2010 Jan. 16@ ALENEX2010 Austin Texas

Conjunctive Filter: Breaking the Entropy Barrier

Daisuke Okanohara[*1, *2]

Yuichi Yoshida [*1*3]

[*1] Preferred Infrastructure Inc.

[*2] Dept. of Computer Science, Univ. of Tokyo

[*3] School of Informatics, Kyoto University

Breaking the entropy barrier

- We want to store data structures using the space less than its entropy
  - The result should contain some errors
- Bloom Filter [Bloom 1970]
  - Store a set of keys, and given a key it answers whether the key exists or not.
  - it always report that a key exists if the key indeed exists, but false positive are allowed (one-sided error).

Problem :
Associate a key with a set of values

- $f : X \rightarrow 2^V$
  - Map from a key to a set of values
  - X : The universe of keys, $|X| = n$
  - V : The universe of values, $|V| = m$
- Example: Posting List
  - X : words, V: hit document IDs

word1	2	7	15	32	33	37	
word2	1	7	14	16	18	30	31
word3	1	2	3	15	37		

Entropy barrier storing a map

- For t integers in $[1...m]$, we can store them in $t \log_2 m$ bits
- If we reduce the space further, we must admit too many errors
  - Storing 1 integer in $[1...2^{20}]$ requires 20 bit.
  - If we use only 5 bit, with one-sided error, since we can distinguish 2^5 cases, each case should contain $2^{20}/2^5=32768$ integers

Too many false positives !

Conjunctive queries

- Can't we reduce the space in the case of a set of values ?

⇒We can !

- If we only consider *conjunctive queries* (or *AND* query), the space can be reduced further with small errors.

$$f^{\cap}(X) := \bigcap_{i=1}^{k} f(x_i)$$

```
Example:
f(x1)={1,2,3,5} f(x2)={2,3,5} f(x3)={2,3,4}
X=(x1,x2,x3)
```

$f(x_1)=\{1,2,3,5\}$ $f(x_2)=\{2,3,5\}$ $f(x_3)=\{2,3,4\}$

$X=(x_1,x_2,x_3)$ $f^{\cap}(X) = \{2,3\}$

(ε, k)-encoding

- Let s be a binary string expressing a map f.
- A binary string s is called (ε, k)-encodes a map if for k-conjunctive query, it holds

