

2<sup>nd</sup> International Workshop on  
Similarity Search and Applications (SISAP 2009)

Prague, Czech Republic

---

# Optimal Pivots to Minimize the Index Size for Metric Access Methods

L. G. Ares, N. R. Brisaboa, M. F. Esteller

O. Pedreira, A. S. Places

Database Laboratory, Universidade da Coruña, Spain



# Outline



1. Motivation
2. Previous work
3. Analysis of pivot effectiveness
4. Minimum-space pivot-based index
5. Experimental evaluation
6. Conclusions

We use methods for similarity search because...

- a) Comparing two objects can be costly  
E.g. DNA sequences with edit distance  
  
and also...
- b) We search in large collections of objects  
E.g. Content generated by users from the Web  
Images, videos, user profiles, ...

# Motivation



Although the complexity is measured as the number of comparisons...

Total search time =

time for comparisons +

extra CPU time (index processing) +

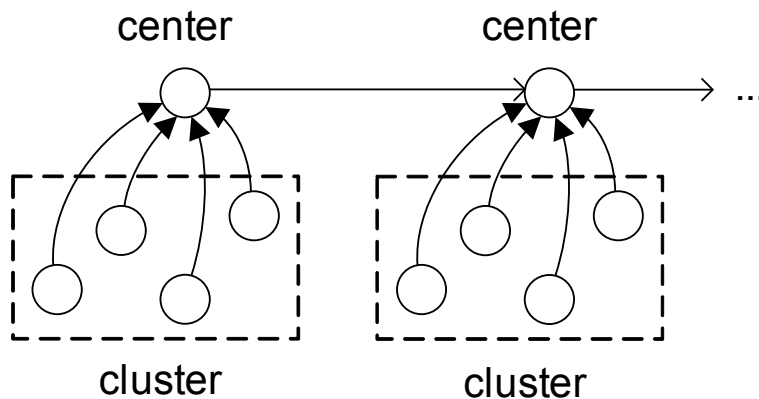
I/O time

Depend on the size of the index

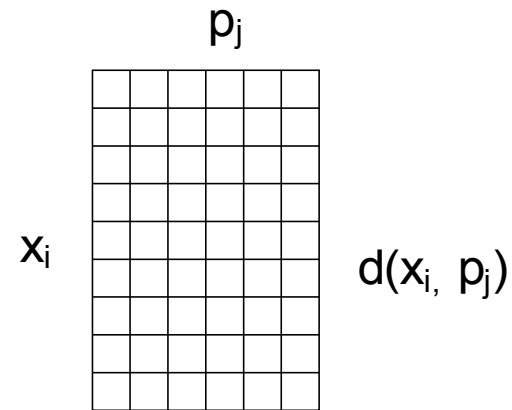
# Motivation



## Clustering-based methods



## Pivot-based methods



Number of comparisons: smaller in pivot-based methods

Space complexity:  $O(n)$  vs.  $O(nk)$

Space can be a problem for pivot indexes in large collections

# Motivation



Reduce the space requirements of pivot-based methods for situations in which the problem is the size of the collection more than the cost of a comparison.

# Outline



1. Motivation
2. Previous work
3. Analysis of pivot effectiveness
4. Minimum-space pivot-based index
5. Experimental evaluation
6. Conclusions

# Previous work



Three approaches for reducing the space of pivot methods

- a) Range coarsening
- b) Bucket coarsening
- c) Scope coarsening

The space is reduced at the cost of more comparisons.

