Case Study: An Inverted Index for Mass Spectra Similarity Query and Comparison with a Metric-space Method

Rui Mao (Shenzhen University)
Smriti R. Ramakrishnan (Univ. of Texas at Austin)
Glen Nuckolls (NetApp)
Daniel P. Miranker (Univ. of Texas at Austin)

Motivation

What is metric space indexing good for?

Background

- MSFound , in the context of MoBIoS :
  - Ramakrishnan, S. R., Mao, R., Nakorchevskiy, A. A., Prince, J. T., Willard, W. S., Xu, W., Marcotte, E. M., and Miranker, D. P. 2006. A fast coarse filtering method for peptide identification by mass spectrometry. *Bioinformatics* 22, 12 (Jun. 2006), 1524-1531.
- A coarse filter
- Tandem cosine distance:
  - Precursor mass: number
  - Cosine distance
- Semi-metric

Outline

- Mass spectra
- Tandem cosine distance
- An inverted index method
- Empirical results
- Conclusions and Future work

Outline

- Mass spectra
- Tandem cosine distance
- An inverted index method
- Empirical results
- Conclusions and Future work

Mass spectra

- Each spectrum represents a fragment of a protein sequence.
  - M: precursor mass
  - P={p_i}: a list of real-valued m/z peaks
- Binary format: P → S={s_i, | s_i = 0 or 1}
  - Range: M_1Da ≤ p_i ≤ M_2Da
  - Resolution: 0 ≤ M_{res} ≤ 1.0Da
  - Commonly: [100, 5000] Da, M_{res} =0.2 Da, 25K dims

$$s[i] = \begin{cases} 1, \exists p \in P, i < \dfrac{(p - M_1)}{M_{res}} \le i+1 \\ 0, otherwise \end{cases}$$

Outline

- Mass spectra
- Tandem cosine distance
- An inverted index method
- Empirical results
- Conclusions and Future work

Tandem cosine distance

$D_{tcd}(A, B) = C_1 D_{ms}(A, B) + C_2 D_{pm}(A, B)$

– Precursor mass distance $D_{pm}(A, B)$:

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

– Fuzzy cosine distance $D_{ms}(A, B)$, $0 \le D_{ms}(A, B) < \pi/2$:

$$D_{ms}(A,B) = \arccos\left(\frac{SPC_t(A,B)}{\|S_A\| \|S_B\|} \right)$$

- Shared peak count with tolerance $SPC_t(A, B)$:

$$SPC_t(A,B) = \sum_{i:S_A[i]=1} match(i,B)$$

$$match(i,B) = \begin{cases} 1, \exists j \in [i-t, i+t], S_B[j] = 1, \text{j is not matched with other i} \\ 0, otherwise \end{cases}$$

Outline

- Mass spectra
- Tandem cosine distance
- An inverted index method
  - Bulkloading the index
  - Range query processing
  - Cost analysis
- Empirical results
- Conclusions and Future work

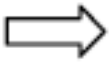
Bulkloading the index

- Index on the precursor mass: any 1-d index
- Inverted index on peaks S:
  - $L = \{L_i \mid L_i = \{j \mid S_j[i] = 1, j = 1, \ldots, M\}, i = 1, \ldots, N\}$
  - Compressed vector:

    $S' = [k_1, k_2, \ldots]$,

    $S[k_i] = 1$.

Compressed vectors
$S_1 = [1, 4]$
$S_2 = [1, 4, 5]$
$S_3 = [2, 4]$
$S_4 = [2]$
$S_5 = [2, 5]$
$S_6 = [5]$
$S_7 = [1, 5]$
$S_8 = [4]$



Inverted index
$L_1 = [1, 2, 7]$
$L_2 = [3, 4, 5]$
$L_3 = [\ ]$
$L_4 = [1, 2, 3, 8]$
$L_5 = [2, 5, 6, 7]$

