



Selectivity Estimation for Extraction Operators over Text Data

Daisy Zhe Wang , Long Wei

UC Berkeley

Yunyao Li, Frederick Reiss, Shivakumar Vaithyanathan

IBM Research - Almaden

Information Extraction (IE) over Text Data



- Email Search
- E-discovery
- News Mashup
- Data Redaction
- Enterprise Search
- Patent Search
-

Tomorrow, we will
meet Mark Scott,
Howard Smith and
....

System

[ICDE 2008]

Algebraic Information Extraction (IE) system

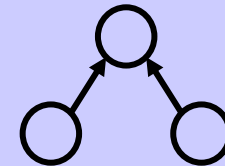
Declaratively specify the semantics extractions.

Choose an efficient execution plan that implements semantics

AQL Language

Optimizer

Operator
Runtime



AQL – Operators

- **Dictionary Extractor**

e.g., $\mathcal{E}_{\text{First}}$: **Dict(*first.dict*) Extractor**

- **Regular Expression Extractor**

e.g., $\mathcal{E}_{\text{Caps}}$: **Regex($\backslash s^*[A-Z]\backslash w+\backslash s^*$) Extractor**

- **Join with *FollowsTok(X, Y, min, max)* Condition**

e.g.,  **FollowTok(X, Y, 0, 0) : Join Two Adjacent Extractions**

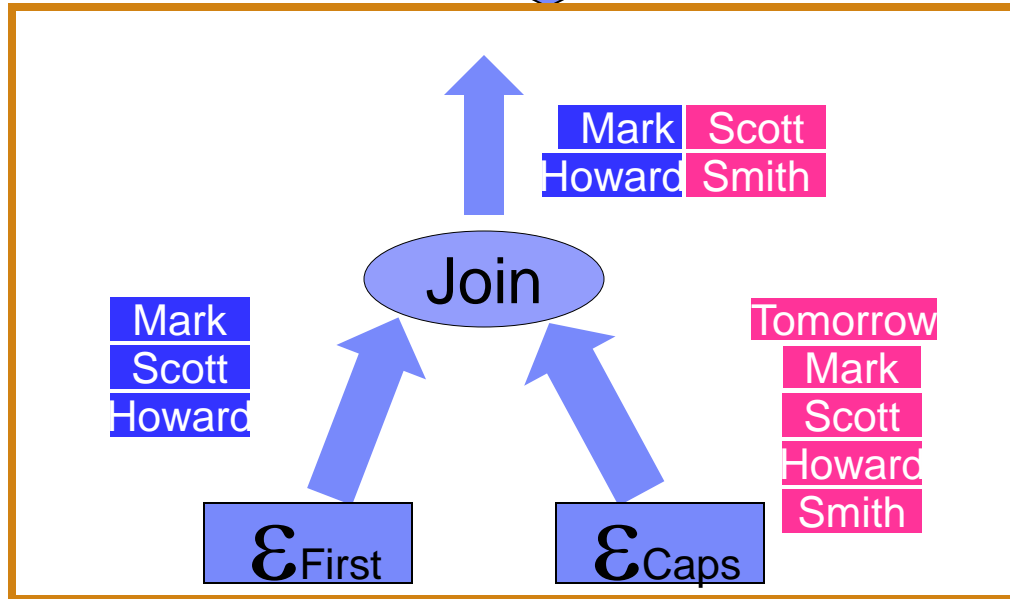
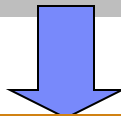
AQL – An Example



```

create view FirstCaps as
select CombineSpans(F.name,C.name) as name
from First F, Caps C
where FollowsTok(F.name, C.name, 0, 0);
  
```

Optimizer



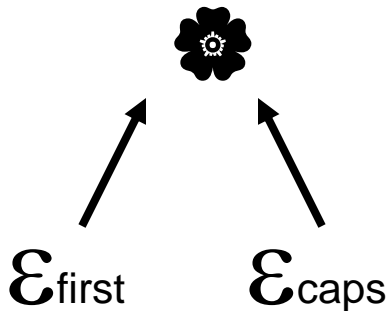
Operator graph

Tomorrow, we will meet Mark Scott, Howard Smith and

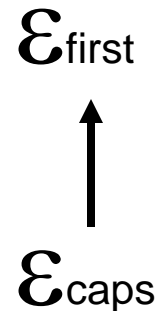
SystemT : Optimizer

```
create view FirstCaps as
select CombineSpans(F.name,C.name) as name
from First F, Caps C
where FollowsTok(F.name, C.number, 0, 0);
```

