspect when evaluating a query can similarly be used to configure the accuracy of the ANN index. Instead of only inspecting the subspace that exactly contains the query point the neighboring subspaces that are closest to the query point across all index trees can be inspected as well. This approach helps to avoid missing the nearest neighbors for the query point. Because the number of nodes to inspect is specified across all trees in the ANN index, this hyperparameter is to some extent related to the number of trees used.

These two hyperparameters constitute a trade-off between speed and accuracy, both when constructing the ANN index and at runtime. In general, using more trees and inspecting more nodes will lead to a more faithful approximation of the set of nearest neighbors retrieved from the ANN index, at the expense of an increase in computational requirements. Based on the iPRG2012 data set there is a clear difference in runtime between a standard search and the traditional brute-force approach of performing an open search on the one hand, and the advantage of using an ANN index for the open search on the other hand (figure 4). As shown previously, the open search allows us to identify a significantly higher number of spectra, with the newly identified spectra corresponding to modified peptides. Unfortunately, this comes at the expense of a large increase in runtime, rendering OMS infeasible in practice. In contrast, by making use of an ANN index ANN-SoLo significantly decreases the time required to perform an open search, making OMS a viable strategy.

The ANN-SoLo speedup results from a reduction in the number of candidates that have to be evaluated during the candidate ranking step (figure 5). A standard search only takes a very short amount of time, with I/O costs for reading the experimental and library spectra forming the major bottleneck. In contrast, the massive increase in runtime of a brute-force open search is caused by the fact that when a very wide precursor mass window is used each query spectrum must be compared against a very large number of library spectra. Using an ANN index instead puts the focus on the candidate selection step to only retrieve the most relevant candidates. As a result, the relative proportion of work during the candidate ranking step significantly decreases. The ANN index also makes it possible to use complex scoring functions without incurring an overly excessive slowdown. Although the shifted dot product is computationally more expensive than the standard dot product, negative effects on the total runtime are limited due to the optimized candidate selection step.

3.3 ANN-SoLo outperforms SpectraST in terms of speed and number of identifications

We compared ANN-SoLo to SpectraST, a commonly used spectral library search engine, in terms of speed and the number of identifications on the HEK293 data set (table 2). For a standard search SpectraST is clearly faster than ANN-SoLo. This can be attributed to simple implementation differences, of which a notable factor is that ANN-SoLo is mainly implemented in the Python programming language while SpectraST is implemented in C++. As Python is an interpreted programming language it can be up to a hundred times slower than a compiled

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

Search engine	Time (min)	# SSMs	# Peptides
Standard search			
SpectraST	5	369 242	109 518
ANN-SoLo	24	352 938	113 108
Open search			
SpectraST	1277	531 886	137 064
ANN-SoLo	107	745 327	195 999

Table 2: Runtime and identification rates for ANN-SoLo and SpectraST on the HEK293 data set. The runtime is reported in minutes and represents the average runtime over all 24 raw files. The identification rate is reported in terms of the number of accepted SSMs at 1 % FDR and the number of corresponding unique peptidoforms and is reported for the entire 24-run data set.

# SSMs	Δm (Da)	Potential modification
369 341	0.003	
46 659	1.005	First isotopic peak
31 473	15.998	Oxidation or hydroxylation / Ala → Ser substitution / Phe → Tyr substitution
14 088	2.006	Second isotopic peak
6 418	−0.991	Amidation
5 777	−17.023	Pyro-glu from Q / loss of ammonia
5 433	17.002	Replacement of proton with ammonium ion
4 815	183.039	Aminoethylbenzenesulfonylation
4 777	27.998	Formylation / Ser → Asp substitution / Thr → Glu substitution
3 266	301.991	Unidentified modification [15, 39]

Table 3: The most frequent precursor mass differences for the HEK293 data set and likely modifications sourced from Unimod corresponding to these precursor mass differences. The delta-mass column contains the median precursor mass difference of that SSM subgroup.

programming language such as C++, as benchmarks have shown for several general tasks. In contrast, for the open search ANN-SoLo is an order of magnitude faster than SpectraST, reducing the search time from almost a day on average to under two hours, despite the inherent programming language disadvantage. This result clearly shows the massive advantage of ANN indexing to speed up OMS.