(1) One-sided error

$$v \in f(X) \text{ then } v \in s(X)$$

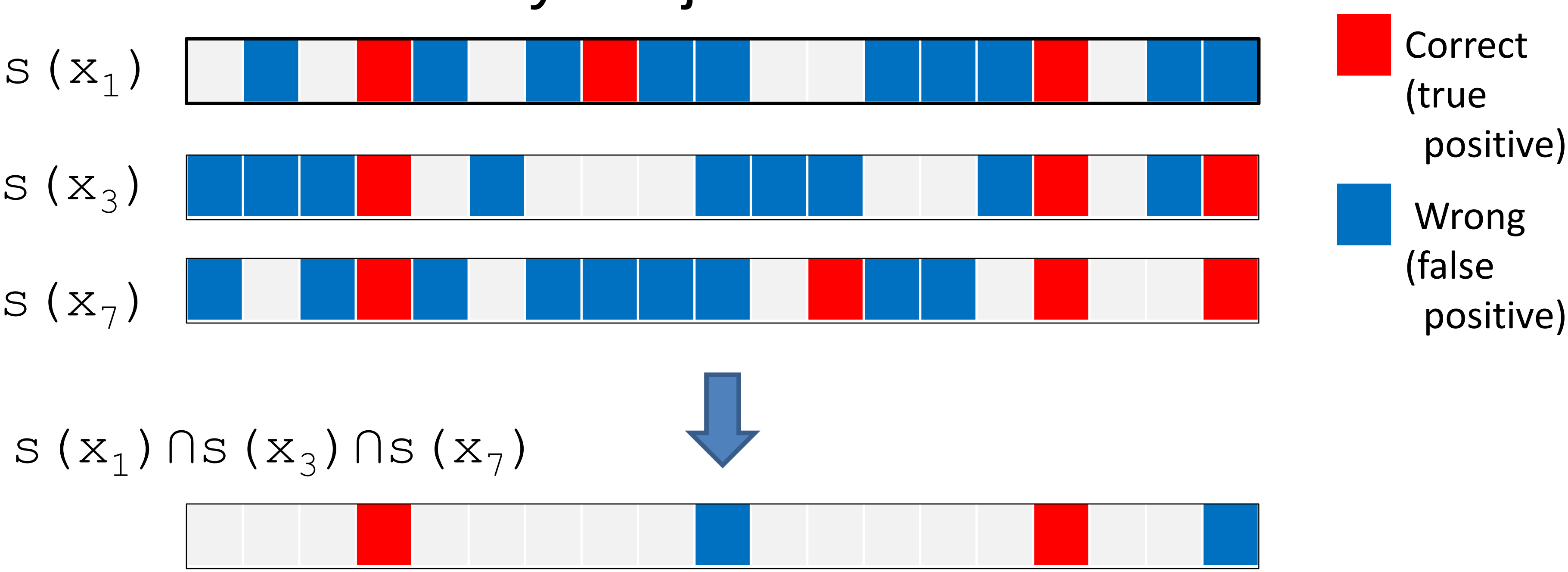
(2) False positive error rate

$$|s(X) - f^{\cup}(X)| \,/\, |V - f^{\cup}(X)| < \varepsilon.$$

- $f^{\cup}(X)$ is the result of a disjunctive query

Key Idea

- Although it is difficult to reduce the size for original queries, it is not for conj. queries.
  - False positives are randomly distributed, and filtered out by conjunction

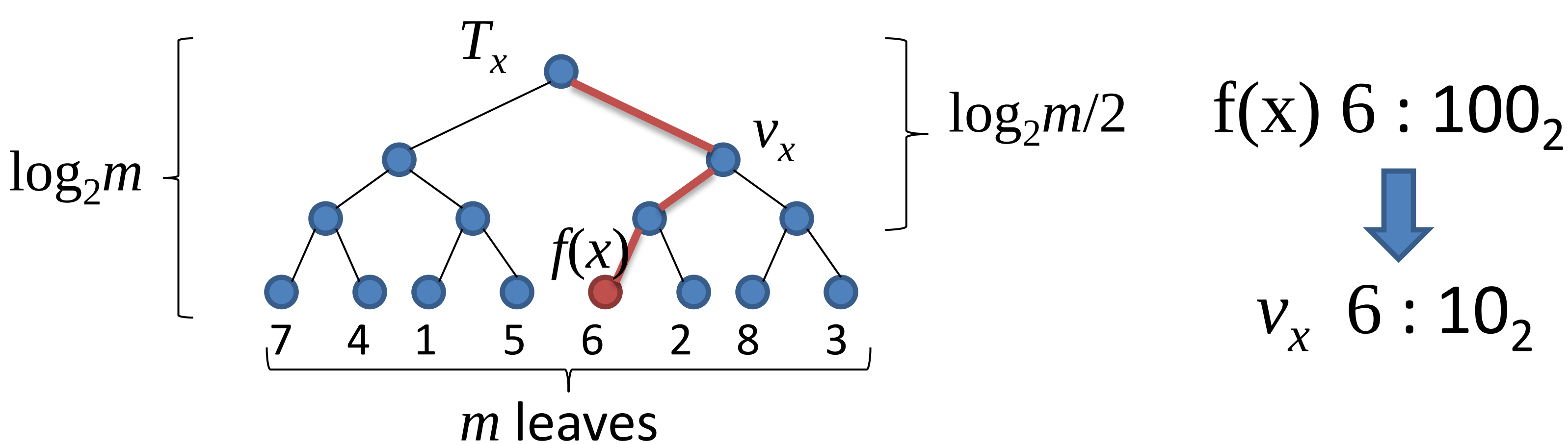


Conjunctive Filter (1/4)

- Conjunctive filter (ε, k)-encodes a map in a space-efficient manner.
- For simplicity, we assume here:
  - Only one value is associated with each key.
  - $k=2$
- Removing these assumptions is easy.