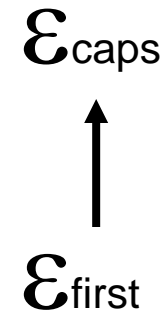
Plan A



Plan B



Plan C



The optimizer needs **Selectivity Estimation** for Extractors!

Outline

- **Introduction**
- **Basics**
 - Selectivity Estimation
 - Synopsis and Baseline Synopses
- **Dictionary Selectivity Estimation**
 - Bloom Filter Synopsis
- **Regex Selectivity Estimation**
 - Roll-up Synopsis
- **Conclusion**

Selectivity and Synopsis

- **Document Corpus:** D
- **Regular Expression** (re)
- **Ngram:** a sequence of N tokens (ngram)
- **Selectivity of an Extractor \mathcal{E} :**

$$sel(\mathcal{E}) = E [match (\mathcal{E}, D)]$$

- **Document Synopsis \emptyset** is a summary of D

\emptyset . build(D, k)

\emptyset . estCount(ngram) / \emptyset . estCount(re)

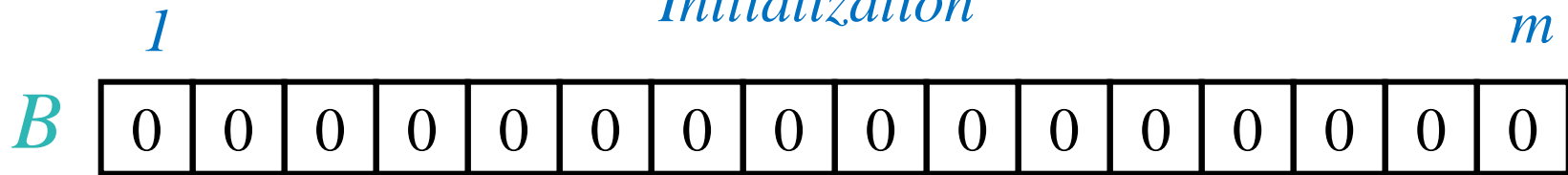
Baseline Synopses

- **Top-k N-gram Synopsis (TopkNgram)**
- **Random Sample Synopsis (Random)**

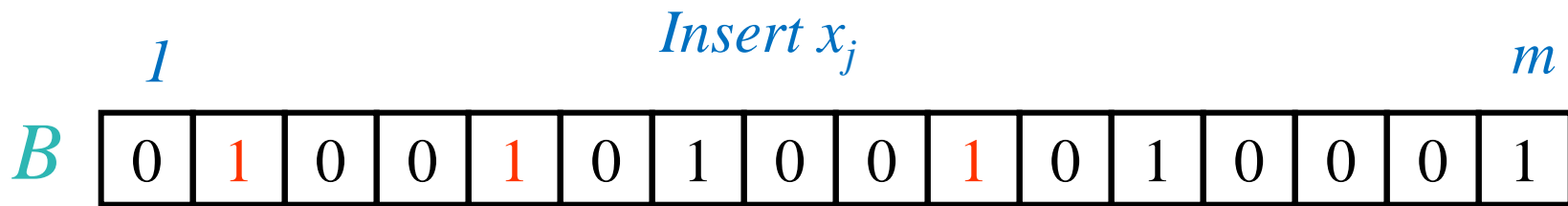
General but Non-optimal in
the Space and Accuracy trade-off

Bloom Filter (BF)

