Indexing chemical fingerprints for efficient querying of molecular databases

Abhik Mondal

IIT Madras

May 5, 2015

Project Guide: Dr. Sayan Ranu

Overview

Motivation

- Fast database search is vital especially in drug discovery, where the aim is identifying chemical compounds with high similarity to known drugs.

Motivation

- Fast database search is vital especially in drug discovery, where the aim is identifying chemical compounds with high similarity to known drugs.
- When a candidate drug is discovered, millions of dollars are spent in laboratory for experiments and trials before it can see the light of day.

Definitions

- Indexing

Definitions

- Indexing
- Fingerprint

Definitions

- Indexing
- Fingerprint
- Querying

Problem Statement

Range Search Problem

Given a fingerprint, say *'f'*, a similarity measure *'sim'*, a threshold distance $'\theta'$ and a database of chemical compounds D, we find the subset $S \subset D$ of all fingerprints, such that:

$$S = \{g \mid g \in D, sim(f, g) < \theta\} \tag{1}$$

Some more concepts/definitions

- Tanimoto similarity

$$T_s(X, Y) = \frac{\sum_i X_i \wedge Y_i}{\sum_i X_i \vee Y_i} \tag{2}$$

Some more concepts/definitions

- Tanimoto similarity

$$T_s(X, Y) = \frac{\sum_i X_i \wedge Y_i}{\sum_i X_i \vee Y_i} \tag{2}$$

- Min-Max similarity

$$M_s(X, Y) = \frac{\sum_i min(X_i, Y_i)}{\sum_i max(X_i, Y_i)} \tag{3}$$

Some more concepts/definitions

- Tanimoto similarity

$$T_s(X, Y) = \frac{\sum_i X_i \wedge Y_i}{\sum_i X_i \vee Y_i} \tag{2}$$

- Min-Max similarity

$$M_s(X, Y) = \frac{\sum_i min(X_i, Y_i)}{\sum_i max(X_i, Y_i)} \tag{3}$$

- Distance measure? Metric?

Contribution

- Proposed an indexing technique based on the M-tree data structure.

Contribution

- Proposed an indexing technique based on the M-tree data structure.
- Proposed an inverted indexing technique.

Contribution

- Proposed an indexing technique based on the M-tree data structure.
- Proposed an inverted indexing technique.
- Explored range search techniques for the above.

Contribution

- Proposed an indexing technique based on the M-tree data structure.
- Proposed an inverted indexing technique.
- Explored range search techniques for the above.
- Extensions to non-binary fingerprints as well.

Contribution

- Proposed an indexing technique based on the M-tree data structure.
- Proposed an inverted indexing technique.
- Explored range search techniques for the above.
- Extensions to non-binary fingerprints as well.
- Tested our method on 2 real world datasets by comparing with the "Bit bound technique".

M-tree

- Routing objects
- Covering radius

M-tree

- Routing objects
- Covering radius

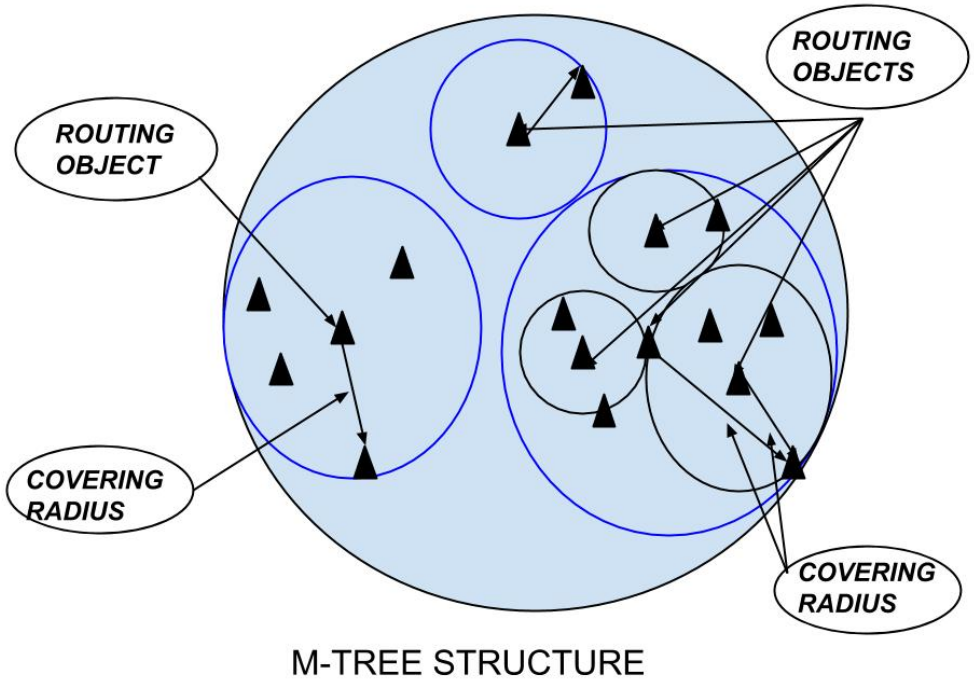


Figure : M-tree Structure Overview

Indexing approach ...