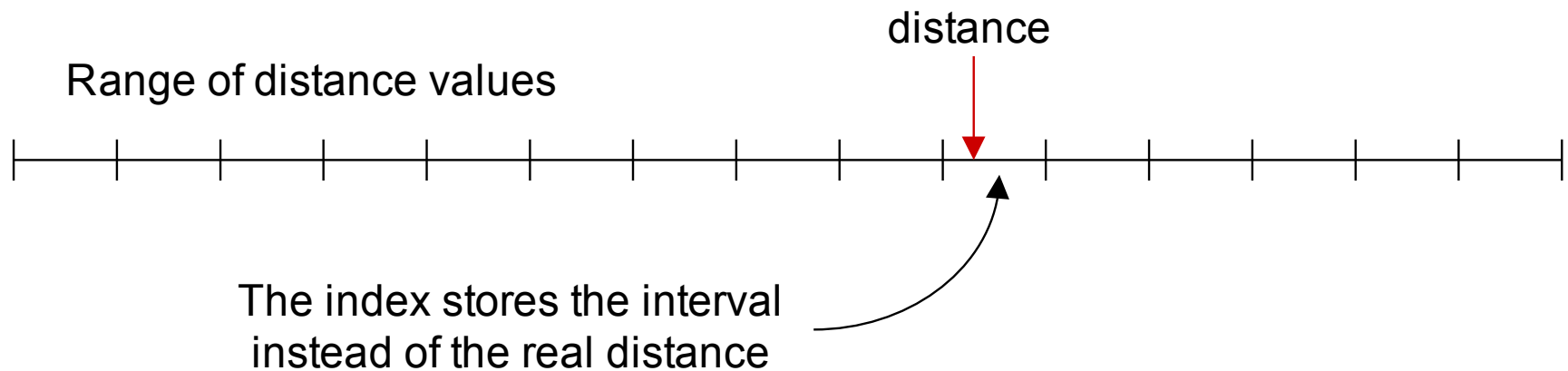


# Previous work

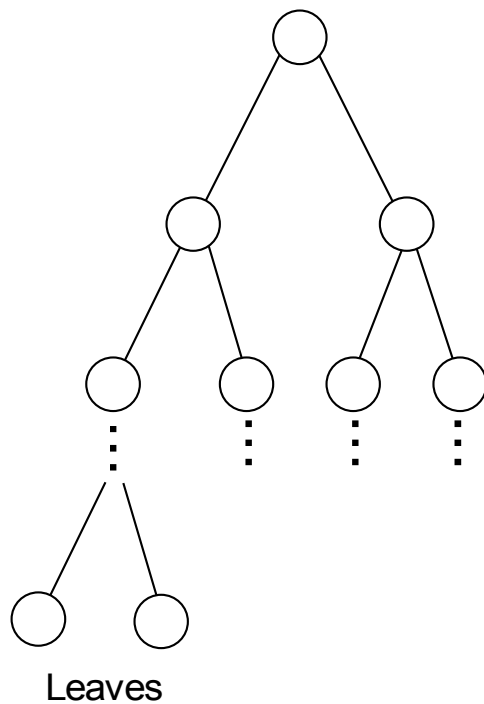


Range coarsening: The distances from pivots to objects are stored with less precision.

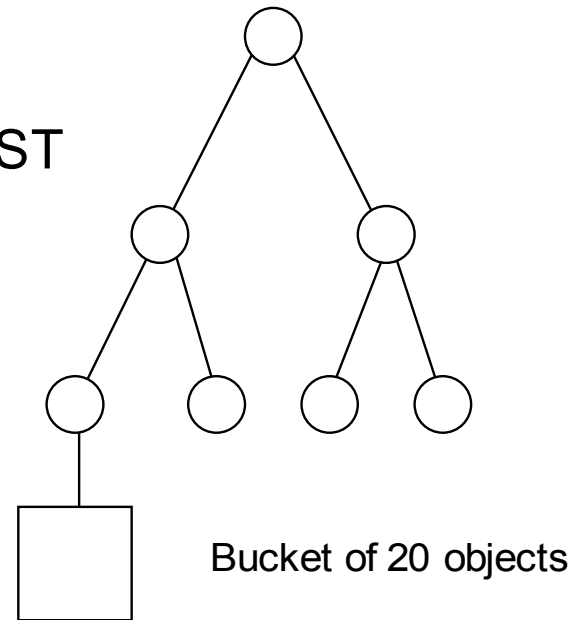
VPT, MVPT, FQA, BAESA



Bucket coarsening: for tree-like structures, stop indexing when bucket has a given size