Furthermore, by using a cascade search strategy to maximize the number of identified unmodified and modified spectra and the shifted dot product to sensitively match modified spectra to their unmodified counterpart ANN-SoLo successfully achieves a higher identification rate than SpectraST, surpassing even previously reported results [15, 39].

Next, we have investigated which modifications frequently occur in this HEK293 human cell line data set based on the precursor mass differences for the identified SSMs. By referencing the observed precursor mass differences against the Unimod public database of protein modifications [16] we can derive the chemical events that likely explain the observed mass differences (table 3). We can see that common modifications, such as oxidation, frequently occur. Additionally, mass shifts corresponding to various amino acid substitutions can be frequently observed. These can potentially indicate single amino acid variants, but can likely also be explained by the incomplete coverage of the human proteome by our spectral library. Consequently, OMS can to some extent alleviate a longstanding criticism of spectral library searching in that it is only able to re-identify spectra that have been previously observed. Finally, the precise mapping of these mass differences to a known modification further substantiates the validity of these identifications and confirms that OMS can be used to accurately identify modified peptides.

4 Conclusions

Here we have introduced the ANN-SoLo spectral library search engine. ANN-SoLo uses ANN indexing to efficiently select the most likely candidate matches from a spectral library based on their spectral similarity with the query spectrum. As candidate retrieval using the ANN index only depends on spectral similarity without taking precursor mass information into account this strategy naturally lends itself to OMS. By using the ANN index the number of candidates that need to be evaluated for each query spectrum can be reduced by orders of magnitude, decreasing the time required to perform an open search accordingly. Furthermore, because the number of potential matches that needs to be evaluated is small this opens up the possibility to use more computationally expensive similarity measures to score SSMs without incurring an overly large performance hit. This is exemplified by our use of the shifted dot product, which allows us to accurately match a modified spectrum to its unmodified counterpart.

Thanks to these advances open spectral library searching has become a feasible strategy for the sensitive identification of modified peptides. We have demonstrated how an extremely large spectral library can be used to detect peptide modifications at a large scale, which can give important insights into their biological activity. Notably, we have used a repository-wide spectral library which has been derived from a massive amount of publicly available spectral data. Using a spectral library of such size for open searching with a traditional search engine would drastically suffer from excessive runtimes. In contrast, ANN-SoLo makes it possible to perform such searches in a reasonable time frame.

A small disadvantage of using an ANN index, however, is that this index needs to be constructed prior to MS/MS spectrum identification. Especially when many index trees are used in combination with a large spectral library building the ANN index may take a non-negligible

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

amount of time. However, this step only needs to happen once, after which the ANN index can be reused for subsequent searches. Additionally, the space requirement of the ANN index is proportional to the size of the spectral library for which it is constructed. The ANN index will typically be larger than other commonly used indexing methods to assist MS/MS spectrum identification, such as a peptide index [22, 47] or a fragment ion index [39], because the ANN index contains a low-resolution vector representation for each spectrum in the spectral library. Nonetheless, for current spectral libraries the memory requirements remain feasible.

The application of ANN indexing need not be restricted to open spectral library searching. Other identification tasks which exhibit a large search space, such as metaproteomics [45, 48], or activities which consist of large-scale spectral processing tasks, such as spectral clustering [25, 30, 31, 56], can similarly benefit from ANN indexing to achieve substantial speedups.

Furthermore, although here we have used an ANN index consisting of an ensemble of random projection trees several alternative methods for ANN indexing exist. Some examples of ANN indexing techniques which have exhibited excellent empirical performance [5] include hierarchical navigable small world graphs [43], LSH [4], product quantization [36], etc. Additionally, non-metric space indexing can potentially be used to retrieve candidates based on the shifted dot product similarity directly [10]. An investigation into whether these techniques are suitable to speed up spectral library searching remains as future work. Another promising approach to achieve further speedups is by making use of specialized hardware such as GPUs, both for candidate selection using an ANN index [35] as well as to evaluate SSMs [6].