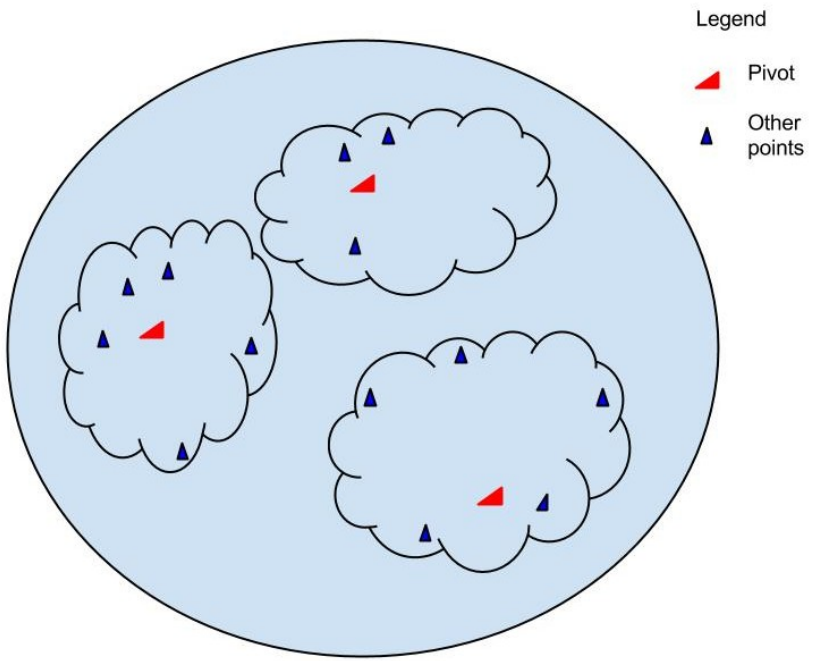
- Select pivots? Number?

Indexing approach ...

- Select pivots? Number?
- Assign each point to a pivot.

Indexing approach ...

- Select pivots? Number?
- Assign each point to a pivot.

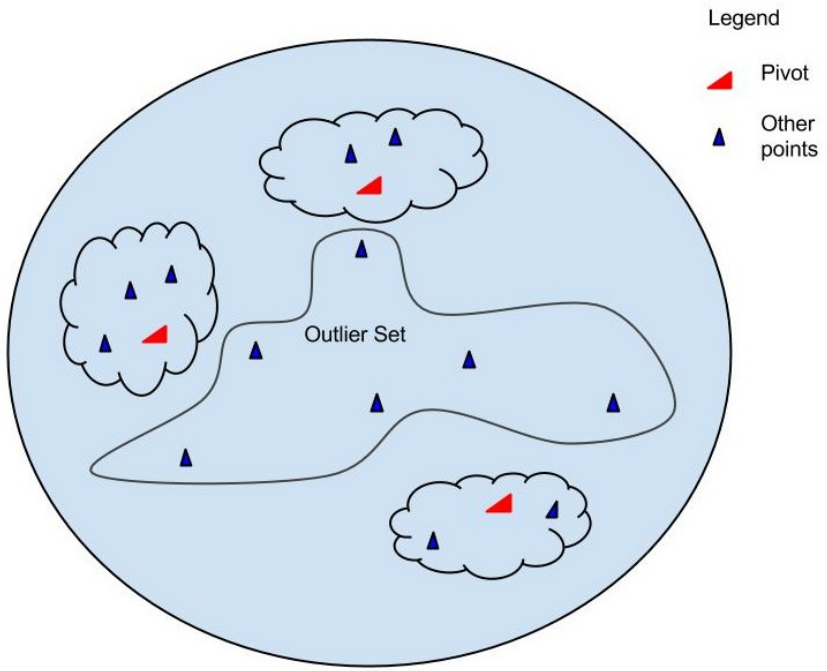


Indexing approach ...

- Choose outliers ?

Indexing approach ...

- Choose outliers ?