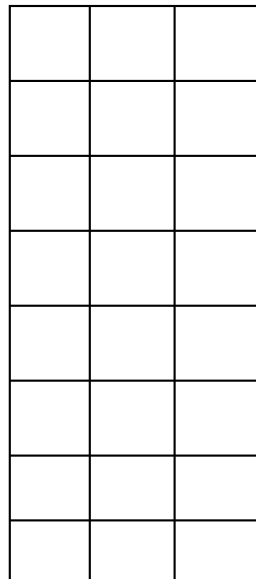
Example with BST



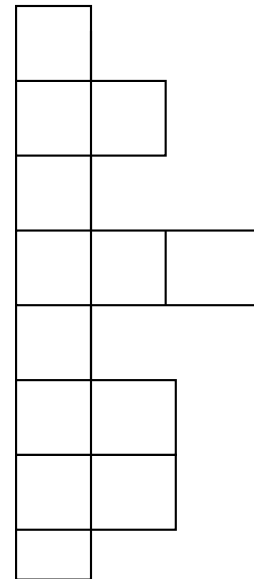
# Previous work



Scope coarsening: reduce the scope of the pivots by storing only distances from each object to its most promising pivots.



VS.



# Previous work



## Sparse Spatial Selection

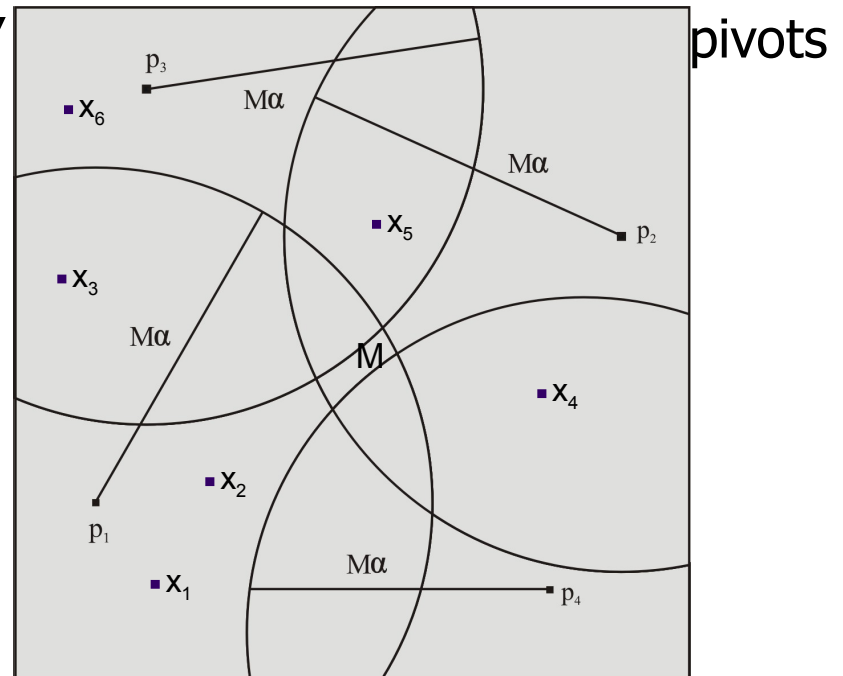
When an object is inserted, it is selected as a new pivot if it is far away enough from the current pivots