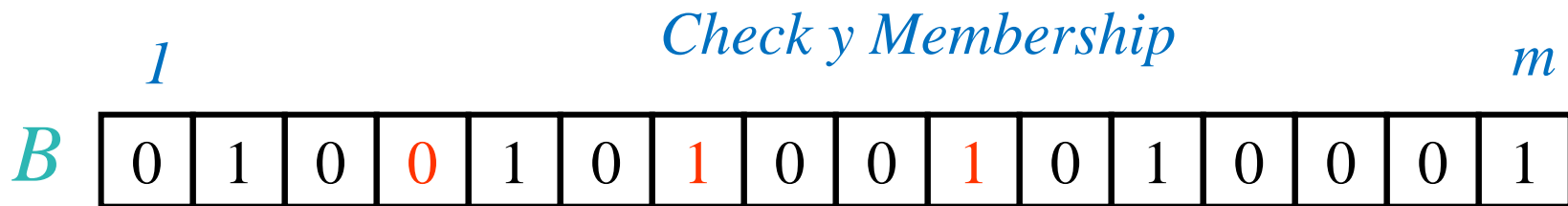
Initialization



Bloom Filter (BF)



Bloom Filter (BF)

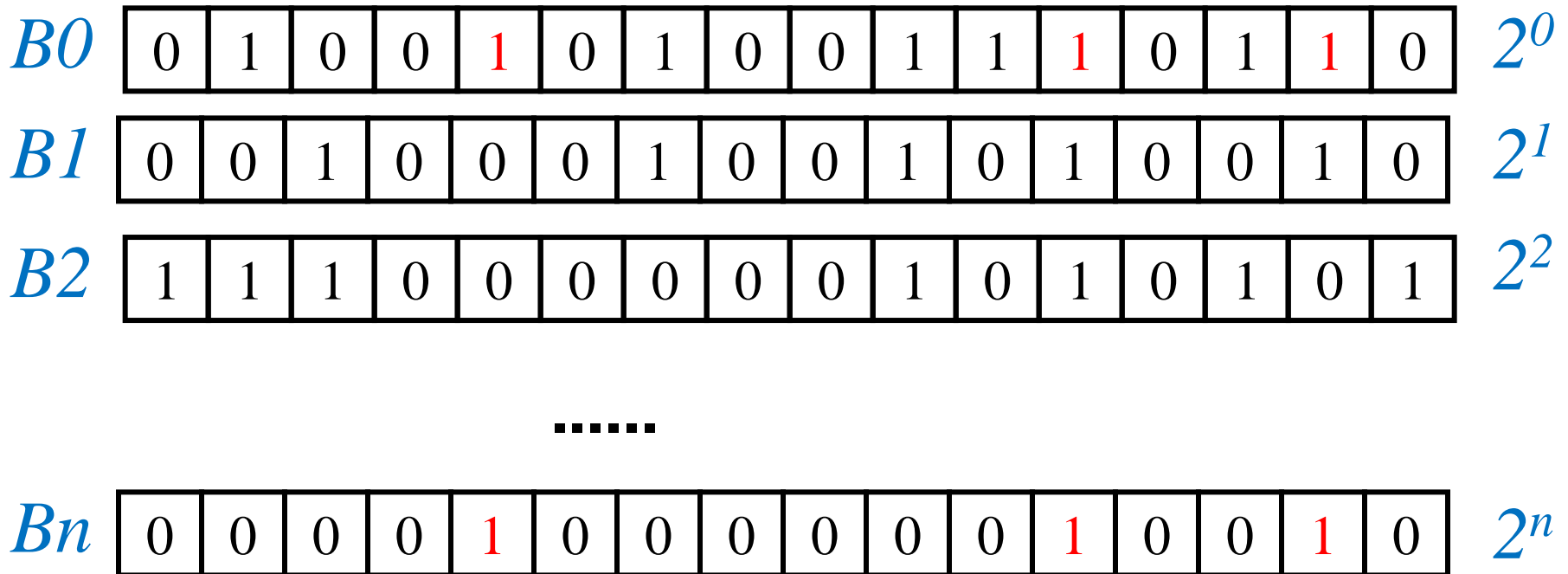


False Negative Error Rate = 0

False Positive Error Rate = $P[FP]$

Stratified Bloom Filter (SBF)

Insert (x_j, c_j) $c_j = 2^0 + 2^n$



Stratified Bloom Filter (SBF)