Indexing approach

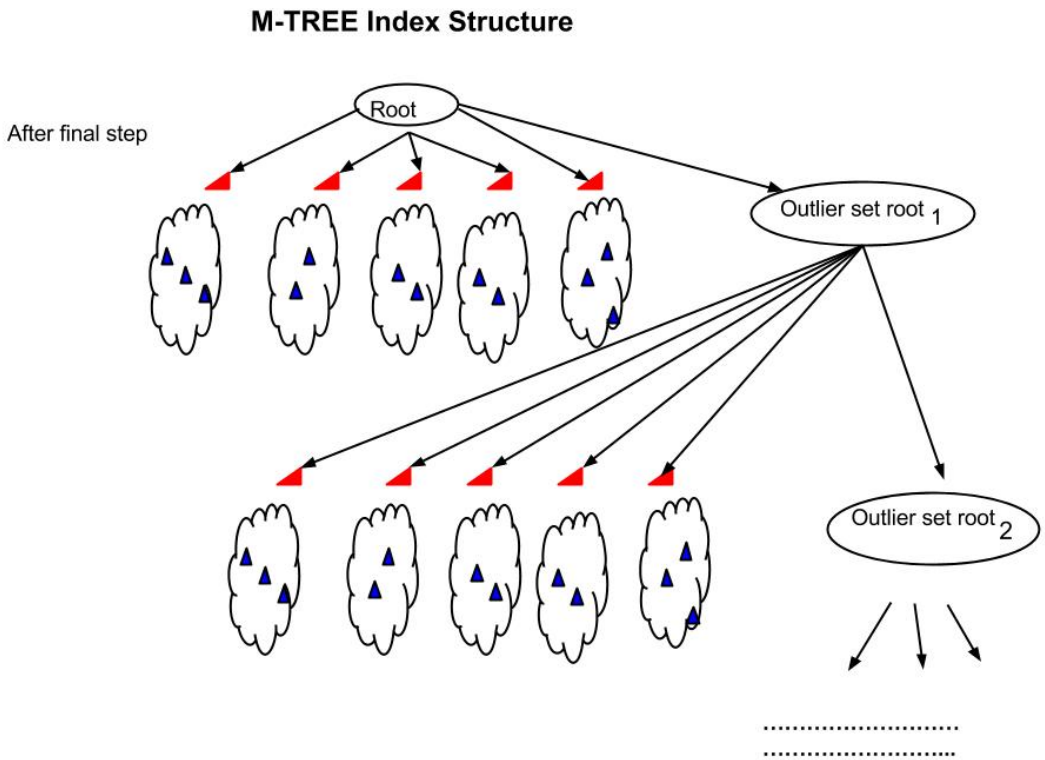
- Repeat procedure on outlier set.

Indexing approach

- Repeat procedure on outlier set.

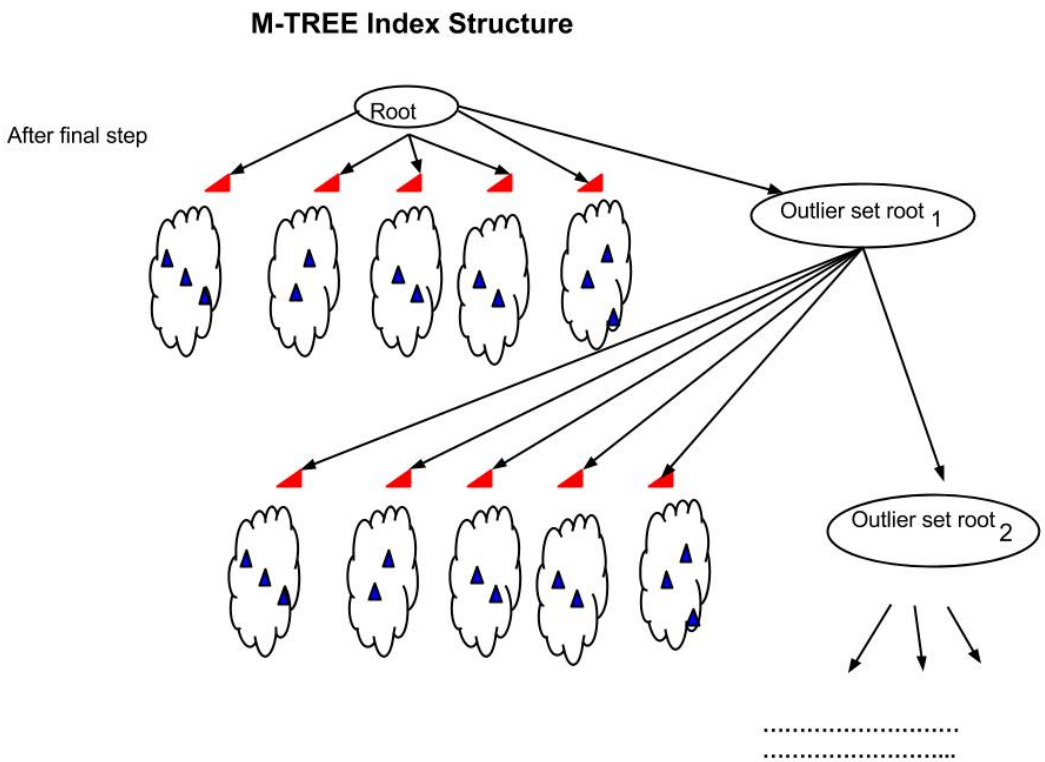
Indexing approach

- Repeat procedure on outlier set.