Conjunctive Filter (2/4)

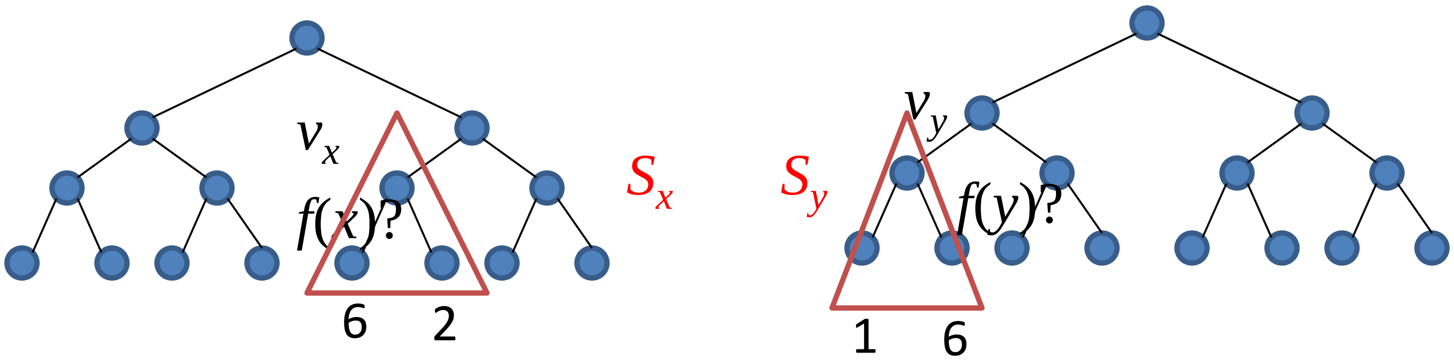
- To save $(x, f(x))$:
- Make random binary tree T_x with m leaves
  - x is used as a seed of a hash function.
  - Each leaf corresponds to a value.



- Save v_x using the first $\log_2 m/2$ bits of $f(x)$

Conjunctive Filter (3/4)

- 2-conjunctive query on x and y:
- Construct T_x and T_y and go down to v_x and v_y:
- Let S_x and S_y be the sub-tree rooted at v_x and v_y



- Take the intersection of values associated to leaves of S_x and S_y.

$\{6, 2\} \cap \{1, 6\} = \{6\}$

Conjunctive Filter (4/4)

- Theorem: $E[|S_x \cap S_y|]=O(1)$.
- Fact:
  - S_x and S_y are randomly distributed.
  - $|S_x|=|S_y|=m^{1/2}$
  - For an element v not in $f(x)$ nor $f(y)$, the probability of v appearing in $S_x \cap S_y$ is $O(1/m)$

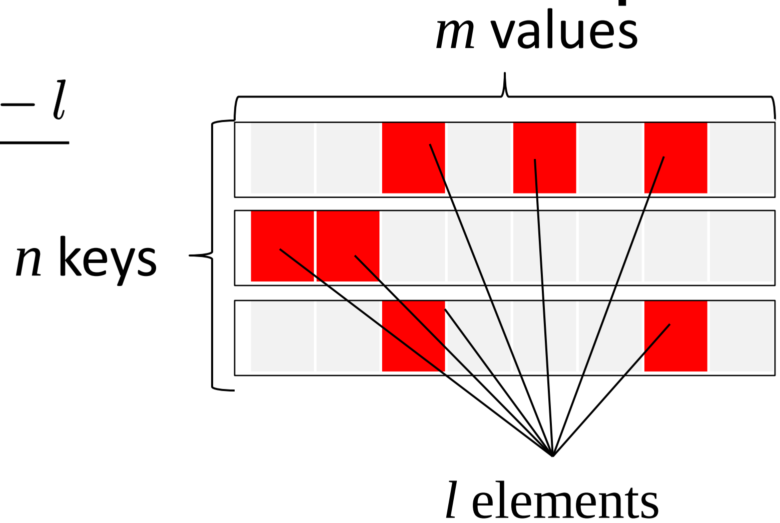
Conjunctive Filter
General case

- For general k, we get $(1/m^{1/2}, k)$-encoding consuming $(\log_2 m)/k$ bits per element.
  - 1/*k* of the original size.
  - The lower bound to store $(1/m^{1/2}, k)$-encoding is $(\log_2 m)/2k$ bits (following slides)

