

## A Two-Tire Index Structure for Approximate String Matching with Block Moves

Bin Wang, Long Xie and Guoren Wang  
binwang@mail.neu.edu.cn

MCIS'09

## Approximate String Selections

$p =$  Nicolas Kage



$S = \{$  Nicolas Cage, Will Smith, Bin Wang  $\}$

Measuring the distance between two strings:

- Edit Distance
- Jaccard
- Cosine
- ...

## Block Edit Distance

$$ED(\text{BinWang}, \text{WangBin}) = 6 \quad O(mn)$$

But,

*It is expected to reduce the search space to answer the query.*

Block Edit Distance extends classical edit distance with block moves.

$$BED(\text{BinWang}, \text{WangBin}) = 1 \quad \text{An NPC Problem!!}$$

## Outline

- Motivation
- Related Work
  - Frequency Distance
  - Positional  $q$ -grams
- Our Approach
  - Index Construction
  - Query Processing
- Experimental Results

## Frequency Distance

Given a string  $s$ , we call the number of duplicated number of a character in  $s$  is the *local frequency*, denoted by  $f(s)$ .

$$f(\text{aggc}) = \{a \cdot 1, c \cdot 1, g \cdot 2\} \quad f(\text{aaccg}) = \{a \cdot 2, c \cdot 2, g \cdot 1\}$$

Need to delete/substitute a **g**  $posDist = 1$

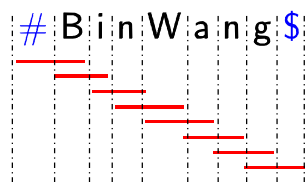
Need to delete/substitute an **a** and a **c**  $negDist = 2$

$$FD(\text{aggc}, \text{aaccg}) = \max(posDist, negDist) = 2$$

$$FD(s, p) \leq BED(s, p) \text{ (VLDB'01)}$$

## Positional $q$ -grams

Given a string  $s$  and a positive integer  $q$ , a *positional  $q$ -gram* of  $s$  is a pair  $(i, s[i, i + q - 1])$ .



2-grams

$$G(\text{BinWang}, 2) = \{(1, \#B), (2, Bi), (3, in), (4, nW), (5, Wa), (6, an), (7, ng), (8, g\$)\}$$

$$G(\text{WangBin}, 2) = \{(1, \#W), (2, \underline{Wa}), (3, \underline{an}), (4, \underline{ng}), (5, gB), (6, \underline{Bi}), (7, \underline{in}), (8, \underline{n\$})\}$$

$BED(\text{BinWang}, \text{WangBin}) = 1$ , and they share 5 common 2-grams.

If  $BED(s, p) \leq k$ , then  $s$  and  $p$  must share at least  $(\max(|s|, |p|) - 1 - 3(q - 1)k)$   $q$ -grams. (VLDB'01)

## Our Approach

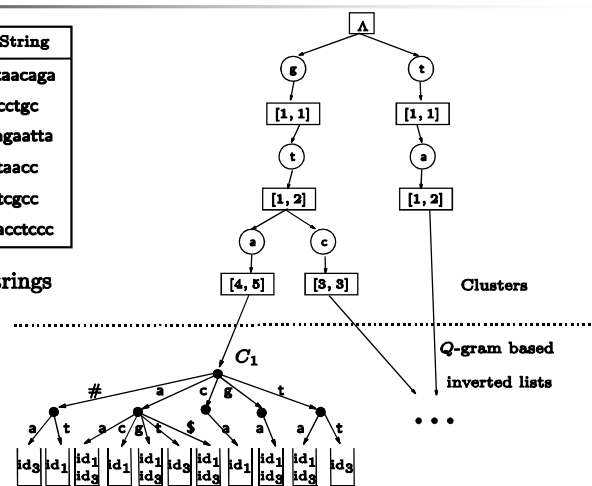
- Frequency Distance:
  - Advantages: it is *easy to calculate*;
  - Disadvantages: the *filterability is weak*.
- Positional  $q$ -grams:
  - Advantages: it can *prune away more non-candidates*;
  - Disadvantages: it requires *more overhead on decomposing strings into gram sets*.

Our approach is using a novel two-tier index structure *to combine the advantages* of both  $FD$  and positional  $q$ -grams.

## A Two-Tier Index Structure

ID	String
id <sub>1</sub>	taacaga
id <sub>2</sub>	cctgc
id <sub>3</sub>	agaatta
id <sub>4</sub>	taacc
id <sub>5</sub>	tcgcc
id <sub>6</sub>	acctccc

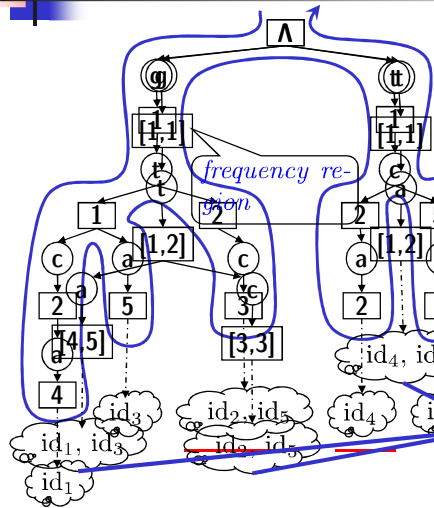
(a) Strings



(b) 2TI



## Compact Cluster Trie

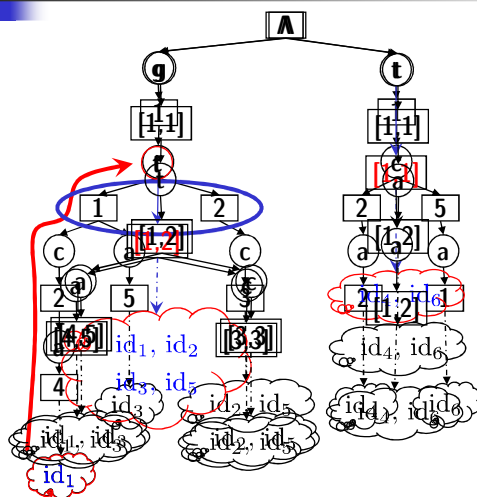


In order to solve the above problems, we merge nodes in the cluster trie to construct a **COMPACT CLUSTER TRIE**.

For strings  $s_1$  and  $s_2$ ,  $f_1$  and  $f_2$  clusters should be skewed distributed.

The size of each cluster is in between  $[\theta_{min}, \theta_{max}]$ .

## Compact Cluster Trie Cont.



$$\theta_r = 1$$

$$[\theta_{min}, \theta_{max}] = [2, 2]$$

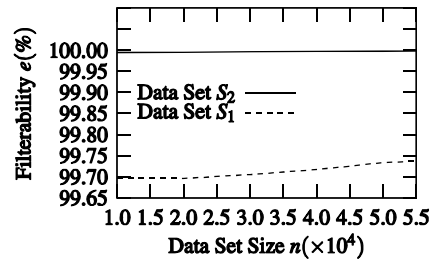
ID	String
id <sub>1</sub>	taacaga
id <sub>2</sub>	cctgc
id <sub>3</sub>	agaatta
id <sub>4</sub>	taacc
id <sub>5</sub>	tcgcc
id <sub>6</sub>	acctccc

(a) Strings





## Performance Evaluation Cont.



$$Filterability = 1 - \frac{|S_C|}{|S|}$$

where  $S_C$  denotes the candidate set generated by filtering with our indices, and  $S$  denotes the string collection.



*Thank You!*