Indexing approach

- Repeat procedure on outlier set.



- Termination?

Range Search

- Start from the root
- Apply triangle inequality bounds.
- Covering radius of pivot p_i being r_i, the maximum distance of any node in S_i (subtree rooted at p_i) to the query q will be $dist(q, p_i) + r_i$.
- Use threshold t, to include the whole sub-tree.

Range Search ...

- Similarly the minimum distance of any node in S_i is $max(dist(q, p_i)$ - $r_i, 0)$.
- Prune?
- If not, then go to the children of p_i and repeat the process till we reach leaf.

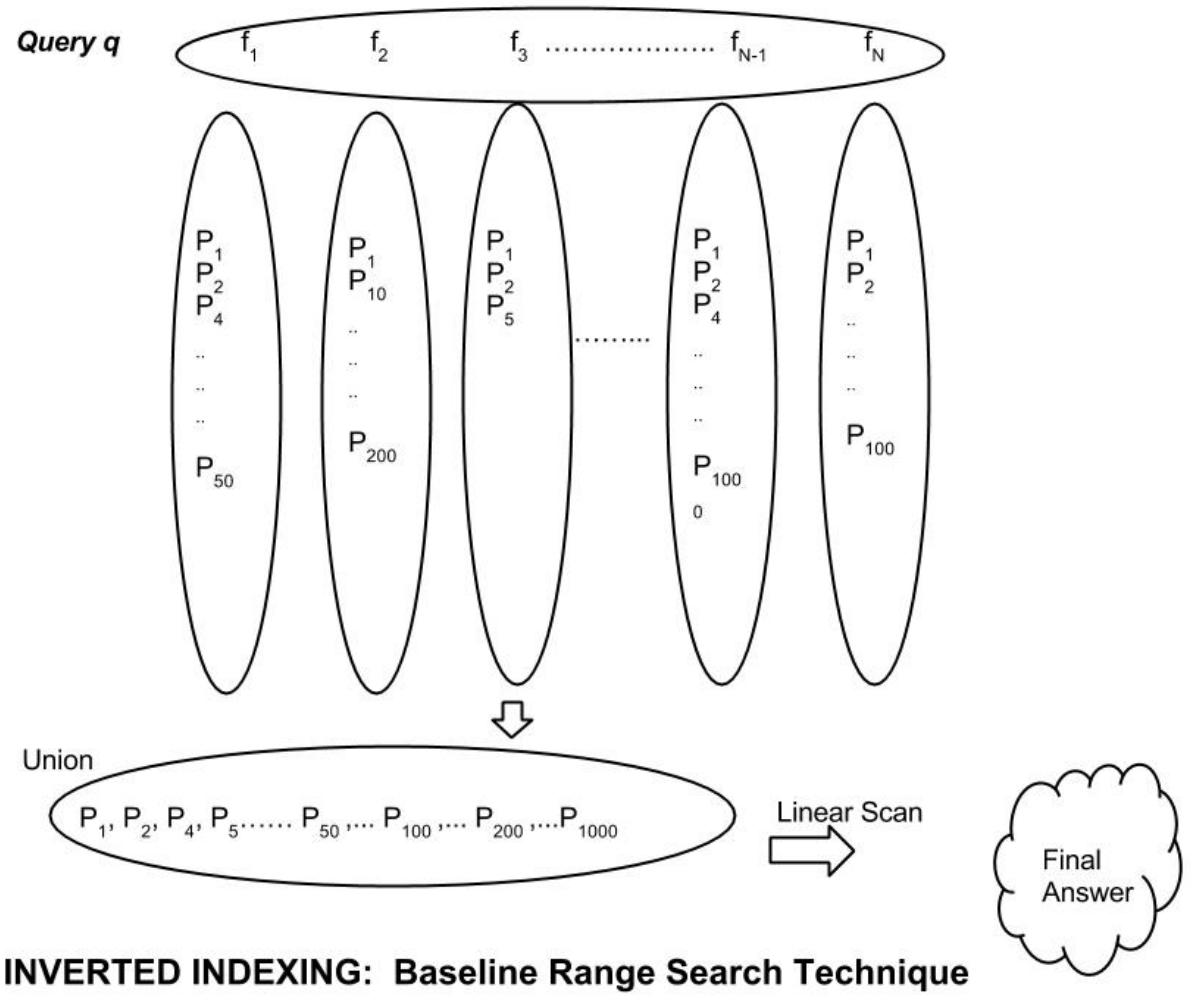
Inverted Index

- High dimensionality and sparsity of chemical data are an impediment to our indexing process.
- Use of inverted index motivated by its use in text mining.

Indexing process

- Index on features.
- Pre-processing?