Lower Bound on Space of Map (k=1)

- #bits per element for naïve map is:

$$1/l \log_2 \binom{nm}{l} \sim \log_2 \frac{nm - l}{l}$$



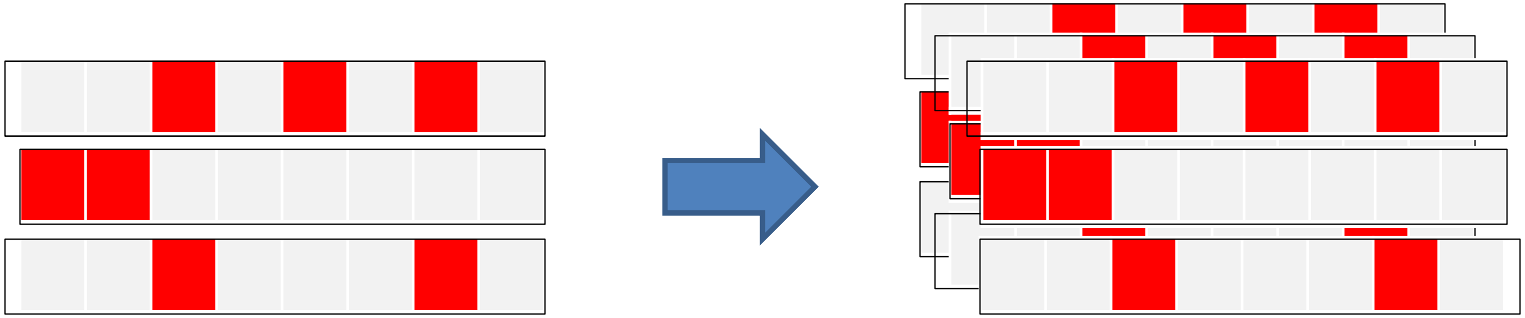
- #bits per element for an (ε,1)-encoding of a map is: $\log_2 \frac{nm - l}{l + \epsilon(nm - l)}$
- Proof idea: count how many maps one bit-string can encode.

Lower Bound on Space of Map (k>1)

- #bits per element for an (ε,k)-encoding of a map is:

$$\frac{1}{k}\log_2 \frac{nm-l}{l+\epsilon(nm-l)}$$

Proof Idea: Duplicate each row k times.

