BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 12 2006, pages 1524–1531
doi:10.1093/bioinformatics/btl118

Data and Text mining

A fast coarse filtering method for peptide identification by mass spectrometry

Smriti R. Ramakrishnan[1,*], Rui Mao[1], Aleksey A. Nakorchevskiy[3], John T. Prince[2], Willard S. Willard[1], Weijia Xu[1], Edward M. Marcotte[2,3] and Daniel P. Miranker[1,2]

[1]Department of Computer Sciences, [2]Institute for Cellular and Molecular Biology and [3]Department of Chemistry and Biochemistry, The University of Texas at Austin, Austin, Texas 78712, USA

Received on August 17, 2005; revised on March 24, 2006; accepted on March 25, 2006
Advance Access publication April 3, 2006
Associate Editor: Satoru Miyano

ABSTRACT

Motivation: We reformulate the problem of comparing mass-spectra by mapping spectra to a vector space model. Our search method leverages a metric space indexing algorithm to produce an initial candidate set, which can be followed by any fine ranking scheme.

Results: We consider three distance measures integrated into a multi-vantage point index structure. Of these, a semi-metric fuzzy-cosine distance using peptide precursor mass constraints performs the best. The index acts as a coarse, lossless filter with respect to the SEQUEST and ProFound scoring schemes, reducing the number of distance computations and returned candidates for fine filtering to about 0.5% and 0.02% of the database respectively. The fuzzy cosine distance term improves specificity over a peptide precursor mass filter, reducing the number of returned candidates by an order of magnitude. Run time measurements suggest proportional speedups in overall search times. Using an implementation of ProFound's Bayesian score as an example of a fine filter on a test set of *Escherichia coli* protein fragmentation spectra, the top results of our sample system are consistent with that of SEQUEST.

Contact: smriti@cs.utexas.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

High-throughput methods for the identification of peptide fragmentation spectra (tandem or MS/MS spectra) are becoming increasingly important owing to fast growing protein databases and the rate of data acquisition using modern instrumentation. Most tools today employ computationally expensive linear scans of large databases of theoretical spectra. For example, typical analyses of an LC/LC/MS/MS experimental data set using the popular BioWorks program (ThermoFinnigan) on a single processor takes on the order of half a day of computation time (30 000 scans against the *Escherichia coli* database). Furthermore, the search hits are only meaningful when ranked by a relatively computationally intensive probabilistic or statistical significance/relevance score (Yates III *et al*., 1995; Mann and Wilm, 1994; Perkins *et al*., 1999).

A determining factor of the computational expense of the search is the similarity measure used. A simple similarity measure is the shared peaks count (SPC), a count of common *m*/*z* values between two spectra. SPC does not account for small peak shifts intrinsic to mass spectra owing to measurement and calibration error of the mass spectrometer, nor does it account for larger peak shifts caused by post-translational peptide modifications and mutations (Pevzner *et al*., 2001). A common solution is to add modified copies of each spectrum to the database (Yates III *et al*., 1995) known as the virtual database approach (Pevzner *et al*., 2001) However, given the 200+ known protein modifications (Gooley and Packer, 1997), this method soon results in exponential blowup of database size owing to combinatorial explosion. The virtual database approach clearly does not scale and linear scans become even more unacceptable. As an alternative, Pevzner *et al*. (2001) proposed an $O(n^2k)$ dynamic programming distance measure that can match two n-dimensional spectra that are up to k peak modifications apart. In the context of current approaches that use linear scans of large databases (size D), this measure must be evaluated for every entry in the database [total time complexity of $O(n^2kD)$].

We present a 'coarse filtering-fine ranking' scheme for protein identification. A coarse filter is a fast computation that produces a solution set (candidate set) with many false positives without eliminating any true positives. The computation is often a lower bound on more accurate matching functions, and hence less computationally intensive. Our search methodology consists of a coarse filtering stage that improves on the shared peaks count, followed by a fine filtering stage in which the candidate spectra output by the coarse filter are ranked by some significance score. As an example of a fine filter we implemented a version of ProFound's (Zhang and Chait, 2000) Bayesian scoring scheme. Coarse filters reduce the search time per query and in high-throughput proteomics, total expected speedup is the speedup per query multiplied by the number of queries.

Scalable coarse filtering may also improve overall search accuracy by facilitating the use of more discriminative, computationally expensive measures on the reduced candidate set, with little increase in search time. For example, approaches that combine the virtual database approach with complex distance functions similar to Pevzner *et al*. (2001) may start to become feasible.

Coarse filtering algorithms have been applied successfully to genomic (Williams, 1998) and speech signal database indexing (Keogh, 2002). Coarse filtering algorithms for genome databases have traditionally drawn inspiration from text (Faloutsos and Oard, 1996) and image retrieval (Smith and Chang, 1996). Our filter, based on metric space indexing, leverages the vector space model

*To whom correspondence should be addressed.

from information retrieval. Documents are commonly represented as sparse high dimensional vectors, where the *i*-th entry represents a measure of occurrence-frequency of the *i*-th word. Matching similar images is also often accomplished by comparing high dimensional histograms of image color (frequency spectra). We represent mass spectra as high dimensional vectors of mass/charge (*m*/*z*) values, creating a search space similar to ones used for documents.

We consider three distance measures for comparison of mass spectra. The first is derived from the cosine similarity measure, and is adapted to account for peak shifts in experimental spectra. The second is similar to the first but includes peptide precursor mass constraints. This method achieves maximum reduction in search time. We also investigated Hamming distance on reduced dimension boolean spectra vectors. We present an empirical evaluation of the different distance functions, based on search time and accuracy of results. In all cases, the candidate set for fine filtering was substantially reduced.