Range query processing

Theorem 1: (1) A is a query result of range query R(q, r) if $M_q - \max(\tau_{pm}, (r - C_1\pi/2)/C_2) \le M_A \le M_q + \max(\tau_{pm}, (r - C_1\pi/2)/C_2)]$ and $r - C_1\pi/2 > 0$; (2) A is not a query result of range query R(q, r) if $M_A > M_q + \max(\tau_{pm}, r/C_2)$, or $M_A < M_q + \max(\tau_{pm}, r/C_2)$

Key idea: (1) $D_{ms} < \pi/2$; (2) $0 \le D_{ms}$

Range query processing (cont'd)

Gross shared peak count with tolerance t

$GSPC_t(q, A)$: number of appearances of A in lists of the inverted index related to q.

Let S_q =[3], t=1, then:

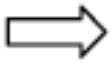
related lists:

L2, L3, L4

$GSPC_t(q, S_1) = 1$

$GSPC_t(q, S_3) = 2$

Compressed vectors
$S_1 = [1, 4]$
$S_2 = [1, 4, 5]$
$S_3 = [2, 4]$
$S_4 = [2]$
$S_5 = [2, 5]$
$S_6 = [5]$
$S_7 = [1, 5]$
$S_8 = [4]$



Inverted index
$L_1 = [1, 2, 7]$
$L_2 = [3, 4, 5]$
$L_3 = [\ ]$
$L_4 = [1, 2, 3, 8]$
$L_5 = [2, 5, 6, 7]$

Range query processing (cont'd)

Theorem 2: A is not a query result of range query R(q, r) if :

$$\arccos\left(\frac{GSPC_t(q,A)}{\|S_q\|\|S_A\|}\right) > \frac{r - C_2 D_{pm}(q,A)}{C_1}$$

Key idea: $SPC_t(q, A) \leq GSPC_t(q, A)$

Range query processing (cont'd)

1. Prune data using bounds of precursor mass computed from Theorem 1. Put data satisfying Theorem 1 (1) in to a result set, and data satisfying Theorem 1 (2) into a candidate set, together with their precursor mass distance to q.
2. Search inverted index to compute $GSPC_t(q, A)$ for any database point A that $GSPC_t(q, A) > 0$, and A appears in the candidate set.
3. Prune data in the candidate set using Theorem 2.
4. For each element of the candidate set, compute its fuzzy cosine distance using algorithm in Figure 1 to answer the query.

Figure 3. Steps of range query processing

Range query processing (cont'd)

Pruning statistics:

1. Precision:

$$\text{Precision} = \frac{\text{Number of results found in step 4}}{\text{Candidate set size before step 4}}$$

2. Pruning efficiency of Theorem 1:

$$PE_1 = 1 - \frac{\text{Candidate set size after step 1}}{\text{Database size}}$$

3. Pruning efficiency of Theorem 2:

$$PE_2 = 1 - \frac{\text{Candidate set size before step 4}}{\text{Candidate set size after step 1}}$$

Cost Analysis

- N: dimension
- M: database size
- p: sparsity, $P(s_i=1)$
- $|S'| = Np$
- $|L_i| = Mp$
- $\sum|L_i| = MNp$
- For q and t=0:

  $\sum|L_i| = M(1-(1-p)^{NP}) \approx MNp^2$

- Bulkload time: O(MlgM)
- Index size: O(MNp)
- Range query time:

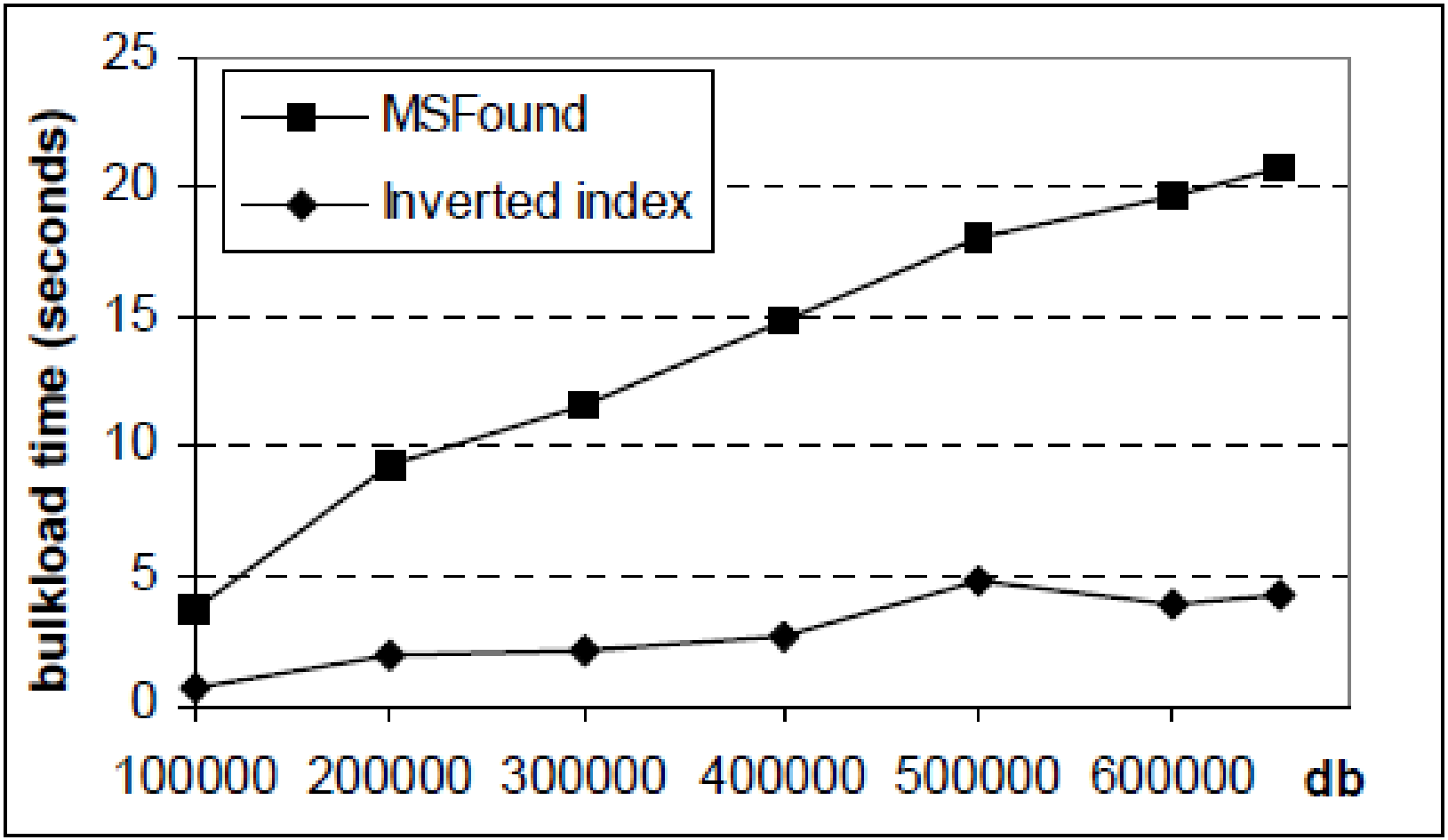
  $O(MNp^2)$, t=0
- I/O cost:

  should use continuous disk blocks

Outline

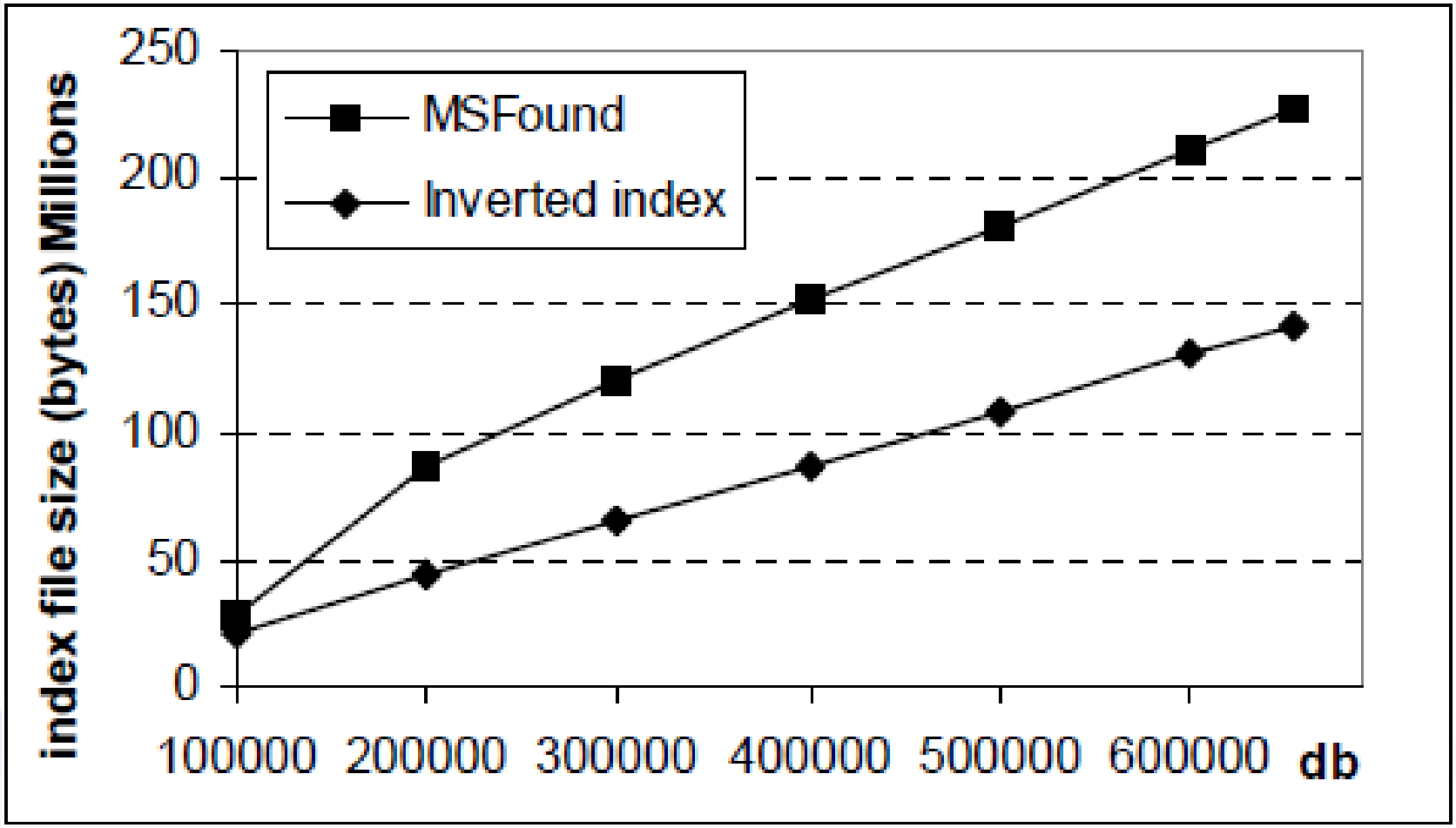
- Mass spectra
- Tandem cosine distance
- An inverted index method
- Empirical results
  - Bulkload time
  - Index file size
  - Range query time
- Conclusions and Future work