Range Search



INVERTED INDEXING: Baseline Range Search Technique

Pruning features for binary fingerprints

- Consider f_i, if the following were to hold:

$$\frac{1}{N_q - 1 + V_i} < 1 - t \tag{4}$$

  we can prune the feature f_i
- Note: Here t is the threshold distance, V_i is the minimum number of features present among all points having the feature f_i

Proposed bounding theorem

Theorem

For a case of binary fingerprints, given a query q and a threshold t, consider the feature set $F = f_1, f_2, ...f_M$. If P is the set of points from the database, which has atleast one of the features f_k ($f_k \in F$) set to 1, then, we can prune all such points p_j of P from being present in the candidate range search set for query compound q if p_j does not have any other common feature other than in the set F, and it follows the following bound.

$$\frac{M}{N_q - 1 + \min\limits_{i \in (1,M)} (V_i)} < 1 - t \tag{5}$$

where M is the number of features in the set F,
N_q is the number of features in query q,
V_i is the minimum number of features present among all points having the feature f_i.

Greedy Technique

- Sort the features based on popularity

Greedy Technique

- Sort the features based on popularity
- Hence if till the i^{th} feature is considered, if j features (call it set R) have been pruned till now, we can prune the i^{th} feature as well if the following holds (as described in Equation 5)

$$\frac{j+1}{N_q - 1 + min(V_i, \rho)} < 1 - t \tag{6}$$

where ρ is the minimum number of features present in any point containing atleast one of the features pruned until now i.e $\rho = \min_{k \in R} V_k$

Extension to non-binary fingerprints

Theorem

Prune i^{th} feature if:

$$\frac{min(j_i, W_i)}{S_q - W_i - k_i + l_i + max(k_i, V_i)} < 1 - t \tag{7}$$

Here j_i is the maximum feature value taken for the feature f_i,
W_i is the i^{th} feature value of query q,
S_q is the sum magnitude of the feature values of the query q,
k_i is the is the minimum feature value taken for the feature f_i,
l_i is the minimum sum of feature values for any point containing the feature f_i ,
t is the threshold similarity.

Datasets

- PubChem Dataset (264016 compounds, 785985 features)
- DUD Dataset (128374 compounds, 32198 features)

Statistics on data

Table : Data Analysis: Statistics of the data-set PubChem-n

Number of data points	264016
Number of unique features is	785985
Maximum number of features in a data point is	1903
Minimum number of features in a data point is	7
Average number of features in a data point is	270.602966
Maximum number of data points with a feature is	259110
Minimum number of data points with a feature is	1
Average number of data points with a feature is	90
Maximum value of a feature	1870
Minimum value of a feature	1
Average value of a feature	1.142210
Maximum number of heavy-hitters	144
Minimum number of heavy-hitters	1
Average number of heavy-hitters	44.5

M-tree based index analysis

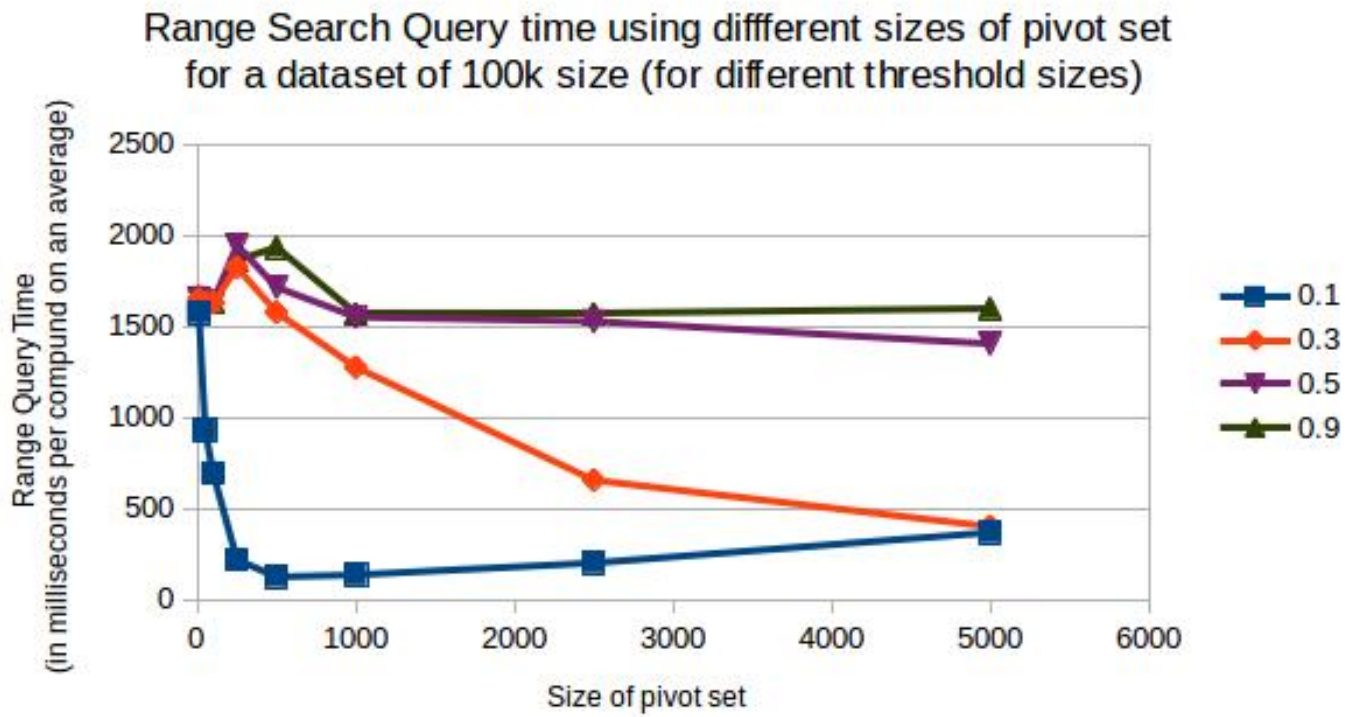
- Indexing time per compound on average increases linearly with data-set size as well as with size of pivot-set
- Outlier base limit size has no significant effect.