Metric space indexing in high dimensional spaces is difficult because nearest neighbor and range query (Chavez *et al*., 2001) algorithms have an exponential dependency on the dimension of the space (Chazelle, 1994). In our case, a semi-metric distance function is most effective at reducing search time by effectively reducing the intrinsic dimensionality of the space. We find that semi-metric searches on a multiple vantage point (MVP) index tree may be approximate, but achieve better search efficiency (pruning).

Section 2 gives a brief overview of related work in protein identification by mass spectrometry. Section 3 introduces metric space indexing and Section 4 details our distance functions for spectra comparison. Section 4.5 introduces semi-metric searches on MVP trees, Section 5 contains empirical results and Sections 6 and 7 contain a detailed discussion of the results and conclusions.

2 RELATED WORK

In bottom-up proteomics, a mass spectrum is a histogram of constituent *m*/*z* ratios of a set of peptides generated from either the enzymatic digestion of a protein (yielding the peptide mass fingerprint or PMF spectrum), or the induced fragmentation of a single peptide (yielding the peptide fragmentation fingerprint or PFF spectrum). When compared against a database of theoretical spectra, a sufficient number of accurately measured *m*/*z* peaks can identify a protein within acceptable statistical significance scores (Zhang and Chait, 2000). In tandem MS, fewer, but more precise, fragmentation spectra can uniquely identify the protein. However, automated searches must account for calibration errors, post-translational peptide modifications and mutations which introduce peak shifts into the experimental spectra.

Several approaches to *in silico* protein identification using MS have been described in the literature. While SPC is an intuitive measure of similarity, its accuracy diminishes drastically in the presence of peak shifts owing to mutations and/or modifications (Pevzner *et al*., 2001). MASCOT (Perkins *et al*., 1999), MS-FIT (Clauser *et al*., 1999) and ProFound (Zhang and Chait, 2000) use statistical or probabilistic scoring schemes that improve on the shared peaks count. MASCOT and MS-FIT are based on MOWSE, (Pappin *et al*., 1993) a scoring scheme that uses the normalized distribution frequency of peptides in the sequence database. MASCOT reports statistical significance levels and expect values for the MOWSE score. ProFound uses a Bayesian scoring scheme. In a recent survey of the three systems, ProFound gave the largest number of correct identifications (Chamrad *et al*., 2004). Popular tools for MS/MS identification include TurboSEQUEST (Yates III *et al*., 1995) and MASCOT (Perkins *et al*., 1999). Band optimization (Sankoff and Kruskal, 1983) has been used for matching gene sequences (Chao *et al*., 1992) and in speech recognition using dynamic time warping (Sakoe and Chiba, 1978). Applying this technique on Pevzner *et al*.'s dynamic programming measure (Pevzner *et al*., 2001) could reduce time complexity. We believe our fuzzy cosine distance search space (Section 4.2) is very similar to band optimization algorithms.

3 METRIC SPACE INDEXING

A metric space (V, D_{met}) is defined by a non-empty set V and a non-negative distance function $D_{\text{met}}(v_1, v_2)$ over objects v_1, v_2 in V, that satisfies the following conditions:

(1) $D_{\text{met}}(v_1, v_2) = 0$ iff $v_1 = v_2$ (identity)

(2) $D_{\text{met}}(v_1, v_2) = D_{\text{met}}(v_2, v_1)$ (symmetry)

(3) $D_{\text{met}}(v_1, v_2) + D_{\text{met}}(v_2, v_3) \geq D_{\text{met}}(v_1, v_3)$ (triangle inequality)

A distance function that satisfies identity and symmetry but fails the triangle inequality is called a semi-metric. A distance function D which fails the identity condition in one direction ($\exists v_1 \neq v_2$, $D(v_1, v_2) = 0$) is called a pseudometric. A function with both these properties is called a semi-pseudometric. In this paper, we use semi-pseudometric interchangeably with semi-metric.

A range query on a metric space $M = (\text{V}, D_{\text{met}})$ returns all points v within a given distance r from a query q ($D_{\text{met}}(v, q) \leq r$). A *k*-nearest neighbour (*k*-NN) query on M returns the k closest points to query point q. The radius bounded *k*-NN query, used in this paper, returns up to k points that are within distance r from q. Using the triangle inequality, an index built over a metric space can avoid distance computations with points unlikely to be within radius r of q, as described in Section Section 4.6, and reduce search time.

In a pivot-based index structure (Chavez *et al*., 2001), like the VP tree, the search space is partitioned into disjoint regions recursively. In each recursion, one or more pivots (vantage points or VPs) are first selected. Then, the data points are partitioned into two (or more) disjoint branches based on their distances from the pivot(s). Multi-Vantage Point, or MVP-trees (Bozkaya and Ozsoyoglu, 1997) extend VP-trees by increasing the number of disjoint datasets into which a dataset is partitioned. Specifically, in a metric space search of radius r for query q, given a pivot p_i and a metric distance function d, we would prune all points u such that

$$|d(u, p_i) - d(q, p_i)| > r. \quad (1)$$

An ideal search proceeds down only one branch of each pivot, effectively pruning all points in other branches, by applying the triangle inequality on the pivot and query points. However more than one branch may be pursued depending on the value of radius r in range search or k in *k*-NN.

4 RESULTS

4.1 Vector representation and distance functions

To leverage the properties of a metric space, a metric distance function with a suitable data representation must first be defined.

We investigated two data representations and three distance metrics. While indexing mass spectra in a metric space can greatly reduce the size of the initial candidate set, it must also be discriminative enough to return a relevant candidate set. Finding a suitable balance between the two objectives is difficult and approximate searches are always accompanied by a recall-precision (sensitivity-specificity) tradeoff. Intuitively, any combination of data representation and distance metric must ensure that we count peaks that differ by one or more known amounts. We investigated a high-dimensional data representation as defined below with less precise distance metrics (fuzzy and tandem cosine distance) and a low-dimensional data representation with an exact distance metric (Hamming distance, discussed in the Supplementary Material).