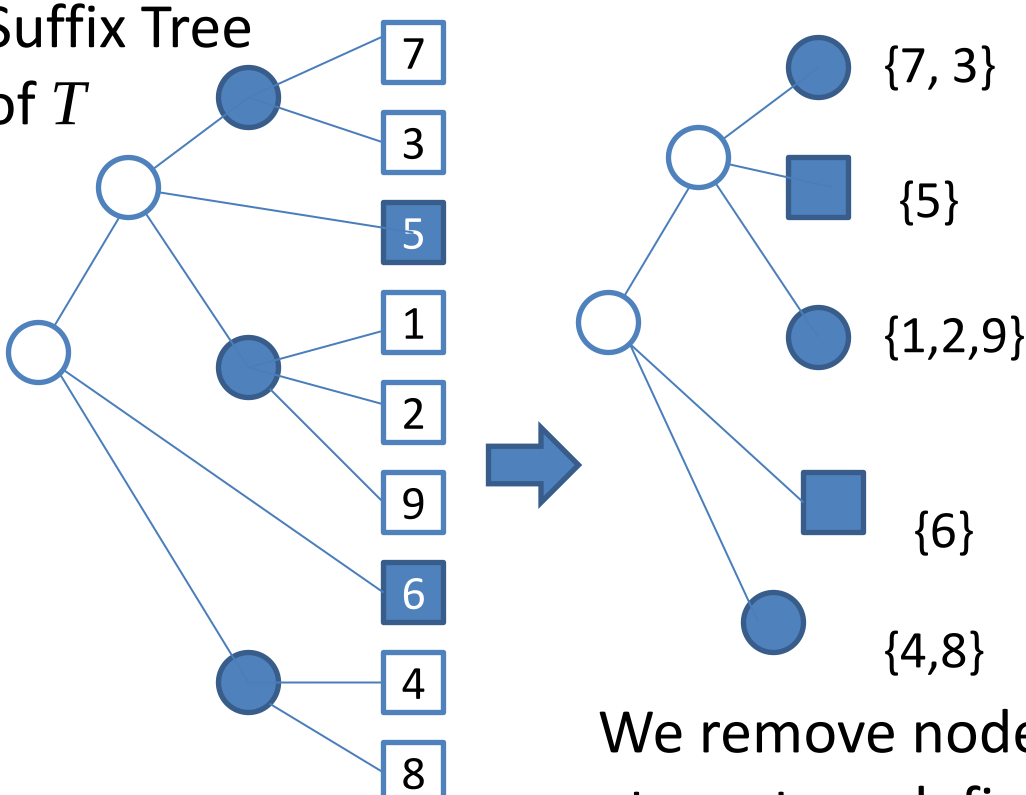


- If we can perform k-conjunctive query on the duplicated map, we can restore the original map.
- Use the previous bound.

Application : full-text search with long queries

- Problem: Given a query q, return the position list of q in the target text T

Suffix Tree of T



`q=abracadabra`

decompose

`q'={abr,aca, dab,ra}`

Conjunctive query on q'

We remove nodes so that each node has at most predefined # positions.

Experiment Setting

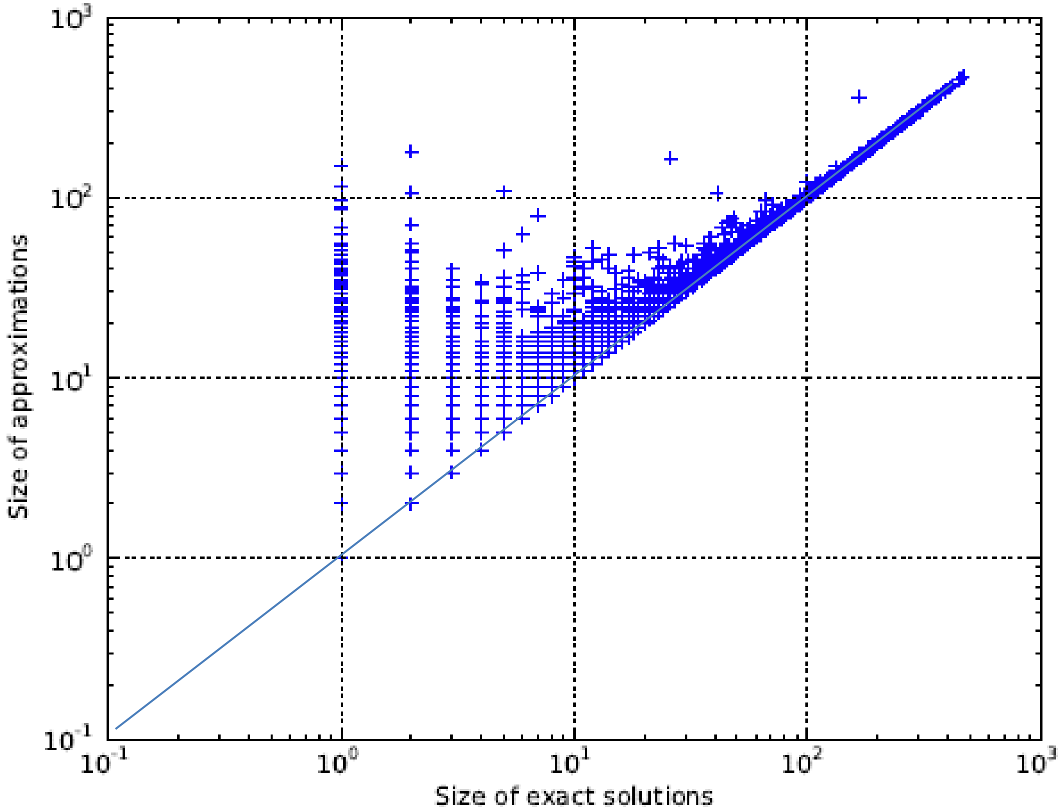
- Data: IMDB data set, actors section
  - 1103393 actors (keys, n)
  - 1791274 movies (value, m)
  - 6493558 relations (who acts in which movies)
- Query : a set of actors
- Result : a set of movies in which all actors act