Retrieve count of y

B_0	0	1	0	0	1	0	1	0	0	1	1	1	0	1	1	0	2^0
B_1	0	0	1	0	0	0	1	0	0	1	0	1	0	0	1	0	2^1
B_2	1	1	1	0	0	0	0	0	0	1	0	1	0	1	0	1	2^2
.....																	
B_n	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	2^n

$$\text{count}(y) = 2^0 + 2^1$$

Important SBF Optimizations

- **Compensate for false positives**

$$count = \sum_{i=0}^n 2^i (1 - P[FP, Bi])$$

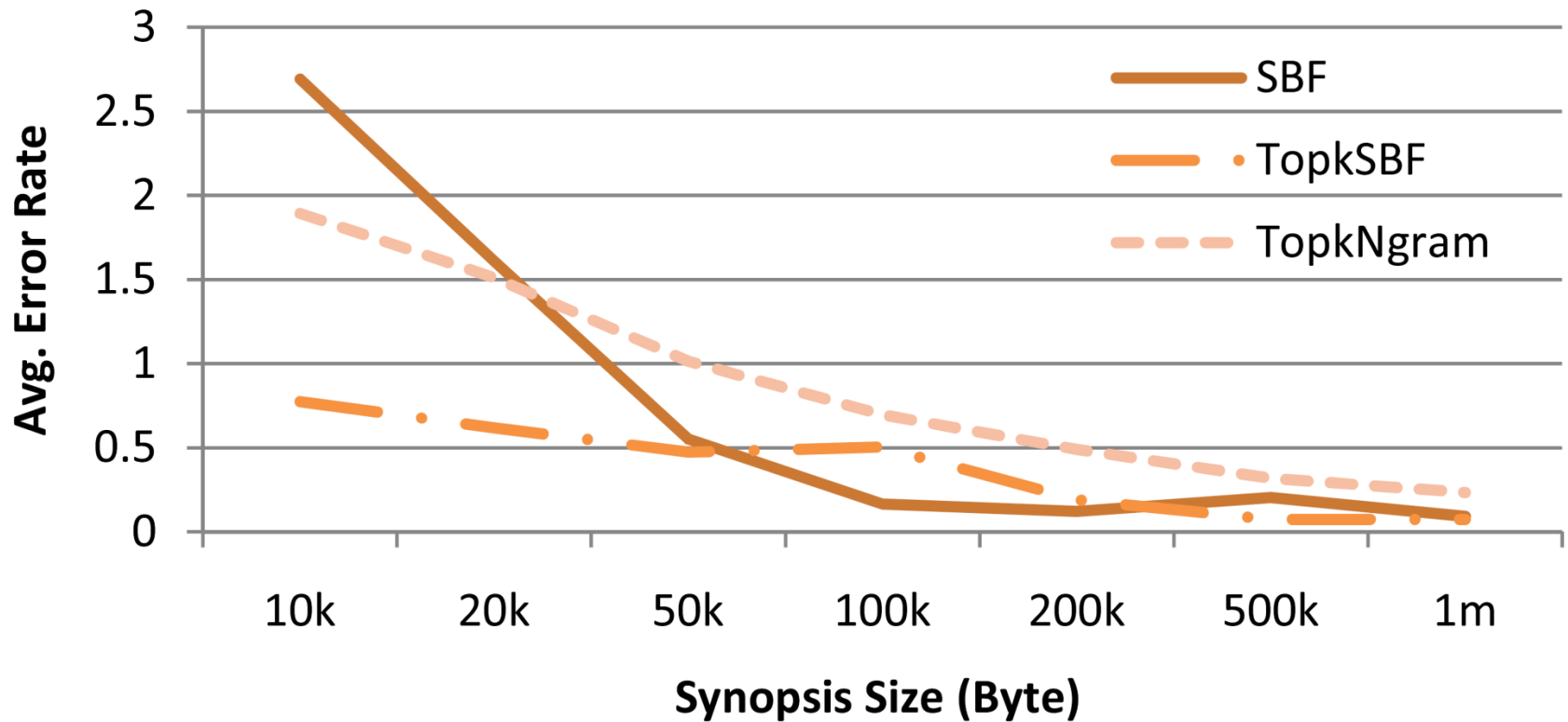
- **Re-allocate space among BFs**

- Optimization problem (minimizing variance)
- Hill-climbing algorithm

- **Top-k SBF**

- Hybrid of TopkNgram and SBF
- Hill-climbing to find proper k

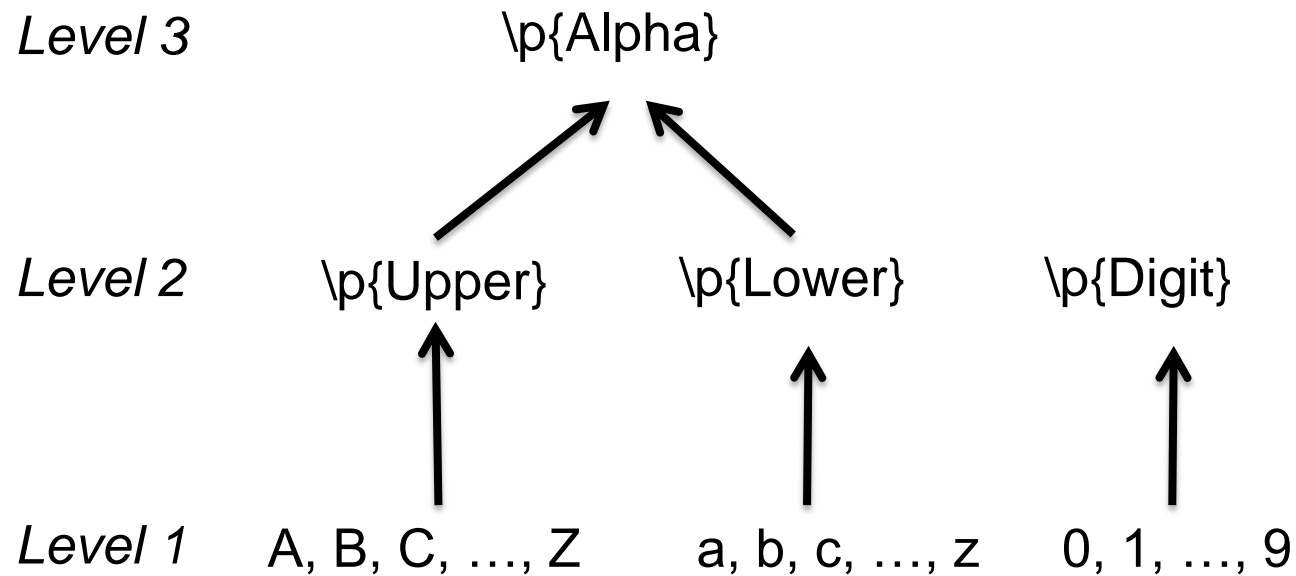
Evaluation 1: SBF vs. TopkNgram



Roll-up Synopsis

- **For regex selectivity estimation**
 - cannot use BF-based synopsis
- **Drop low-count N-grams → Summarize N-grams**
 - Multiple Ngrams summarized by one regex-Ngram
 - Summarize by roll-up operations

Character Class Lattice and Roll-up Operations



Utility Function for Roll-up Operations

- **Benefit: Reduced N-grams**

$$f_b = k - 1$$

- **Cost: Error induced by summarization**

$$f_c = d \times \sum_{i=1}^k (n_i \times c_i)$$

- **Utility Function**

$$f = k - 1 - d \times \sum_{i=1}^k (n_i \times c_i)$$

What is the best set of roll-up ops resulting in highest utility value?