Given a list of m/z peaks P, resolution $0 \leq M_{\text{res}} < 1.0$ Da, and mass range $[M1, M2]$ Da, we define a high-dimensional boolean vector S in $(0, 1)^N$ space, where $N = (M2 - M1 + 1)/M_{\text{res}}$:

$$S[i] = \begin{cases} 1 & \exists p \in P, (p - M_{\text{res}} * i) \leq M_{\text{res}}, \text{ and} \\ & (p - M_{\text{res}} * (i-1)) > M_{\text{res}} \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

The second condition ensures that each peak in P maps to only one non-zero entry in S ($0 \leq i \leq N$). For a mass range [100, 5000] Da, and $M_{\text{res}} = 0.1$ Da, S is a sparse ~49 000 dimension vector. Though Equation (2) defines equi-sized boolean vectors, our implementation uses non-boolean compressed vectors storing m/z values directly.

4.2 Fuzzy cosine distance

Given two N-dimensional boolean vectors, A and B as defined above, where $a_i = A[i]$ and $b_j = B[j]$, and a peak mass tolerance $\tau_{\text{ms}} \geq M_{\text{res}}$, we can define SPC within a peak tolerance window $t = \tau_{\text{ms}}/M_{\text{res}}$ as

$$\text{SPC}_\tau(A,B) = \sum_i \text{match}(a_i, b_j);\ j \in [i-t, i+t] \quad (3)$$

$$\text{match}(a_i, b_j) = \begin{cases} 1 & a_i = b_j = 1 \\ & \text{match}(a_m, b_j) = 0, m \in [1, i) \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

Equation (4) counts two peaks as equal (a match) if they lie within t vector elements of each other. The second condition ensures that one peak counts exactly as one match—multiple matches are not counted. We observe that for zero peak tolerance ($\tau_{\text{ms}} = 0, t = 0$) the shared peaks count reduces to the dot product on boolean vectors.

$$\text{SPC}_\tau(A,B) = \sum_i \text{match}(a_i, b_i) = A \cdot B. \quad (5)$$

We also note that cosine similarity is defined as the normalized dot-product between two vectors.

$$\text{Cos}(A,B) = \frac{A \cdot B}{\|A\|\,\|B\|}, \quad (6)$$

where $\|A\|$ is the L2 norm over vector A. Modifying Equation (6) for $\tau_{\text{ms}} > 0$, we define a fuzzy cosine similarity measure:

$$\text{Cos}_\tau(A,B) = \frac{\text{SPC}_\tau(A,B)}{\|A\|\,\|B\|}. \quad (7)$$

Finally in Equation (8) we define fuzzy cosine distance, D_{ms}, as the inverse cosine of Cos_τ. D_{ms} is a semi-pseudometric; as a consequence of the tolerance window, D_{ms} may not satisfy the triangle inequality and it is possible that $D_{\text{ms}}(A, B) = 0$, $A \neq B$.

$$D_{\text{ms}}(A,B) = \arccos(\text{Cos}_\tau(A,B)). \quad (8)$$

4.3 Tandem cosine distance

Peptides with vastly differing precursor masses are unlikely to be similar, and should be further apart in the vector space. Tandem cosine distance factors a peptide precursor mass term into fuzzy cosine distance. Given peptide sequences A, B and precursor masses M_A, M_B, we define tandem cosine distance D_{tcd} as

$$D_{\text{tcd}}(A,B) = c_1 D_{\text{ms}}(A,B) + c_2 D_{\text{pm}}(A,B), \quad (9)$$

where c_1, c_2 are constants. D_{pm} computes absolute difference in precursor mass within a tolerance window. In order to account for slight differences in analytical and experimentally measured precursor mass (Figure S5, Supplementary Material), we introduce a precursor mass tolerance factor, τ_{pm} and define D_{pm} as

$$D_{\text{pm}}(A,B) = \begin{cases} 0 & |M_A - M_B| \leq \tau_{pm} \\ |M_A - M_B| & \text{otherwise} \end{cases} \quad (10)$$

D_{pm} is also a semi-pseudometric owing to the precursor mass tolerance τ_{pm}, and by the additive property of metric spaces, so is tandem cosine distance D_{tcd}. However, D_{tcd} is a better coarse filter for reasons detailed in Section 4.5 and reduces search time drastically when compared with fuzzy cosine distance. Since the precursor mass error between predicted and measured spectra, like peak mass error, can be modeled as following a normal distribution (Zhang and Chait, 2000), an exponential term in place of the linear D_{pm} might be a more faithful representation of the error model. We could replace D_{pm} by $D_{\text{exp}} = 1 - S_{\text{exp}}$, where $S_{exp} = c_2 \exp^{-c_1|M_A - M_B|^2}$. However, D_{exp} is not a metric distance, and even metric space mappings that are approximations of the normal distribution have high intrinsic dimensionality (we tried $S_{\text{exp}} = \exp^{-c_1|M_A - M_B|}$); suggesting poor metric space index performance as discussed in Section 4.5 (for details see Figures 6a and 6b in the Supplementary Material).

4.4 Comparison with a simple precursor mass filter

D_{pm} is mathematically equivalent to a simple precursor mass based filter. To compare the effectiveness of adding a fuzzy cosine distance term, we compared the performance of a metric space index using both a simple precursor mass filter (D_{pm}) and a linear combination of precursor mass filter and fuzzy cosine distance (D_{tcd}). As Figure 1 shows, the percentage of database returned by D_{tcd} is an order of magnitude less than that returned by D_{pm}. D_{pm} returns a large number of false positives, which are eliminated by D_{tcd}'s fuzzy cosine distance term. Even though D_{pm} computes a smaller number of distances (faster and coarser filter) at acceptable radii, both functions search <0.5% of the database in this experiment (Figure S4, Supplementary Material). Our coarse filter function, D_{tcd}, thus effectively combines both the speed of simple precursor mass filtering and the higher accuracy of fuzzy cosine distance into a single distance function. In fact, using SEQUEST as an ad hoc example fine filter on the output of our D_{tcd} coarse filter on Database II, search times were about seven times faster per query than SEQUEST searches on the complete Database II (*E.coli* + Human + 7 protein). Section 6 has a larger discussion on run time measurement issues.

