Probabilistic String Similarity Joins

Jeffrey Jestes Feifei Li

Florida State University

Zhepeng Yan Ke Yi

Hong Kong University
of Science and Technology

Introduction

- Probabilistic data naturally arises in
  - data cleaning
  - data integration
  - scientific applications
  - many other applications

Introduction

- Probabilistic data naturally arises in
  - data cleaning
  - data integration
  - scientific applications
  - many other applications
- String similarity join is an important operator for these applications

R	
id	S
1	Microsoft
2	Yahoo!
⋮	⋮

$\bowtie_S$

T	
id	S
1	Google
2	AT&T
⋮	⋮

Introduction

- Uncertainty naturally arises in string data

Yahoo!

Yahoo Labs

Yahoo

Yahoo! Labs

Introduction

- Uncertainty naturally arises in string data

??? Yahoo! ??? Yahoo Labs

Yahoo ??? Yahoo! Labs ???

Introduction

- Uncertainty naturally arises in string data

0.10 Yahoo

Yahoo! 0.15

0.40 Yahoo! Labs

Yahoo Labs 0.35

Introduction

- Uncertainty naturally arises in string data

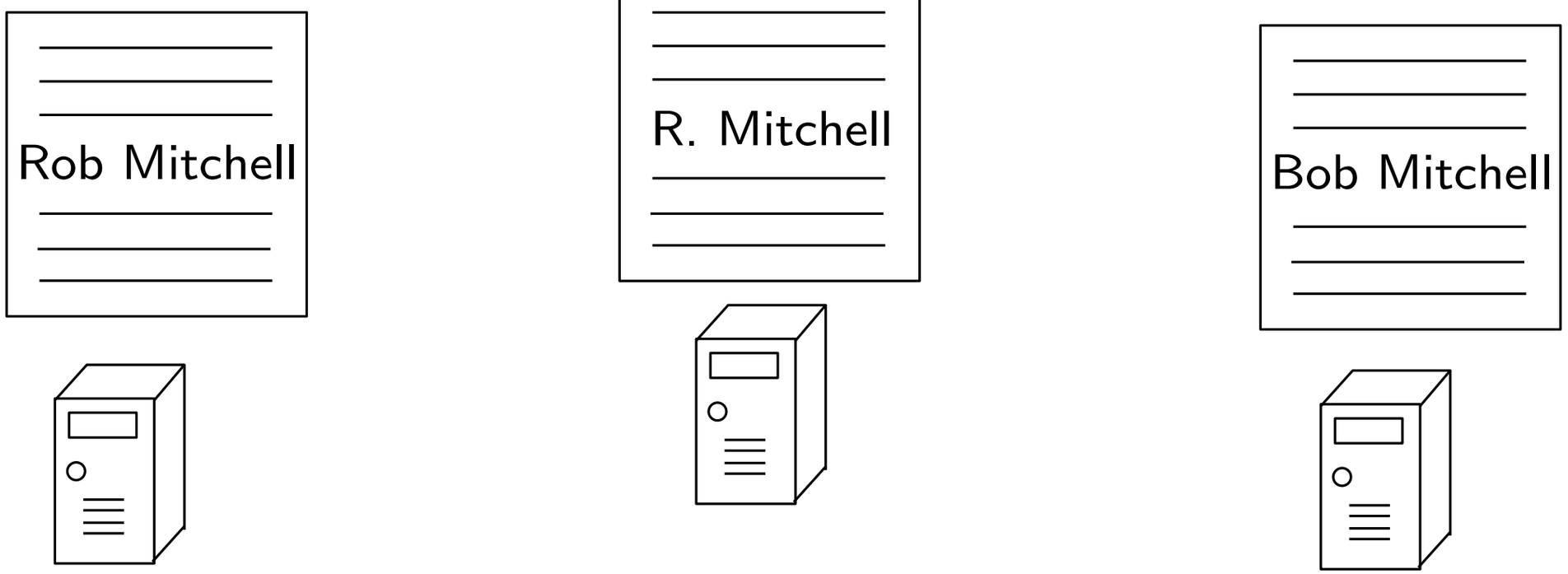
0.10 Yahoo
Yahoo! 0.15
0.40 Yahoo! Labs
Yahoo Labs 0.35