M-tree based index analysis...

- Range Search varies for different pivot-set sizes.

M-tree based index analysis...

- Range Search varies for different pivot-set sizes.



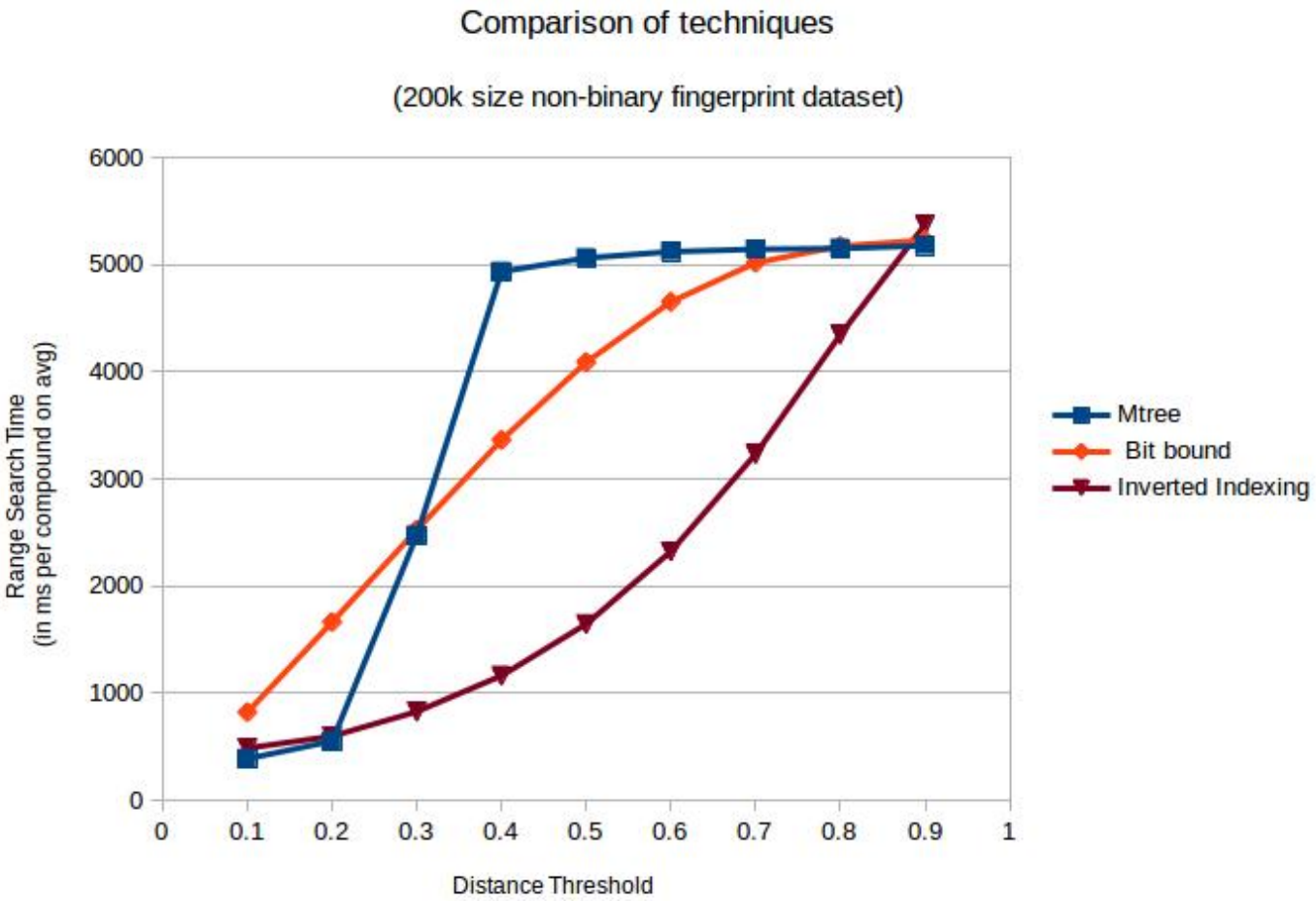
Inverted index analysis

- Indexing time per compound on average is constant. Does not change with data-set size.