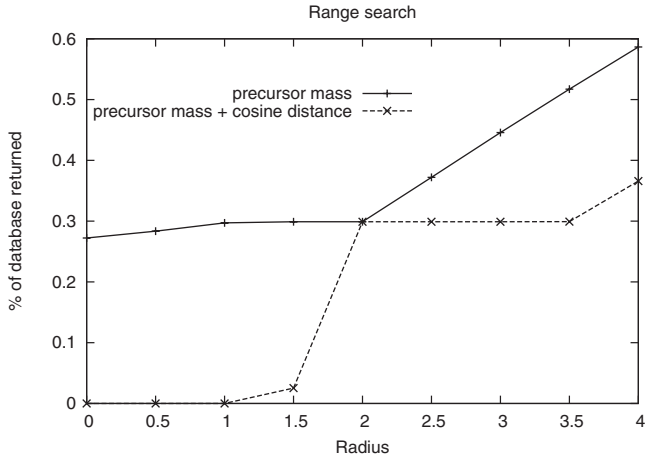

Fig. 1. The fuzzy cosine distance term in D_{tcd} improves specificity over a simple precursor mass based filter, returning about an order of magnitude fewer results. D_{tcd} returns only about 0.02% of the database at acceptable radius ($R = 1.48$ for D_{tcd}), while the precursor mass based filter, D_{pm}, returns 0.25% of the database at acceptable radius ($R = 0.0$ for D_{pm}) on Database I using $\tau_{\text{pm}} = 2.0$ Da.

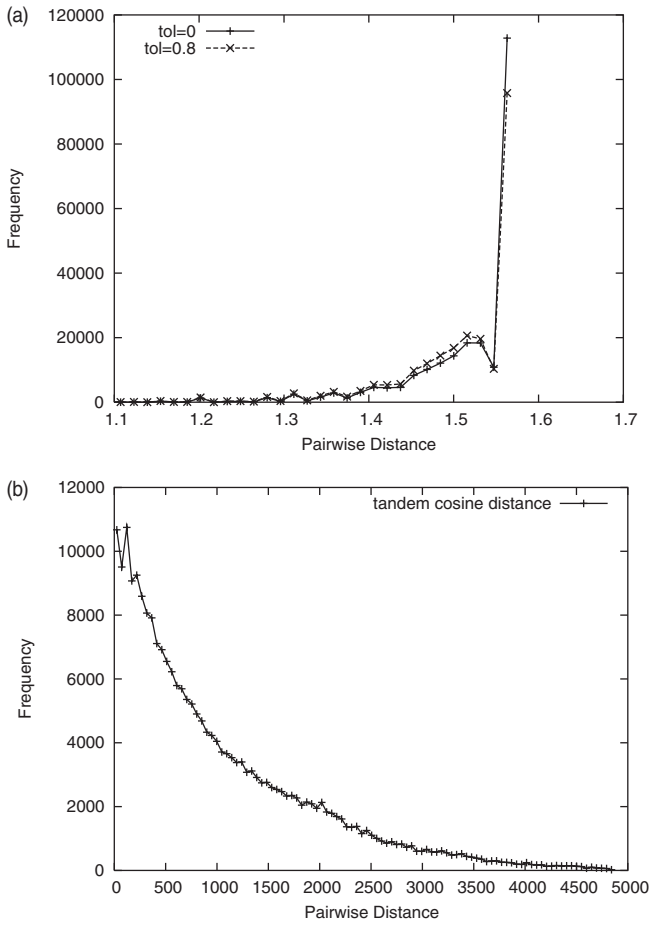

Fig. 2. Frequency plots of distances between pairs of spectra using exact cosine distance (peak tolerance $\tau_{\text{ms}} = 0$ Da) and fuzzy cosine distance ($\tau_{\text{ms}} = 0.8$ Da) in (**a**) and tandem cosine distance in (**b**). From the graphs, exact and fuzzy cosine distance in (a) have lower variance σ^2, and thus higher intrinsic dimensionality $\rho = (\mu^2/2\sigma^2)$—making these distance metrics less suitable for metric space indexing than tandem cosine distance. Intrinsic dimensionality is also expected to reduce with increase in the semi-metric nature of the search (increasing τ_{ms} and precursor mass tolerance τ_{pm}). Indeed, for fuzzy cosine distance: $\rho \simeq 579$ ($\tau_{\text{ms}} = 0$ Da), $\rho \simeq 445$ ($\tau_{\text{ms}} = 0.2$ Da) and $\rho \simeq 176$ ($\tau_{\text{ms}} = 2.2$ Da). For tandem cosine distance: $\rho \simeq 0.62$ ($\tau_{\text{ms}} = 0.2$ Da, $\tau_{\text{pm}} = 2.0$ Da). Exact and fuzzy cosine range from 0 to $\pi/2$ and tandem cosine ranges from 0 to about 5000. Spectra were randomly sampled from Database III in Table 1.

4.5 Reducing the intrinsic dimensionality

The curse of dimensionality (Chavez *et al*., 2001) refers to the phenomenon of algorithmic performance degrading exponentially with increase in intrinsic dimensionality. Dimensionality is not easily defined for metric spaces with no geometric restrictions on objects. An alternative is to define the intrinsic dimensionality as $\rho = (\mu^2/2\sigma^2)$ where μ and σ are the mean and variance of a histogram of pairwise distances (Chavez *et al*., 2001). Owing to high intrinsic dimensionality of the search space, an exact metric space solution to our problem suffers from the curse of dimensionality and is only slightly more efficient than a linear scan.