Finally, the FDR procedure plays an important role in evaluating the identification results of an open search. As using a very wide precursor mass window leads to a large increase in search space the probability of having a high-scoring spurious match is considerably higher for an open search compared to a standard search, and such high-scoring decoy matches can have a large influence on the number of accepted identifications when using a global FDR strategy. In contrast, the subgroup FDR procedure calculates the FDR separately for spectra that have distinct modifications [27]. In practice we have observed that subgroups that unambiguously correspond to known modifications often contain very few decoy matches. In contrast, many decoy matches do not belong to a common subgroup as their precursor mass difference is randomly distributed across the range of the precursor mass window. Instead these decoy SSMs are combined in the residual group for FDR calculation, minimizing their negative influence on the accepted identifications. Caution has to be observed though as the actual FDR might be underestimated when too small groups are used [27]. It is a standing research question whether alternative approaches are needed for the accurate FDR estimation of open searches [21, 39].

The ANN-SoLo spectral library searching engine is freely available as open source. The source code and detailed instructions can be found at `https://github.com/bittremieux/ANN-SoLo`.

Acknowledgement

W.B. is a postdoctoral researcher of the Research Foundation – Flanders (FWO). W.S.N. was supported by National Institutes of Health award R01 GM121818.

The computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation – Flanders (FWO) and the Flemish Government – department EWI.

References

[1] Aebersold, R., Agar, J. N., Amster, I. J., Baker, M. S., et al. "How Many Human Proteoforms Are There?" In: *Nature Chemical Biology* 14.3 (Feb. 14, 2018), pp. 206–214. DOI: 10.1038/nchembio.2576.

[2] Ahrné, E., Müller, M., and Lisacek, F. "Unrestricted Identification of Modified Proteins Using MS/MS." In: *PROTEOMICS* 10.4 (Feb. 2010), pp. 671–686. DOI: 10.1002/pmic.200900502.

[3] Ahrné, E., Nikitin, F., Lisacek, F., and Müller, M. "QuickMod: A Tool for Open Modification Spectrum Library Searches." In: *Journal of Proteome Research* 10.7 (July 1, 2011), pp. 2913–2921. DOI: 10.1021/pr200152g.

[4] Andoni, A., Indyk, P., Laarhoven, T., Razenshteyn, I., et al. "Practical and Optimal LSH for Angular Distance." In: *Proceedings of the 28th International Conference on Neural Information Processing Systems - NIPS '15*. Montreal, Canada: MIT Press, Dec. 7, 2015, pp. 1225–1233.

[5] Aumüller, M., Bernhardsson, E., and Faithfull, A. "ANN-Benchmarks: A Benchmarking Tool for Approximate Nearest Neighbor Algorithms." In: *Proceedings of the 10th International Conference on Similarity Search and Applications - SISAP '17*. Ed. by Beecks, C., Borutta, F., Kröger, P., and Seidl, T. Vol. 10609. Munich, Germany: Springer International Publishing, Oct. 4, 2017, pp. 34–49.

[6] Baumgardner, L. A., Shanmugam, A. K., Lam, H., Eng, J. K., et al. "Fast Parallel Tandem Mass Spectral Library Searching Using GPU Hardware Acceleration." In: *Journal of Proteome Research* 10.6 (June 3, 2011), pp. 2882–2888. DOI: 10.1021/pr200074h.

[7] Bittremieux, W., Tabb, D. L., Impens, F., Staes, A., et al. "Quality Control in Mass Spectrometry-Based Proteomics." In: *Mass Spectrometry Reviews* Early view (Aug. 12, 2017). DOI: 10.1002/mas.21544.

[8] Bogdanow, B., Zauber, H., and Selbach, M. "Systematic Errors in Peptide and Protein Identification and Quantification by Modified Peptides." In: *Molecular & Cellular Proteomics* 15.8 (Aug. 1, 2016), pp. 2791–2801. DOI: 10.1074/mcp.M115.055103.

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

[9] Borgelt, C. "Frequent Item Set Mining." In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2.6 (Nov. 2012), pp. 437–456. DOI: 10.1002/widm.1074.