The object is considered “far-away” if greater than  $M\alpha$

M maximum distance

$0 < \alpha < 1$

$\alpha = 0.5$



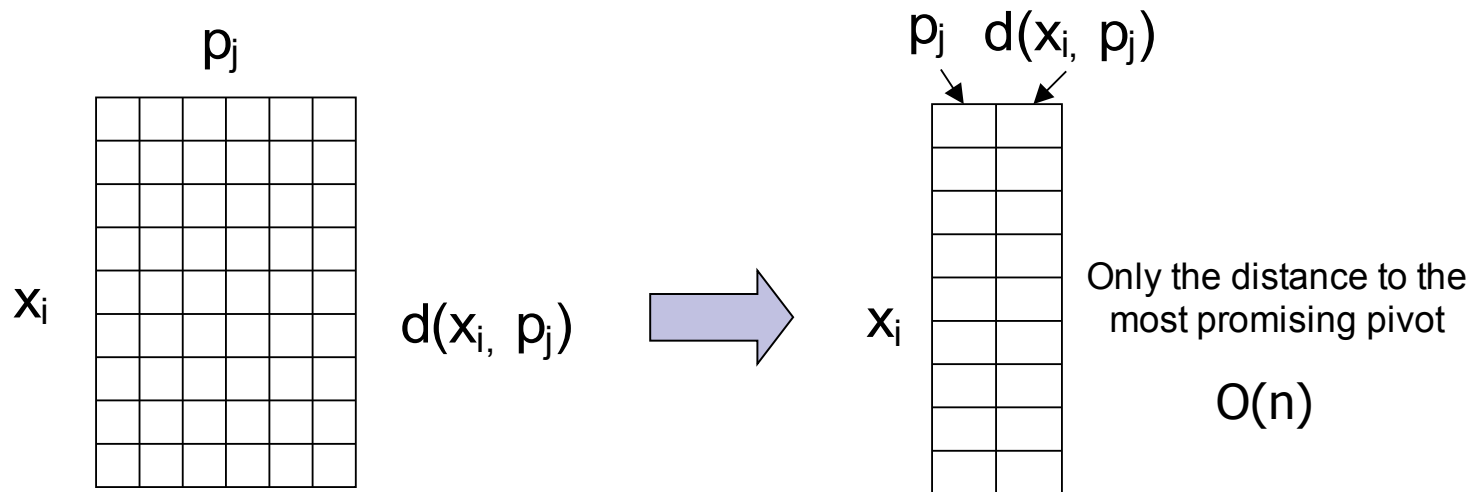
# Outline



1. Motivation
2. Previous work
3. Analysis of pivot effectiveness
4. Minimum-space pivot-based index
5. Experimental evaluation
6. Conclusions

# Hypothesis of this work

- a) Reduce as much as possible the space of pivot-based indexes.



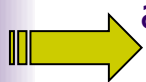
- b) Analyze how to obtain the most promising pivot for each object in the DB.

# Analysis of pivot effectiveness



## Some questions about pivots...

- a) How can we find good pivots for a given object? ✓
- [Celik, 2002] showed that the near and far pivots for an object are the most promising ones.



- a) How can we ensure that near and far pivots will be available for each object?
- b) Which pivot among those near and far do we choose as the most promising ones?

# Analysis of pivot effectiveness



- b) How can we ensure that near and far pivots will be available for each object?

A random selection does not guarantee it.

SSS obtains a set of pivots well distributed in the space.  
[Brisaboa, 2006]

**Hypothesis:** SSS is an effective way for obtaining near and far pivots for each object.

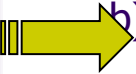
Since we do not store all distances, we can select a larger set of pivots to cover the space appropriately.



# Analysis of pivot effectiveness



## Some questions about pivots...

- a) How can we find good pivots for a given object? ✓
  - [Celik, 2002] showed that the near and far pivots for an object are the most promising ones.
- a) How can we ensure that near and far pivots will be available for each object? ✓
-  b) Which pivot among those near and far do we choose as the most promising ones?