In the metric space indexing world, histograms with relatively smaller means and larger variances usually indicate better search efficiency (Chavez *et al*., 2001). Pairwise distance histograms of exact and fuzzy cosine distance in Figure 2a have small variances and large means. A corresponding Figure 2b for tandem cosine distance has larger variance and smaller mean indicating that tandem cosine distance is a better distance function for this index (the terms 'small' and 'large' are relative to the range of expected distance values). Also, semi-metric distance functions may have the effect of reducing the intrinsic dimensionality of the search space. This has also been observed in document vector spaces (Skopal *et al*., 2004). Values for ρ support this hypothesis (Fig. 2). However, we need to modify standard metric space indexes to use semi-metric distance functions.

4.6 Modifying the index for a semi-metric search

A distance function d is a semi-metric if it fails the triangle inequality i.e. $d(q, p) + d(p, u) < d(q, u)$. However, if there exists some upper bound κ such that $d(q, p) + d(p, u) + \kappa \geq d(q, u)$, we say d fails the triangle inequality by this amount κ. In this case, there may exist some point u and query q such that $d(q, u) + \kappa > r$, but $d(q, u) < r$, causing u to be incorrectly pruned. However, if we can predict an upper bound, κ_u on κ, the metric space index equations can be adjusted (Sahinalp *et al*., 2003) or fixed to return exact results. We briefly describe our procedure for the case of MVP-trees. Equation (1) can be modified to

$$|d(u, p_i) - d(q, p_i)| > (r + \kappa_u). \tag{11}$$

All points lying within distance r from the query are returned—only the pruning equations are adjusted using κ_u. For tandem cosine distance, we can derive (proof omitted) a loose upper bound $\kappa_u = (\pi/2) + 2\tau_{\text{pm}}$, when every peak in one vector differs from its corresponding matching peak in the other spectrum by the peak tolerance τ_{ms}. In practice however, setting $\kappa = \kappa_u$ generates a large number of false positives owing to conservative

Table 1. Description of test databases

Database	Database description	Database size	Test set size
Database I	*E.Coli* K12 + 7 protein mix	92 769	49 (7 protein mix)
Database II	Database I + Human	654 276	49 (7 protein mix)
Database III	*E.Coli* K12	92 373	14 (*E.coli*)

Acceptable radius for D_{tcd} is 1.48 for Database I and II and 1.46 for Database III. Acceptable k values for k-NN search are $k < 20$ for Databases I and II. The test set for Database III is from the Open Proteomics Database (Prince *et al*., 2004), accession number opd00006_ECOLI.

Table 2. Digest parameters for Database I of Table 1

E.coli K12 proteins	4824
Mixed proteins	15 (7 plus isoforms of Myosin)
Total database size	92 769 fragmentation spectra
Fragment mass tolerance	0.2 Da
Precursor mass tolerance	2.0 Da
Known modifications	none
Charge state	+2
Ion type	b, y
Missed cleavages	0
Protease	Trypsin
Mass range	0–5000 Da

pruning. Searches using lower κ values may be approximate, but are faster owing to aggressive pruning opportunities. A κ that maintains reasonable accuracy is dataset dependent and must be empirically determined. We set $\kappa = \tau_{\text{ms}} + \tau_{\text{pm}} \leq \kappa_u$ and this was sufficient to retrieve all true positives, while keeping the number of false positives small.

5 EXPERIMENTAL EVALUATION

The underlying MVP-tree implementation is part of MoBIoS (Miranker and Mao, 2003), a special purpose database management system for molecular biology. MoBIoS comprises an object-relational storage manager extended with metric-space indexes, a query language (MoBIoS SQL or mSQL), and built-in data types for biological sequences (Xu *et al*., 2004) and mass-spectra. We ran range and k-NN queries using the MVP tree index structure modified to incorporate semi-metric searches.

Our experiments test the capability of the index to prune distance computation, reducing the number of distance computations and in turn wall clock time, while returning all true positives (i.e. identifying all 49 spectra and thus seven proteins correctly in Databases I and II) and limiting the size of the result set. The test databases and query sets in Table 1 are open source proteomics data from the Open Proteomics Database (Prince *et al*., 2004) and Sashimi (sashimi.sourceforge.net). Database I contains theoretical spectra from the *E.coli* K12 (*E.coli*) genome and a seven protein mixture from the Sashimi proteomics repository. Database II combines Database I with theoretical mass spectra from the human genome. The digest parameters used for all databases are in Table 2. We constructed our ground truth set (query set) of 49 spectra by first identifying the 4000+ scans of the Sashimi seven protein mixture, using BioWorks 3.1 (peptide identification software), and choosing all +2 charged spectra that were identified with XCorr score >2.4. To determine the acceptable values of r and k given in Table 1, we plotted the percentage of true positives returned versus r (Figure 1a, Supplementary Material) and k (Figure 1b, Supplementary Material), and chose r and k as the smallest values for which all true positives were returned.

5.1 Index performance

We measure the percentage of database searched and returned for range and k-NN queries. Among all distance metrics, tandem cosine distance gave the best reduction in both measures. Figure 3 shows the percentage of database searched and returned for tandem cosine distance. At acceptable radius $r = 1.48$, tandem cosine distance searches an average of ~0.5% of the database and returns ~0.02% of the database. On the other hand, as shown in Figure 4, fuzzy cosine distance with no precursor mass difference term (D_{ms}) searches ~95% of the database at acceptable radii, and thus is not a good coarse filter. Figure S3 (Supplementary Material) plots the percentage of results returned by fuzzy cosine distance.