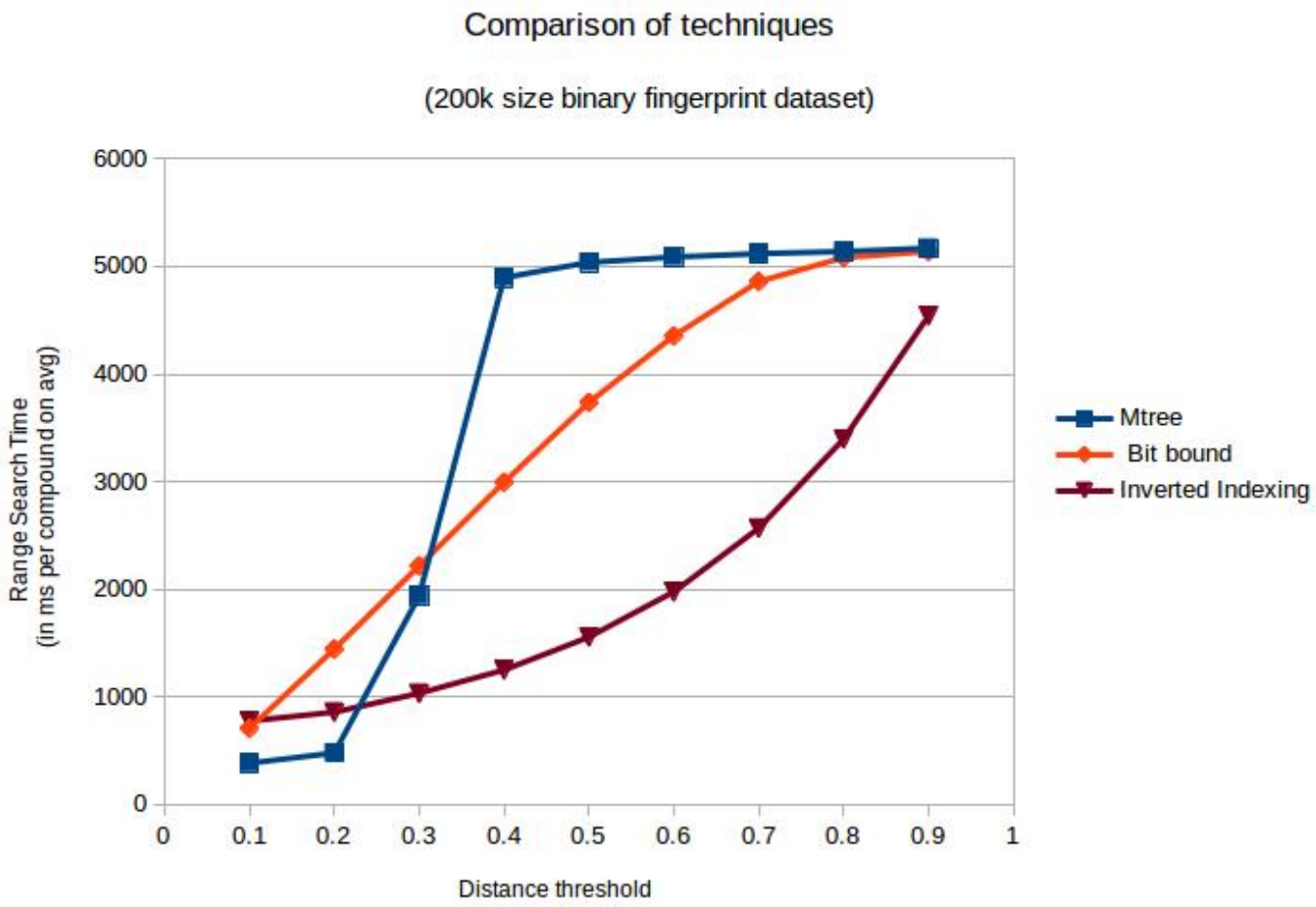
Inverted index analysis

- Indexing time per compound on average is constant. Does not change with data-set size.
- Pruning upto 50-100 features on average for low threshold distances.