[10] Boytsov, L. and Naidan, B. "Engineering Efficient and Effective Non-Metric Space Library." In: *Proceedings of the 6th International Conference on Similarity Search and Applications - SISAP '13*. Ed. by Brisaboa, N. R., Pedreira, O., and Zezula, P. Vol. 8199. A Coruña, Spain: Springer Berlin Heidelberg, Oct. 2, 2013, pp. 280–293. DOI: 10.1007/978-3-642-41062-8_28.

[11] Burke, M. C., Mirokhin, Y. A., Tchekhovskoi, D. V., Markey, S. P., et al. "The Hybrid Search: A Mass Spectral Library Search Method for Discovery of Modifications in Proteomics." In: *Journal of Proteome Research* (Apr. 11, 2017). DOI: 10.1021/acs.jproteome.6b00988.

[12] Chalkley, R. J., Bandeira, N., Chambers, M. C., Clauser, K. R., et al. "Proteome Informatics Research Group (iPRG)_2012: A Study on Detecting Modified Peptides in a Complex Mixture." In: *Molecular & Cellular Proteomics* 13.1 (Jan. 1, 2014), pp. 360–371. DOI: 10.1074/mcp.M113.032813.

[13] Chambers, M. C., Maclean, B., Burke, R., Amodei, D., et al. "A Cross-Platform Toolkit for Mass Spectrometry and Proteomics." In: *Nature Biotechnology* 30.10 (Oct. 10, 2012), pp. 918–920. DOI: 10.1038/nbt.2377.

[14] Chi, H., Liu, C., Yang, H., Zeng, W.-F., et al. "OpenpFind Enables Precise, Comprehensive and Rapid Peptide Identification in Shotgun Proteomics." In: *bioRxiv* (Mar. 20, 2018). DOI: 10.1101/285395.

[15] Chick, J. M., Kolippakkam, D., Nusinow, D. P., Zhai, B., et al. "A Mass-Tolerant Database Search Identifies a Large Proportion of Unassigned Spectra in Shotgun Proteomics as Modified Peptides." In: *Nature Biotechnology* 33.7 (June 15, 2015), pp. 743–749. DOI: 10.1038/nbt.3267.

[16] Creasy, D. M. and Cottrell, J. S. "Unimod: Protein Modifications for Mass Spectrometry." In: *PROTEOMICS* 4.6 (Apr. 5, 2004), pp. 1534–1536. DOI: 10.1002/pmic.200300744.

[17] David, M., Fertin, G., Rogniaux, H., and Tessier, D. "SpecOMS: A Full Open Modification Search Method Performing All-to-All Spectra Comparisons within Minutes." In: *Journal of Proteome Research* 16.8 (Aug. 4, 2017), pp. 3030–3038. DOI: 10.1021/acs.jproteome.7b00308.

[18] Desiere, F., Deutsch, E. W., King, N. L., Nesvizhskii, A. I., et al. "The PeptideAtlas Project." In: *Nucleic Acids Research* 34.90001 (Jan. 1, 2006), pp. D655–D658. DOI: 10.1093/nar/gkj040.

[19] Deutsch, E. W., Mendoza, L., Shteynberg, D., Farrah, T., et al. "A Guided Tour of the Trans-Proteomic Pipeline." In: *PROTEOMICS* 10.6 (Mar. 6, 2010), pp. 1150–1159. DOI: 10.1002/pmic.200900375.

[20] Deutsch, E. W., Csordas, A., Sun, Z., Jarnuczak, A., et al. "The ProteomeXchange Consortium in 2017: Supporting the Cultural Change in Proteomics Public Data Deposition." In: *Nucleic Acids Research* 45 (D1 Jan. 4, 2017), pp. D1100–D1106. DOI: 10.1093/nar/gkw936.

[21] Devabhaktuni, A., Olsson, N., Gonzalez, C. G., Rawson, K., et al. "Measuring Proteomes with Long Strings: A New, Unconstrained Paradigm in Mass Spectrum Interpretation." In: *bioRxiv* (Mar. 15, 2018). DOI: 10.1101/282624.