Empirical results: Bulkload time



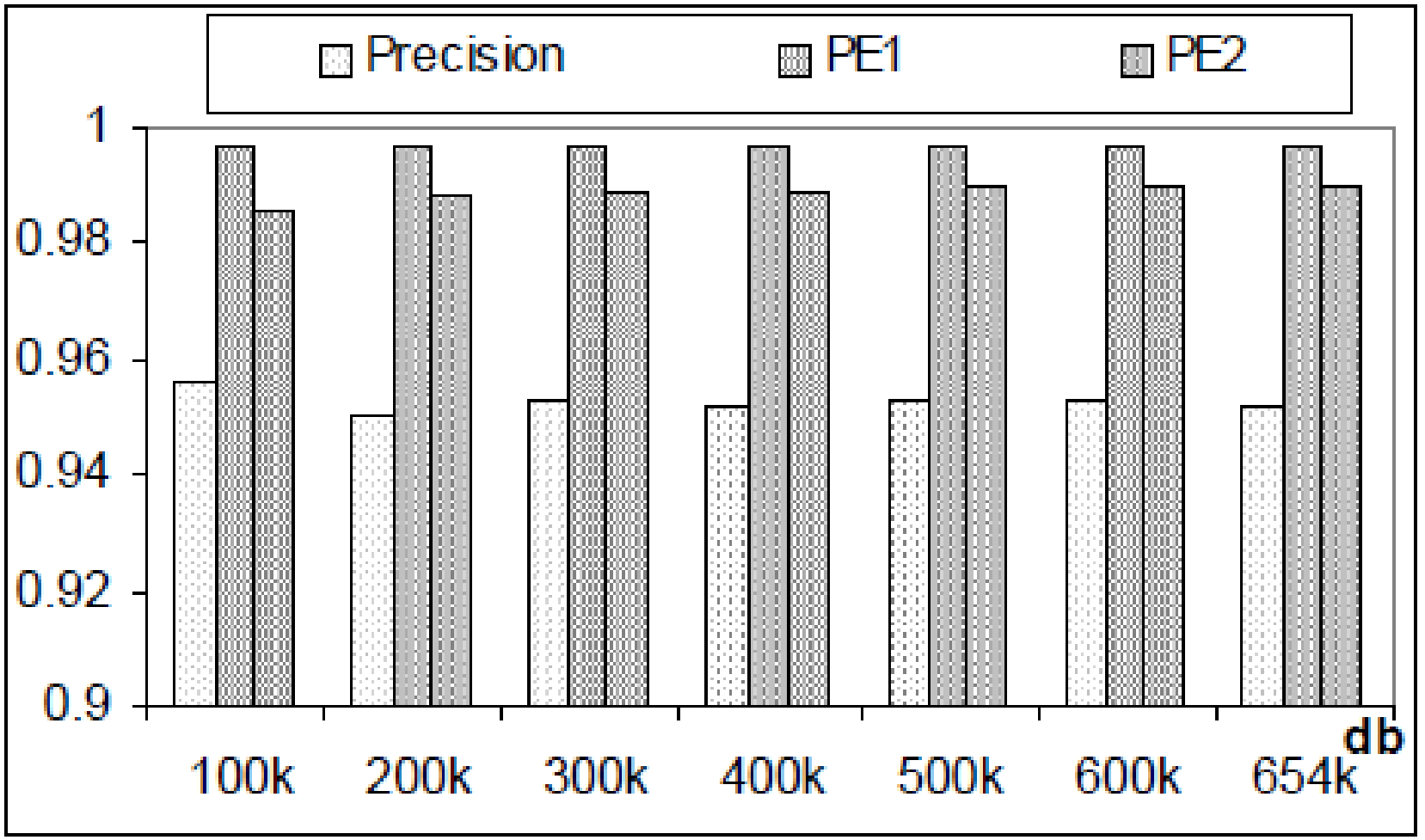
(b) Bulkload time of Dataset II: Human+Ecoli