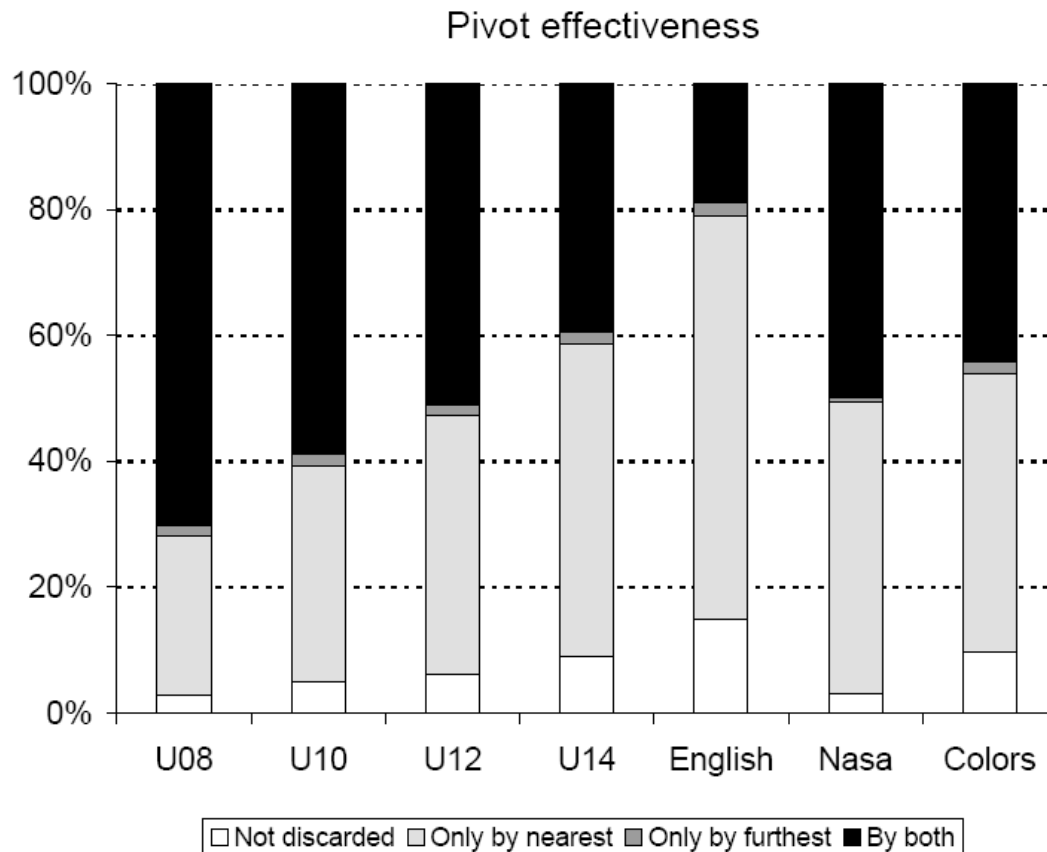
# Analysis of pivot effectiveness

Coll.	Random			SSS (optimal $\alpha$ )		
	Piv.	Space	Eval. $d$	Piv.	Space	Eval. $d$
UV08	85	29.1828	211.78	53	18.1963	141.43
UV10	190	65.2321	468.23	176	60.4255	367.14
UV12	460	157.9303	998.13	250	85.8317	645.08
UV14	1000	343.3266	2077.44	491	168.5734	1381.64
English	200	27.5696	443.85	212	45.0553	354.89
Nasa	77	10.6143	276.34	55	7.4438	168.62

Coll.	SSS ( $\alpha = 0.25$ , 4 dist.)			SSS ( $\alpha = 0.25$ , 2 dist.)		
	Piv.	Space	Eval. $d$	Piv.	Space	Eval. $d$
UV08	494	2.7485	1231.66	494	1.3752	3094.13
UV10	1461	2.7522	3011.61	1461	1.3789	5891.64
UV12	4303	2.7630	6803.09	4303	1.3897	9739.49
UV14	10000	2.7848	14229.20	10000	1.4115	18160.91
English	3100	1.9089	7931.30	3100	0.9604	12373.45
Nasa	871	1.1061	1308.28	871	0.5547	1958.34

# Analysis of pivot effectiveness

- c) Which pivot among those near and far do we choose as the most promising ones?