[22] Diament, B. J. and Noble, W. S. "Faster SEQUEST Searching for Peptide Identification from Tandem Mass Spectra." In: *Journal of Proteome Research* 10.9 (Sept. 2, 2011), pp. 3871–3879. DOI: 10.1021/pr101196n.

[23] Dutta, D. and Chen, T. "Speeding up Tandem Mass Spectrometry Database Search: Metric Embeddings and Fast near Neighbor Search." In: *Bioinformatics* 23.5 (Mar. 5, 2007), pp. 612–618. DOI: 10.1093/bioinformatics/btl645.

[24] Elias, J. E. and Gygi, S. P. "Target-Decoy Search Strategy for Increased Confidence in Large-Scale Protein Identifications by Mass Spectrometry." In: *Nature Methods* 4.3 (Feb. 27, 2007), pp. 207–214. DOI: 10.1038/nmeth1019.

[25] Frank, A. M., Bandeira, N., Shen, Z., Tanner, S., et al. "Clustering Millions of Tandem Mass Spectra." In: *Journal of Proteome Research* 7.1 (Jan. 4, 2008), pp. 113–122. DOI: 10.1021/pr070361e.

[26] Frewen, B. E., Merrihew, G. E., Wu, C. C., Noble, W. S., et al. "Analysis of Peptide MS/MS Spectra from Large-Scale Proteomics Experiments Using Spectrum Libraries." In: *Analytical Chemistry* 78.16 (Aug. 15, 2006), pp. 5678–5684. DOI: 10.1021/ac060279n.

[27] Fu, Y. and Qian, X. "Transferred Subgroup False Discovery Rate for Rare Post-Translational Modifications Detected by Mass Spectrometry." In: *Molecular & Cellular Proteomics* 13.5 (May 1, 2014), pp. 1359–1368. DOI: 10.1074/mcp.O113.030189.

[28] Gatto, L., Hansen, K. D., Hoopmann, M. R., Hermjakob, H., et al. "Testing and Validation of Computational Methods for Mass Spectrometry." In: *Journal of Proteome Research* 15.3 (Mar. 4, 2016), pp. 809–814. DOI: 10.1021/acs.jproteome.5b00852.

[29] Goloborodko, A. A., Levitsky, L. I., Ivanov, M. V., and Gorshkov, M. V. "Pyteomics-a Python Framework for Exploratory Data Analysis and Rapid Software Prototyping in Proteomics." In: *Journal of The American Society for Mass Spectrometry* 24.2 (Feb. 1, 2013), pp. 301–304. DOI: 10.1007/s13361-012-0516-6.

[30] Griss, J. "Spectral Library Searching in Proteomics." In: *PROTEOMICS* 16.5 (Mar. 2016), pp. 729–740. DOI: 10.1002/pmic.201500296.

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

[31] Griss, J., Foster, J. M., Hermjakob, H., and Vizcaíno, J. A. "PRIDE Cluster: Building a Consensus of Proteomics Data." In: *Nature Methods* 10.2 (Jan. 30, 2013), pp. 95–96. DOI: 10.1038/nmeth.2343.

[32] Griss, J., Perez-Riverol, Y., Lewis, S., Tabb, D. L., et al. "Recognizing Millions of Consistently Unidentified Spectra across Hundreds of Shotgun Proteomics Datasets." In: *Nature Methods* 13.8 (June 27, 2016), pp. 651–656. DOI: 10.1038/nmeth.3902.

[33] Horlacher, O., Lisacek, F., and Müller, M. "Mining Large Scale Tandem Mass Spectrometry Data for Protein Modifications Using Spectral Libraries." In: *Journal of Proteome Research* 15.3 (Mar. 4, 2016), pp. 721–731. DOI: 10.1021/acs.jproteome.5b00877.

[34] Hunter, J. D. "Matplotlib: A 2D Graphics Environment." In: *Computing in Science & Engineering* 9.3 (June 18, 2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[35] Johnson, J., Douze, M., and Jégou, H. "Billion-Scale Similarity Search with GPUs." In: *arXiv:1702.08734 [cs]* (Feb. 28, 2017). arXiv: 1702.08734.