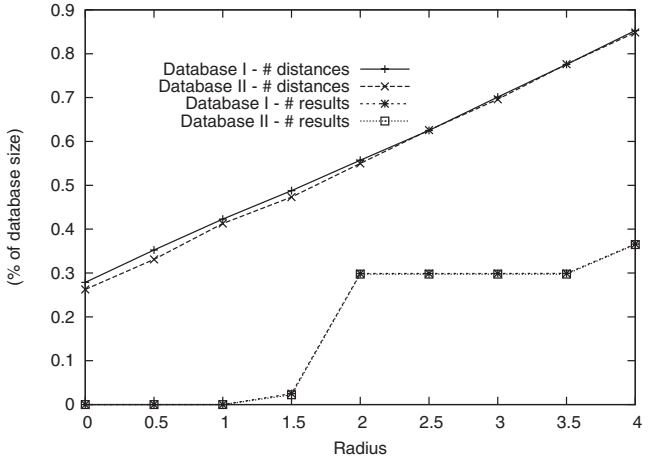

Fig. 3. Number of distance computations and returned results as a percentage of database size plotted against range query radius, using tandem cosine distance on Databases I and II. At an acceptable radius of $R = 1.48$, the number of returned results is <0.02% of the database size, and the number of distance computations is only 0.5% of the database size.

Figure 5 shows the percentage of database searched using a radius bounded k-NN search. The size of the candidate set is reduced to <20 spectra. Increasing k up to 100 on other databases, k-NN searches still produce smaller candidate sets (reduce the number of results returned) when compared with range searches. To summarize, both k-NN and range queries search <0.5% of the entire database, with k-NN producing smaller candidate sets.

5.2 Scalability

We want to ensure that the gains obtained using a coarse filtering stage scale with the size of the database. Ideally we want a constant number of distance computations per query, independent of database size. Radius bounded k-NN searches are expected to provide better scalability than range searches. To test performance with

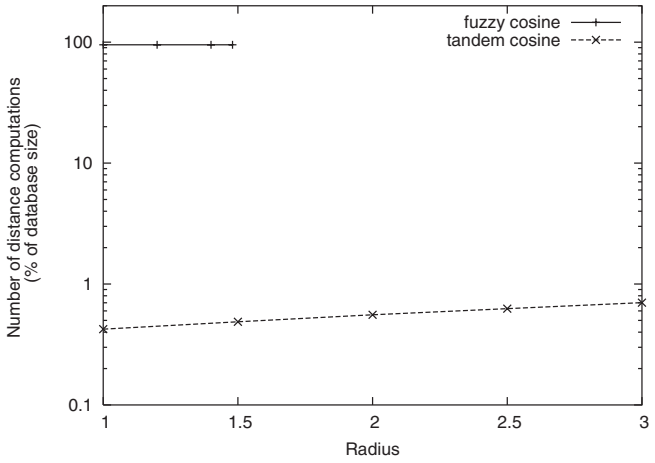

Fig. 4. Number of distance computations as a percentage of database size plotted against range query radius, using tandem cosine distance D_{tcd} and fuzzy cosine distances on Database I. D_{tcd} computes fewer distances, effectively searching a smaller percentage of the database. Acceptable $R = 1.48$ is the smallest radius at which all the true positives are returned.

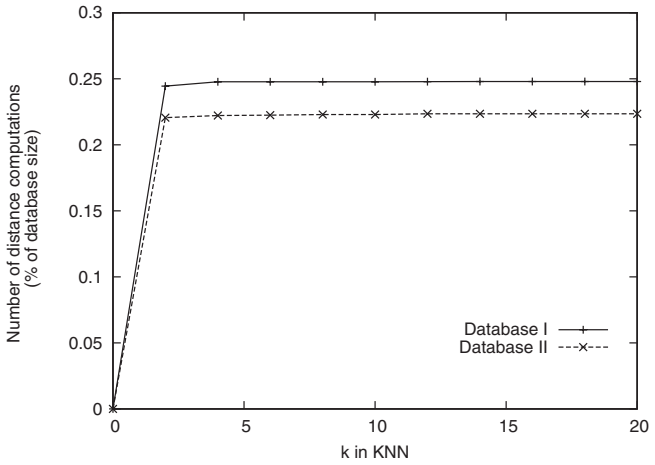

Fig. 5. Number of distance computations as a percentage of database size for radius bounded k-NN searches, using tandem cosine distance on Database I and II. The number of distance computations is <0.3% of the database size and the number of returned results is 'k'. Here, $k = 3$ (Database I) and $k = 16$ (Database II) are the smallest k at which all true positives are returned.

database size, we created multiple small databases from Database II in Table 1, which consists of 654 276 theoretical fragmentation spectra: *E.coli* and human proteins combined with the seven protein mixture used in Section 5.1. For each small database, we ran a radius bounded k-NN search, using bounding radius R and k that return 100% true positives. Figure 6 shows the number of distance computations per database. The number of distance computations remain near constant as database size increases for fixed k and R.

Using $k = 253$ across all databases, we get near constant scalability. $k = 100$ results in fewer distance computations but is less accurate, returning only 98% of the true positives for databases with >400 000 spectra. The third curve in Figure 6 uses $k = 100$ for a smaller databases (maintaining scalability and returning 100% true

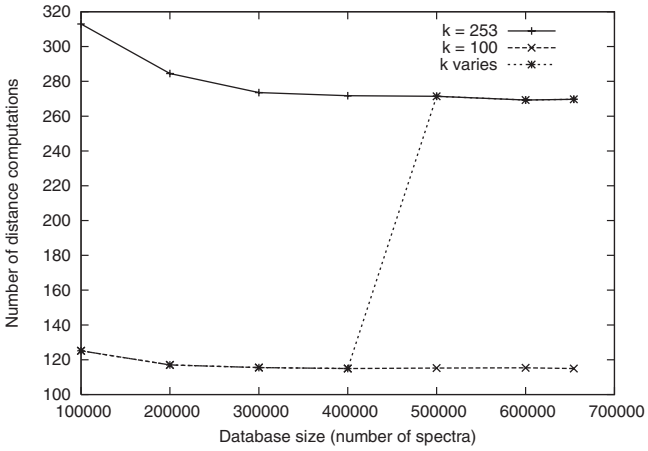

Fig. 6. Scalability results using radius bounded k-NN searches on Database II. The y-axis shows the number of distance calculations plotted against database size. The plot for $k = 253$ returns all true positives for all databases. The plot for $k = 100$ has fewer distance computations but returns only 98% of the true positives for databases > 400 000 spectra. The third curve (k varies: $k = 100$ for smaller databases and $k = 253$ for larger ones) returns all true positives for all databases, while still reducing distance computations.

positives at fewer distance computations) and then switches to a higher $k = 253$ for larger databases.