# Analysis of pivot effectiveness



- c) Which pivot among those near and far do we choose as the most promising ones?

Collection	$\mu$	$\sigma$	Nearest		Furthest		Both	
			$zd_{nearest}$	$zd_{furthest}$	$zd_{nearest}$	$zd_{furthest}$	$zd_{nearest}$	$zd_{furthest}$
UV08	1.5086	0.2452	-4.3989	0.9845	-4.1949	1.3923	-4.3989	1.4331
UV10	1.4032	0.2456	-3.7182	2.2671	-3.5147	2.6743	-3.7182	2.6743
UV12	1.2652	0.2450	-3.0008	3.5298	-2.7151	3.9788	-3.0416	3.9380
UV14	1.1244	0.2469	-2.3669	4.6804	-1.9214	5.0855	-2.4480	5.0450
English	8.3176	2.0260	-2.3335	4.6113	-1.9040	5.1591	-2.2742	4.9617
Nasa	1.2342	0.3424	-2.2611	3.1419	-2.0859	2.9083	-2.1735	2.7623

# Outline



1. Motivation
2. Previous work
3. Analysis of pivot effectiveness
4. Minimum-space pivot-based index
5. Experimental evaluation
6. Conclusions

# Minimum-Space Pivot-based Index



Minimum-Space Pivot-based Index combines:

1) Scope coarsening

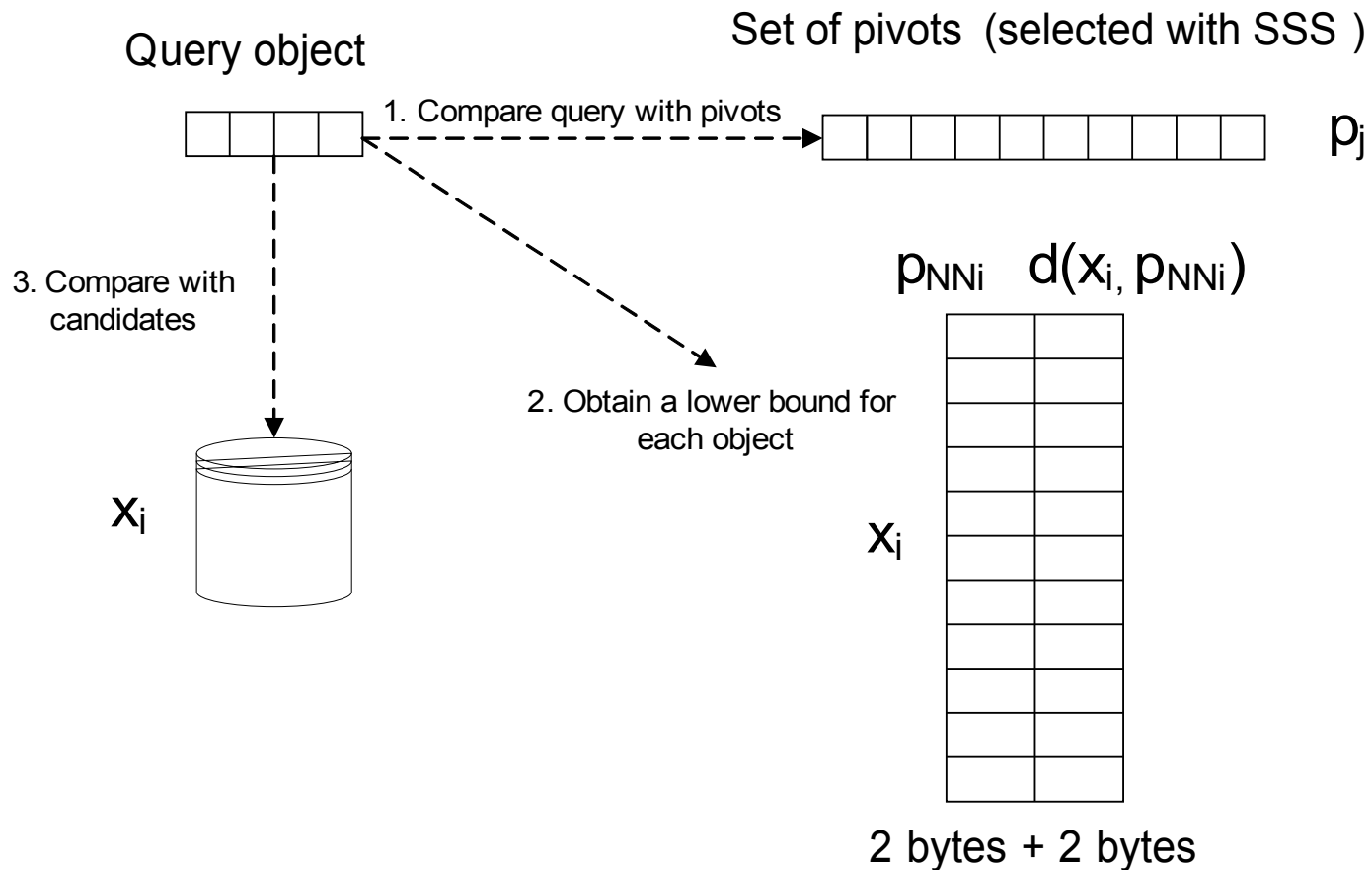
For each object, store only the distance to its most promising pivot and its identifier.