Roll-up Synopsis – An Example

- **N-gram and count pairs**

{“the” 100, “App” 1, “All” 10}

- **3×3^3 candidate regex-Ngrams, If $d = 0.01$**

$f(\backslash p\{Alpha\}\{3\}) = -4.66$

$f(A\backslash p\{Lower\}\{2\}) = 0.78$

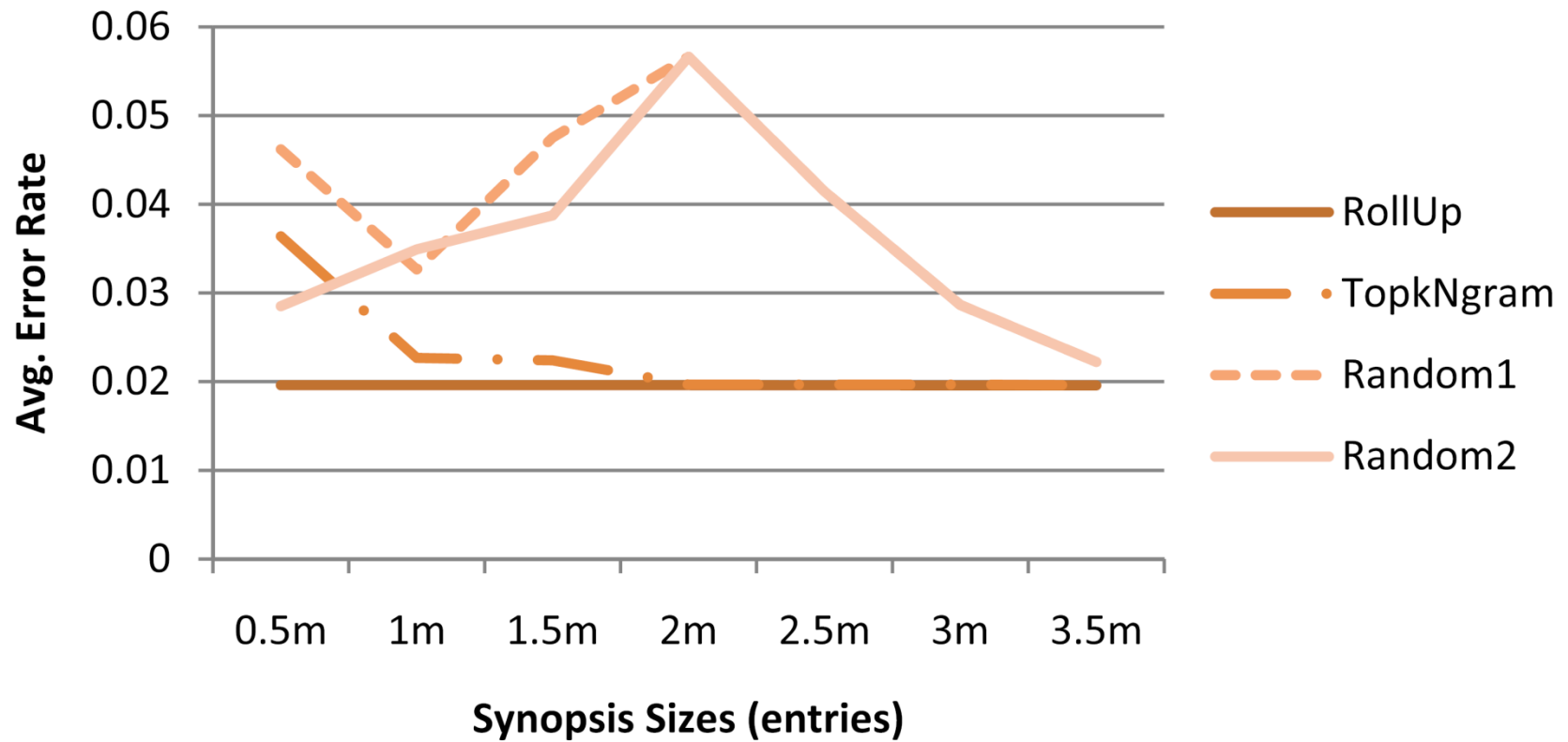
- **Resulting best Roll-up Synopsis**

{“the” 100, “A\p{Lower}\{2}” 11}

Algorithms and Challenges

- **Constructor – Greedy algorithm**
 - Maximize the utility function
- **Prune the huge search space**
- **Dealing with complex regexes**
 - Decompose into sub-regexes
 - Combine the selectivity estimates

Evaluation 2: Roll-up vs. TopkNgram



Conclusion

- **Defined selectivity estimation problem**
- **Developed novel document synopses**
- **Reduce estimation error to $\frac{1}{2}$ -- $\frac{1}{5}$ of the baselines**
- **Future work**
 - Cost Estimation for extractors
 - Selectivity Estimation for Join
 - Enhance the cost-based SystemT Optimizer



Thank you! Question?