6 DISCUSSION

6.1 Fine filtering

Our coarse filter can be combined with any fine ranking scheme for mass spectra identification. Since fine ranking schemes are computationally expensive, an initial coarse filtering stage should reduce overall search time by reducing the number of spectra input to the fine ranking stage. We tested a combined coarse filtering-fine ranking system which ranks each result produced by the coarse filter, using a version of ProFound's (Zhang and Chait, 2000) Bayesian scoring scheme. We ran tests on a few pre-identified human peptide fingerprint spectra, as well as on 14 *E.coli* peptide fragmentation spectra (Database III in Table 1). We only use the spectra in Database III to illustrate the viability of a combined scheme for a number of reasons. First, the main goal of this paper is to identify a coarse filter suitable for mass spectra. Second, given high sample complexity and the difficulty in predicting the exact fragmentation pattern for a peptide of given sequence, it is nearly impossible to acquire unambiguously identified complex spectra to test this combined system. Third, the investigation of whether the Bayesian fine filtering scheme will provide high confidence identification of the protein mixtures in Databases I and II deserves a separate study, independent of the coarse filtering or indexing schemes.

To check correctness of the ranking scheme, we defined a ground truth set using the top hit per query from TurboSEQUEST (Yates III *et al.*, 1995). This top result is expected to be correct because it also had high protein and peptide probability scores from ProteinProphet and PeptideProphet (Nesvizhskii *et al.*, 2003; Keller *et al.*, 2002). In all queries, our fine filter ranked this 'correct answer' as the top hit, with an identification probability of >99% in most cases. The scores between first and second ranked peptides

differed by at least three and up to eight orders of magnitude in many cases. Prototype versions of the MoBIoSFound system (coarse filtering followed by fine filtering) for peptide mass fingerprinting and tandem MS identification are available online (http://aug.csres.utexas.edu:8080/). To summarize, this combined system demonstrates the feasability of combining the accuracy of a fine ranking scheme, with the speedup gain of the coarse filtering stage.

6.2 Scalability and run time measurements

We use a version of k-NN radius bounded search that relaxes the requirement that the k best neighbours are returned first (Xu *et al*., 2003); the nearest point will be returned first, but the next $k-1$ may not be the nearest to the query (Fig. 6). This version is expected to have better scalability, with acceptable accuracy. But since the top hit for our application is determined by the fine ranking phase, the topmost hit may not be 'nearest' to the query in the coarse filtering phase, especially as database size grows, and we have to increase k to achieve 100% accuracy. This suggests that lower k values can be used if a fine ranking 'metric' distance can directly be incorporated into the index, instead of a two-step coarse-fine filter solution.

A reason to account for the increase in k and in the number of distances between database size 400 000 and 500 000 in Figure 6 is that although the system is main-memory based in this study, the MoBIoS MVP-tree is organized for pagination to disk. Our MVP-tree implementation has discontinuous increases in height as the database grows, much like the depth of a B+ tree in relational databases (Ramakrishnan and Gehrke, 2003). Hence the performance is subject to sudden increments when the index increases height.

We have proposed that by simply feeding the coarse filter's output as input to a system like SEQUEST, we can achieve significant runtime speedup. However, comparative timing measurements require access to the internals of existing systems. For instance, we need to be able to measure both predicted database creation time and actual search time. We need I/O cost estimates to measure disk write times and also include possibilities of internal caching. With no source code access to proprietary systems, it is impossible to determine and compare the time taken by different stages, and timing measurements are not very meaningful as absolute measures of comparison. However, under simple assumptions of operation, available runtime studies are already encouraging. We compared SEQUEST run times on the entire predicted database against those on a reduced predicted database created from the candidate set identified by the coarse filter. For Database II queries, SEQUEST searches on the reduced databases were about seven times faster per query than searches on the entire Database II. We expect source code level integration of fine and coarse filters to result in higher speedups.

7 CONCLUSIONS

We identify a 'coarse filtering-fine ranking' metric space indexing approach for protein mass spectra database searches. Our coarse filter approach speeds up searches by reducing both the number of distance computations in the index search and the number of candidate spectra input to a fine filtering stage.

We achieve fast, lossless metric space indexing of high-dimensional mass spectra vectors by defining a number of biologically meaningful and computationally efficient distance measures that account for peak shift and precursor mass error. Tandem cosine distance is the most efficient of these, achieving maximal reduction in the intrinsic dimensionality of the search space. This enables the creation of indexes with sufficient pruning power to reduce the number of distance computations to <0.5% of the database and the number of candidates for fine filtering to ~0.02% of the database. We demonstrate scalable index perfomance for different database sizes using a version of k-NN search. Available runtime measurements support the speedup hypothesis.