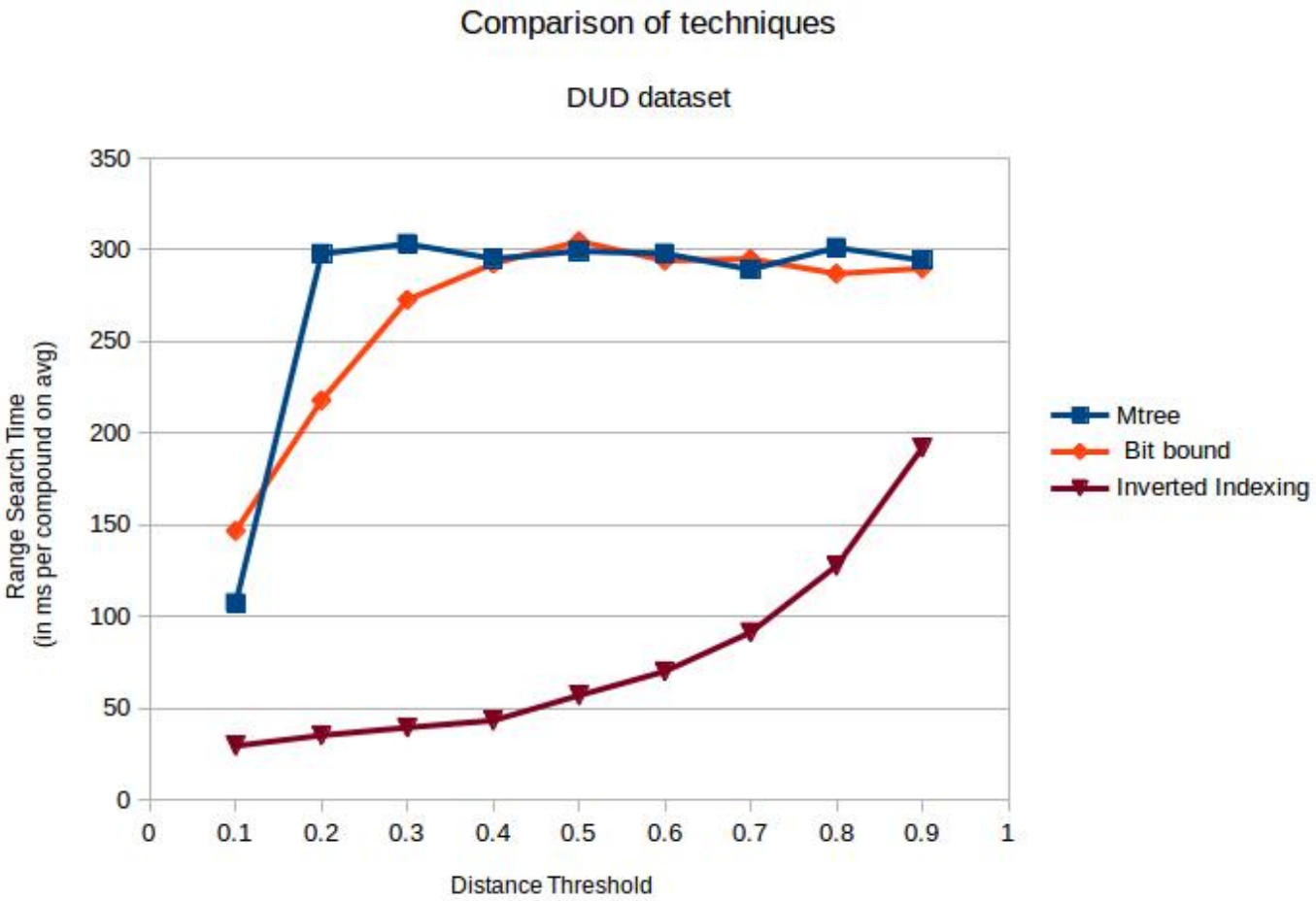
Comparison


Comparison of techniques
(200k size non-binary fingerprint dataset)
Range Search Time
(in ms per compound on avg)
6000
5000
4000
3000
2000
1000
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Distance Threshold
Mtree
Bit bound
Inverted Indexing

Comparison ...


Comparison of techniques
(200k size binary fingerprint dataset)
Range Search Time
(in ms per compound on avg)
6000
5000
4000
3000
2000
1000
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Distance threshold
Mtree
Bit bound
Inverted Indexing

Comparison ...


Comparison of techniques
DUD dataset
Range Search Time
(in ms per compound on avg)
350
300
250
200
150
100
50
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Distance Threshold
Mtree
Bit bound
Inverted Indexing

Conclusion

- Proposed an M-tree based index approach which exploited the metric property.

Conclusion

- Proposed an M-tree based index approach which exploited the metric property.
- Proposed a novel inverted indexing technique which relied on pruning of features.

Conclusion

- Proposed an M-tree based index approach which exploited the metric property.
- Proposed a novel inverted indexing technique which relied on pruning of features.
- Showed the effectiveness of our techniques through comprehensive analysis on 2 real world datasets.