We take the nearest pivot as the most promising.

2) Range coarsening

Store both the identifier and the distance in 4 bytes.

# Minimum-Space Pivot-based Index



# Outline



1. Motivation
2. Previous work
3. Analysis of pivot effectiveness
4. Minimum-space pivot-based index
5. Experimental evaluation
6. Conclusions



# Experimental evaluation of the method



Experimental evaluation of the method:

- 1) Loss of efficiency due to loss of precision
- 2) Comparison with other methods

# Experimental evaluation



## 1) Loss of efficiency due to loss of precision

Collection	UPI (4 bytes)		UPI (2 bytes)	
	Space	Eval. $d$	Space	Eval. $d$
UV08	0.4311	3094.13	0.2594	3095.72
UV10	0.4348	5891.64	0.2631	5893.84
UV12	0.4456	9739.49	0.2740	9741.91
UV14	0.4673	18160.91	0.2957	18164.03
English	0.3083	12373.45	0.1897	12373.45
Nasa	0.1757	1958.34	0.1068	1958.72

# Experimental evaluation



## 2) Comparison with other methods

Collection	SSS (optimal $\alpha$ )		KVP (k=2)		UPI		List of Clusters	
	Space	Eval. $d$	Space	Eval. $d$	Space	Eval. $d$	Space	Eval. $d$
UV08	18.1963	141.43	1.3752	8724.62	0.2594	3095.72	0.3643	6139.21
UV10	60.4255	367.14	1.3789	13063.63	0.2631	5893.84	0.3710	11264.98
UV12	85.8317	645.08	1.3897	18955.94	0.2740	9741.91	0.3710	17273.55
UV14	168.5734	1381.64	1.4115	28502.26	0.2957	18164.03	0.3710	28253.04
English	45.0553	354.89	0.9604	18305.43	0.1897	8872.37	0.2651	7885.79
Nasa	7.4438	168.62	0.5547	2676.93	0.1068	1958.72	0.1427	2027.08

# Outline



1. Motivation
2. Previous work
3. Analysis of pivot effectiveness
4. Minimum-space pivot-based index
5. Experimental evaluation
6. Conclusions

# Conclusions



- a) Space requirements can be a problem pivot-based indexes when working with large collections.
- b) New method proposed.
- c) It is possible to reduce the space of a pivot based index and get results better than with clusters.

## Future work...

- a) Complete the analysis on the search complexity in large collections.
- b) Test new criteria for obtaining the most promising pivot.

Thanks for your attention!

Questions?