Experiment 1
The size of conjunctive filters

	Conj. Filter Size (ratio to naive)	Lower bounds (ratio to naive)
Naive Encoding	1	
k=2	0.73	0.29
k=3	0.53	0.20
k=4	0.44	0.15

The size of conjunctive filter is smaller than raw, but it is still larger than the lower bound

Experiment 2 :
false positives in 2-conj. queries



$|s(x)|$

$|f(x)|$

Almost results have at most $m^{1/2}$ errors

The result of 3-conjunctive queries is similar

Conclusion and Future Work

- In k-conjunctive queries, we can break the entropy barrier
  - when # elements is linear to # keys
- A conjunctive filter achieves an (ε,k)-encode map using $1/k$ of the original size

- Reduce the working space
  - Consider the entropy of value distribution
- Reduce the time complexity
  - Fast intersection

Conjunctive/Disjunctive queries

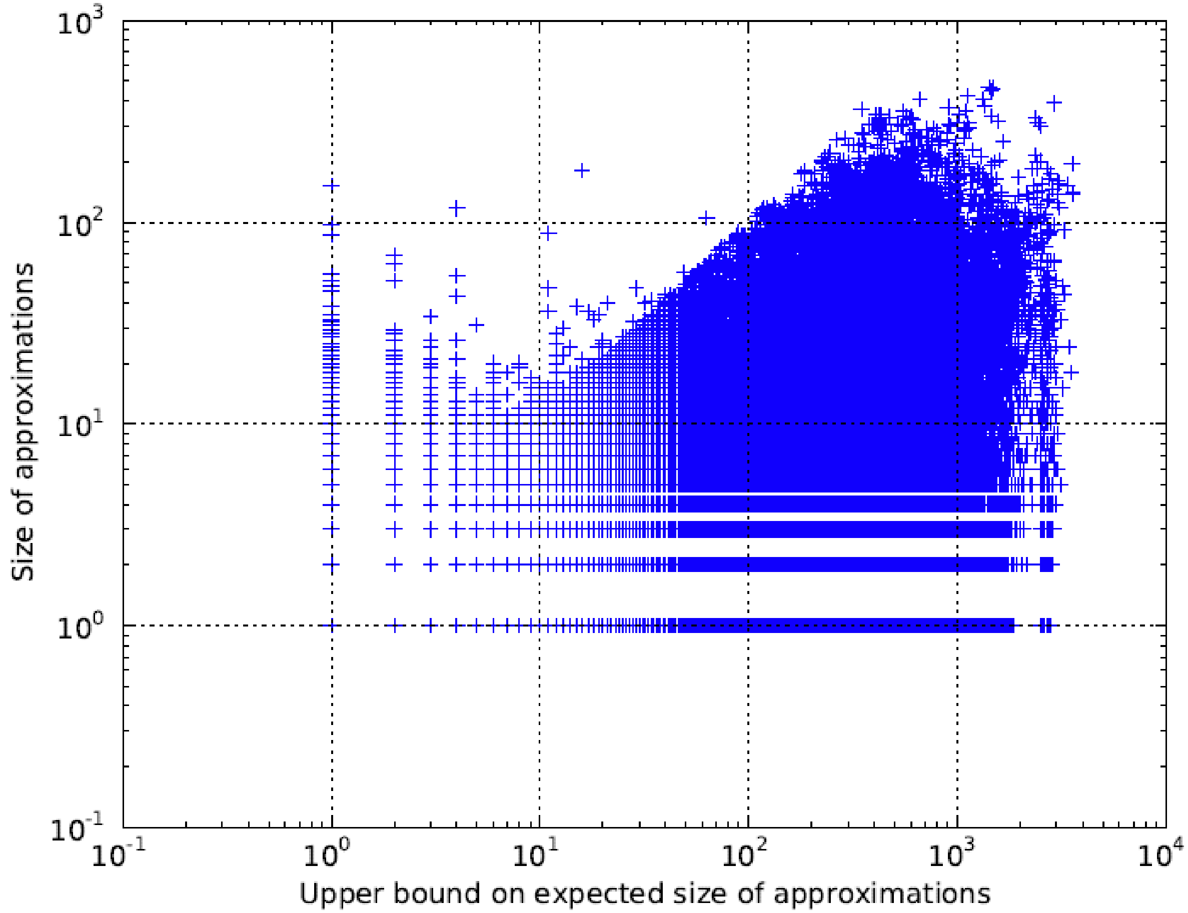
- Query : a set of keys $X = (x_1, x_2, ..., x_k)$
- Conjunctive query $f^{\cap}(X) := \bigcap_{i=1}^{k} f(x_i)$
- Disjunctive query $f^{\cup}(X) := \bigcup_{i=1}^{k} f(x_i)$