[36] Jégou, H., Douze, M., and Schmid, C. "Product Quantization for Nearest Neighbor Search." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.1 (Jan. 2011), pp. 117–128. DOI: 10.1109/TPAMI.2010.57.

[37] Kertesz-Farkas, A., Keich, U., and Noble, W. S. "Tandem Mass Spectrum Identification via Cascaded Search." In: *Journal of Proteome Research* 14.8 (Aug. 7, 2015), pp. 3027–3038. DOI: 10.1021/pr501173s.

[38] Kim, M.-S., Zhong, J., and Pandey, A. "Common Errors in Mass Spectrometry-Based Analysis of Post-Translational Modifications." In: *PROTEOMICS* 16.5 (Mar. 2016), pp. 700–714. DOI: 10.1002/pmic.201500355.

[39] Kong, A. T., Leprevost, F. V., Avtonomov, D. M., Mellacheruvu, D., et al. "MSFragger: Ultrafast and Comprehensive Peptide Identification in Mass Spectrometry–based Proteomics." In: *Nature Methods* (Apr. 10, 2017). DOI: 10.1038/nmeth.4256.

[40] Lam, H., Deutsch, E. W., and Aebersold, R. "Artificial Decoy Spectral Libraries for False Discovery Rate Estimation in Spectral Library Searching in Proteomics." In: *Journal of Proteome Research* 9.1 (Jan. 4, 2010), pp. 605–610. DOI: 10.1021/pr900947u.

[41] Lam, H., Deutsch, E. W., Eddes, J. S., Eng, J. K., et al. "Development and Validation of a Spectral Library Searching Method for Peptide Identification from MS/MS." In: *PROTEOMICS* 7.5 (Mar. 5, 2007), pp. 655–667. DOI: 10.1002/pmic.200600625.

[42] Ma, C. W. M. and Lam, H. "Hunting for Unexpected Post-Translational Modifications by Spectral Library Searching with Tier-Wise Scoring." In: *Journal of Proteome Research* 13.5 (May 2, 2014), pp. 2262–2271. DOI: 10.1021/pr401006g.

[43] Malkov, Y. A. and Yashunin, D. A. "Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs." In: *arXiv:1603.09320 [cs]* (Mar. 30, 2016). arXiv: 1603.09320.

[44] McKinney, W. "Data Structures for Statistical Computing in Python." In: *Proceedings of the 9th Python in Science Conference*. Ed. by van der Walt, S. and Millman, J. Austin, Texas, USA, 2010, pp. 51–56.

[45] Muth, T., Renard, B. Y., and Martens, L. "Metaproteomic Data Analysis at a Glance: Advances in Computational Microbial Community Proteomics." In: *Expert Review of Proteomics* 13.8 (Aug. 2, 2016), pp. 757–769. DOI: 10.1080/14789450.2016.1209418.

[46] Na, S. and Paek, E. "Software Eyes for Protein Post-Translational Modifications." In: *Mass Spectrometry Reviews* 34.2 (Apr. 2015), pp. 133–147. DOI: 10.1002/mas.21425.

[47] Park, C. Y., Klammer, A. A., Käll, L., MacCoss, M. J., et al. "Rapid and Accurate Peptide Identification from Tandem Mass Spectra." In: *Journal of Proteome Research* 7.7 (July 3, 2008), pp. 3022–3027. DOI: 10.1021/pr800127y.

[48] Payne, S. H., Monroe, M. E., Overall, C. C., Kiebel, G. R., et al. "The Pacific Northwest National Laboratory Library of Bacterial and Archaeal Proteomic Biodiversity." In: *Scientific Data* 2 (Aug. 18, 2015), p. 150041. DOI: 10.1038/sdata.2015.41.

[49] Perez, F. and Granger, B. E. "IPython: A System for Interactive Scientific Computing." In: *Computing in Science & Engineering* 9.3 (June 18, 2007), pp. 21–29. DOI: 10.1109/MCSE.2007.53.

[50] Perez-Riverol, Y., Alpi, E., Wang, R., Hermjakob, H., et al. "Making Proteomics Data Accessible and Reusable: Current State of Proteomics Databases and Repositories." In: *PROTEOMICS* 15 (5-6 Mar. 2015), pp. 930–950. DOI: 10.1002/pmic.201400302.