Empirical results: Index file size



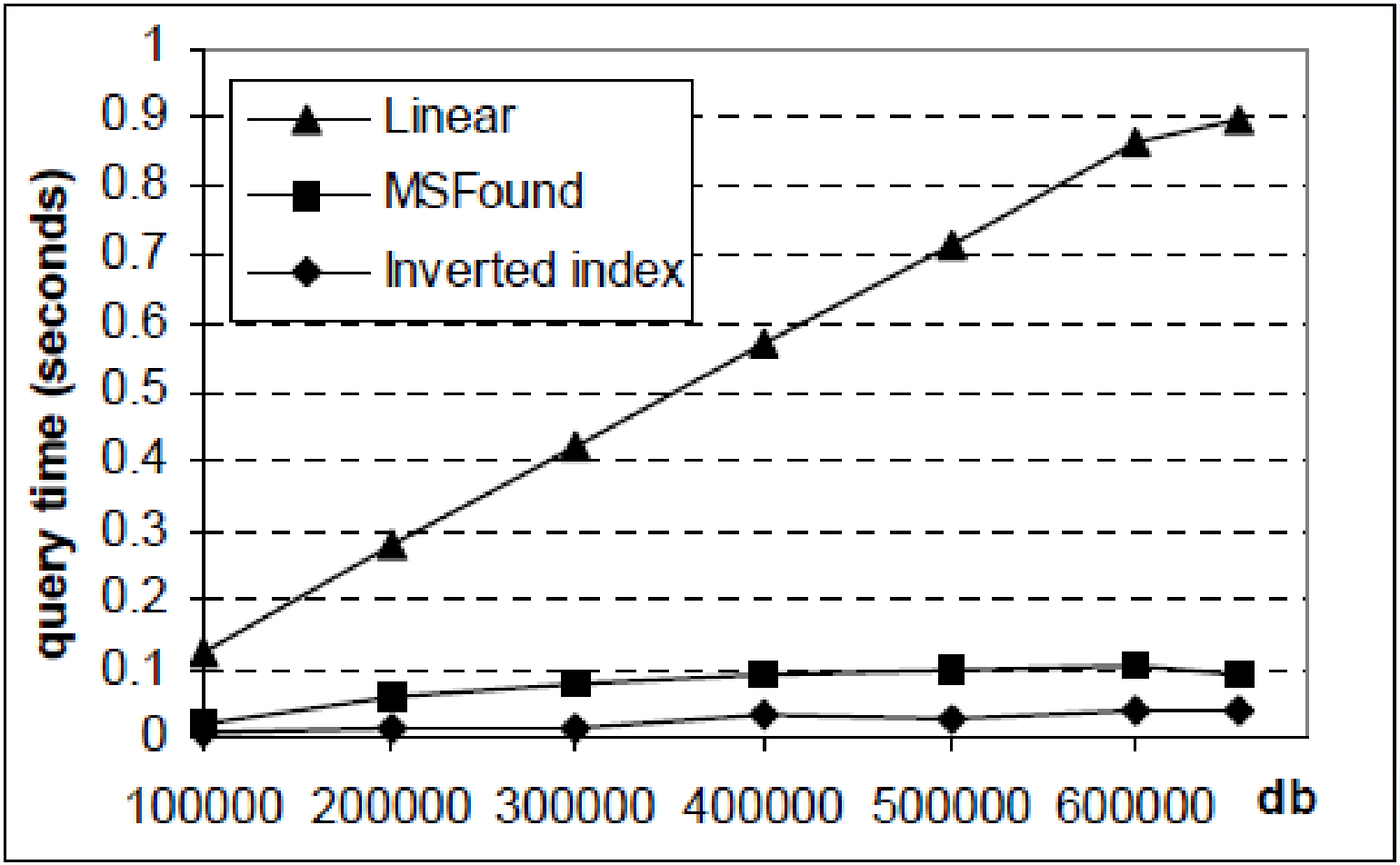
(b) Index file sizes of Dataset II: Human+Ecoli

Empirical results: Pruning Efficiency



(b) Query statistics of Dataset II: Human+Ecoli

Empirical results: Range queries



(b) Range query running time of Dataset II: Human+Ecoli

Outline

- Mass spectra
- Tandem cosine distance
- An inverted index method
- Empirical results
- Conclusions and Future work

Conclusions and Future Work

- How much is the cost to be general?
  - The inverted index has high pruning efficiency
  - The inverted index outperforms metric space method
  - Good scalability of metric space method
- Future work
  - Larger datasets
  - More distance functions, more data types
  - Biologically better distance functions

Thank you!