<u>Example</u>

- $\mathtt{f(x_1)=\{1,3,5\}}$ $\mathtt{f(x_2)=\{2,3,5\}}$ $\mathtt{f(x_3)=\{2,3\}}$
- $\mathtt{X=(x_1,x_2,x_3)}$

$$f^{\cap}(X) = \{3\} \qquad f^{\cup}(X) := \{1,2,3,5\}$$

Experiment 3 : comparison of $|f^{U}(X)|$ and $s(X)$ in 2-conj. query

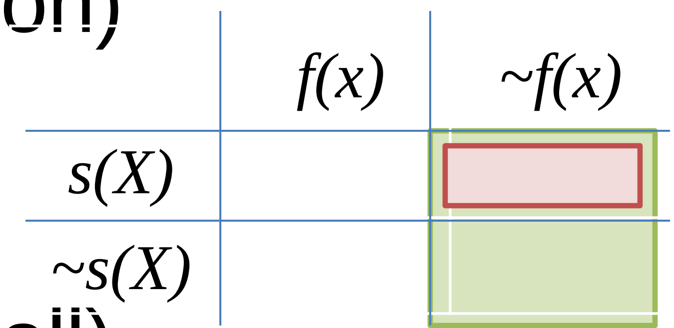


The result of 3-conjunctive queries is similar

Error Measures

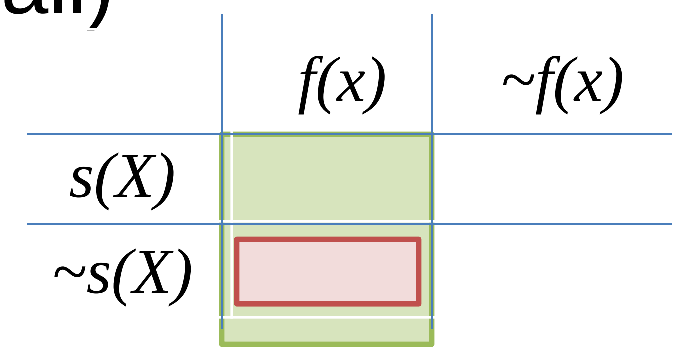
- Let *s* be a binary string of a map *f*.
- False Positives (1-Precision)

$$\epsilon_X^{+}(s) = \frac{|s(X) \setminus f(X)|}{|\mathcal{V} \setminus f(X)|}$$



- False Negatives (1 – Recall)

$$\epsilon_X^{-}(s) = \frac{|f(x) \setminus s(X)|}{|f(X)|}$$



- Union False Positives

$$\epsilon_X^{\cup}(s) = \frac{|s(X) \setminus f^{\cup}(X)|}{|\mathcal{V} \setminus f^{\cup}(X)|}$$