[51] Ramakrishnan, S. R., Mao, R., Nakorchevskiy, A. A., Prince, J. T., et al. "A Fast Coarse Filtering Method for Peptide Identification by Mass Spectrometry." In: *Bioinformatics* 22.12 (June 15, 2006), pp. 1524–1531. DOI: 10.1093/bioinformatics/btl118.

[52] Renard, B. Y., Kirchner, M., Monigatti, F., Ivanov, A. R., et al. "When Less Can Yield More - Computational Preprocessing of MS/MS Spectra for Peptide Identification." In: *PROTEOMICS* 9.21 (Nov. 2009), pp. 4978–4984. DOI: 10.1002/pmic.200900326.

[53] Shao, W. and Lam, H. "Tandem Mass Spectral Libraries of Peptides and Their Roles in Proteomics Research." In: *Mass Spectrometry Reviews* 36.5 (Sept. 2017), pp. 634–648. DOI: 10.1002/mas.21512.

[54] *spotify/annoy: Approximate Nearest Neighbors in C++/Python optimized for memory usage and loading/saving to disk*. https://github.com/spotify/annoy. (accessed: 2017-10-04).

bioRxiv preprint doi: https://doi.org/10.1101/326173; this version posted May 19, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY 4.0 International license.

[55] Stein, S. E. and Scott, D. R. "Optimization and Testing of Mass Spectral Library Search Algorithms for Compound Identification." In: *Journal of the American Society for Mass Spectrometry* 5.9 (Sept. 1994), pp. 859–866. DOI: 10.1016/1044-0305(94)87009-8.

[56] The, M. and Käll, L. "MaRaCluster: A Fragment Rarity Metric for Clustering Fragment Spectra in Shotgun Proteomics." In: *Journal of Proteome Research* 15.3 (Mar. 4, 2016), pp. 713–720. DOI: 10.1021/acs.jproteome.5b00749.

[57] Van der Walt, S., Colbert, S. C., and Varoquaux, G. "The NumPy Array: A Structure for Efficient Numerical Computation." In: *Computing in Science & Engineering* 13.2 (Mar. 2011), pp. 22–30. DOI: 10.1109/MCSE.2011.37.

[58] Vizcaíno, J. A., Csordas, A., del Toro, N., Dianes, J. A., et al. "2016 Update of the PRIDE Database and Its Related Tools." In: *Nucleic Acids Research* 44 (D1 Jan. 4, 2016), pp. D447–D456. DOI: 10.1093/nar/gkv1145.

[59] Vu, T. N., Bittremieux, W., Valkenborg, D., Goethals, B., et al. "Efficient Reduction of Candidate Matches in Peptide Spectrum Library Searching Using the Top k Most Intense Peaks." In: *Journal of Proteome Research* 13.9 (Sept. 5, 2014), pp. 4175–4183. DOI: 10.1021/pr401269z.

[60] Walsh, C. T., Garneau-Tsodikova, S., and Gatto, G. J. "Protein Posttranslational Modifications: The Chemistry of Proteome Diversifications." In: *Angewandte Chemie International Edition* 44.45 (Nov. 18, 2005), pp. 7342–7372. DOI: 10.1002/anie.200501023.

[61] Waskom, M., Botvinnik, O., O'Kane, D., Hobson, P., et al. *Mwaskom/Seaborn: V0.8.1 (September 2017)*. Sept. 3, 2017. DOI: 10.5281/zenodo.883859.

[62] Ye, D., Fu, Y., Sun, R.-X., Wang, H.-P., et al. "Open MS/MS Spectral Library Search to Identify Unanticipated Post-Translational Modifications and Increase Spectral Identification Rate." In: *Bioinformatics* 26.12 (June 15, 2010), pp. i399–i406. DOI: 10.1093/bioinformatics/btq185.

[63] Yu, F., Li, N., and Yu, W. "PIPI: PTM-Invariant Peptide Identification Using Coding Method." In: *Journal of Proteome Research* 15.12 (Dec. 2, 2016), pp. 4423–4435. DOI: 10.1021/acs.jproteome.6b00485.