The speedups owing to coarse filtering open up possibilities for the automatic detection of mutations and modifications. Currently, the virtual database approach is computationally infeasible owing to the exponential blowup in database size because of the addition of all theoretically possible modified spectra. Alternatives (Pevzner *et al*., 2001) use algorithmically expensive search functions. In a combined system, virtual databases can be large and distance functions can be more accurate and expensive (e.g. a metric distance approximation of the Pevzner *et al*., 2001 algorithm), since we search only a fraction of the database. An alternate solution to searching for mutations/modifications is to identify and evaluate other (semi) metric distances that take larger peak shifts into account, like coarse resolution Hamming distance with precursor mass constraints. Also, if we can identify a metric fine ranking function (or a metric approximation), fine scoring can be incorporated into the index; resulting in an integrated fast and accurate search model.

ACKNOWLEDGEMENTS

The authors acknowledge support from the National Science Foundation (DBI-0241180, IIS-0325116), National Institutes of Health, Welch Foundation and Packard Foundation.

Conflict of Interest: none declared.

REFERENCES

Bozkaya,T. and Ozsoyoglu,M. (1997) Distance-based indexing for high-dimensional metric spaces. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Tucson, Arizona, USA, pp. 357–368.

Chamrad,D. *et al*. (2004) Evaluation of algorithms for protein identification from sequence databases using mass spectrometry data. *Proteomics*, **4**, 619–628.

Chao,K. *et al*. (1992) Aligning two sequences within a specified diagonal band. *Comput. Appl. Biosci*., **8**, 481–487.

Chavez,E. *et al*. (2001) Searching in metric spaces. *ACM Comp. Surv*., **33**, 273–321.

Chazelle,B. (1994) Computational geometry: a retrospective. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing*, Montreal, Quebec, Canada, pp. 75–94.

Clauser,K. *et al*. (1999) Role of accurate mass measurement (± 10 p.p.m) in protein identification strategies employing MS or MS/MS and database searching. *Anal. Chem*., **71**, 2871–2882.

Faloutsos,C. and Oard,D. (1996) A Survey of Information Retrieval and Filtering Methods. *Technical Report*. University of Maryland, College Park, MD.

Gooley,A. and Packer,N. (1997) The importance of co- and post-translational modifications in proteome projects. *Proteome Research: New Frontiers in Functional Genomics*, Springer-Verlag, Berlin, New York, pp. 65–91.

Keller,A. *et al*. (2002) Empirical statistical model to estimate the accuracy of peptide identifications made by ms/ms and database search. *Anal. Chem*., **74** pp. 5383–5392.

Keogh,E. (2002) Exact indexing of dynamic time warping. In *Proceedings of the 28th International Conference on Very Large Data Bases*, Hong Kong, China, pp. 406–417.

Mann,M. and Wilm,M. (1994) Error-tolerant identification of peptides in sequence databases by peptide sequence tags. *Anal. Chem*., **66**, 4390–4399.

Mao,R., Xu,W., Ramakrishnan,S., Nuckolls,G. and Miranker,D.P. (2005) On optimizing distance-based similarity search for biological databases. In *Proceedings 4th International IEEE Computer Society Computational Systems Bioinformatics Conference*, Stanford, CA, USA, pp. 351–361.

Miranker,W.X.D. and Mao,R. (2003) Mobios: a metric-space dbms to support biological discovery. In *Proceedings of the International Conference on Scientific and Statistical Database Management System*, pp. 241.

Nesvizhskii,A. *et al*. (2003) A statistical model for identifying proteins by tandem mass spectrometry. *Anal. Chem.*, **75**, 4646–4658.

Pappin,D. *et al*. (1993) Rapid identification of proteins by peptide-mass fingerprinting. *Curr. Biol.*, **3**, 327–332.

Perkins,D. *et al*. (1999) Probability-based protein identification by searching sequence databases using mass spectrometry data. *Electrophoresis*, **20**, 3551–3567.

Pevzner,P. *et al*. (2001) Efficiency of database search for identification of mutated and modified proteins via mass spectrometry. *Genome Res.*, **11**, 290–299.

Prince,J. *et al*. (2004) The need for a public proteomics repository. *Nat. Biotechnol.*, **22**, 471–472.

Ramakrishnan,R. and Gehrke,J. (2002) *Database Management Systems*. McGraw-Hill Science/Engineering/Math.

Sahinalp,S., Tasan,M., Macker,J. and Özsoyoglu,Z. (2003) Distance based indexing for string proximity search. In *proceedings of the 19th International Conference on Data Engineering*, Bangalore, India, p. 125.

Sakoe,H. and Chiba,S. (1978) A dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans. Acoustics Speech Signal Proc.*, **26**.

Sankoff,D. and Kruskal,J. (1983) *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley, Massachussets.

Skopal,T., Moravec,P., Pokorný,J. and Snásel,V. (2004) Metric indexing for the vector model in text retrieval. In *Proceedings of 11th International Conference on String Processing and Information Retreival*, Badova, Italy, pp. 183–195.

Smith,J.R. and Chang,S.-F. (1996) Tools and techniques for color image retrieval. *Storage and Retrieval for Image and Video Databases*, San Jose, CA, pp. 426–437.

Williams,H. (1998) Cafe: an indexed approach to searching genomic databases. In *Proceedings 21st Annual International ACM SIGIR Conference on Research and Development in Information Retreival*, Melbourne, Australia, pp. 389.

Xu,W., Miranker,D., Mao,R. and Wang,S. (2003) Indexing protein sequences in metric space. *Technical report*, Deptartment of Computer Sciences, University of Texas at Austin.

Xu,W. *et al*. (2004) Using mobios‘ scalable genome join to find conserved primer pair candidates between two genomes. *Bioinformatics*, **20**, 355–362.

Yates III,J. *et al*. (1995) Method to correlate tandem mass spectral data of modified peptides to amino acid sequences in the protein database. *Anal. Chem.*, **67**, 1426–1436.

Zhang,W. and Chait,B. (2000) ProFound—an expert system for protein identification using mass spectrometric peptide mapping information. *Anal. Chem.*, **72**, 2482–2489.