R	
id	S
1	{ (Microsoft, 0.10), (Microsoft Research, 0.70), (Microsoft Labs, 0.20) }
2	{ (Yahoo, 0.10), (Yahoo!, 0.15), (Yahoo! Labs, 0.40), (Yahoo Labs, 0.35) }
⋮	⋮

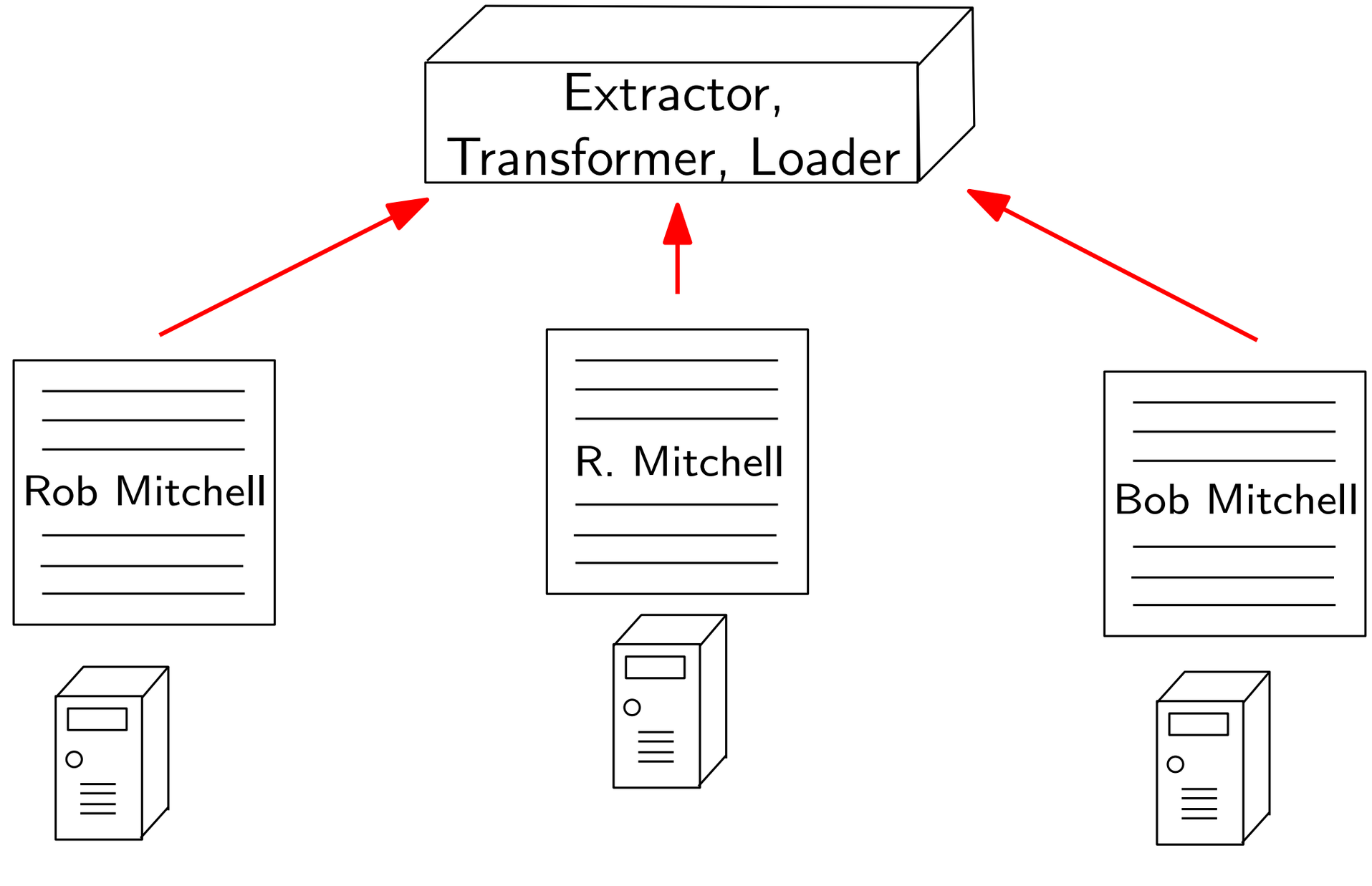
$\bowtie S$

T	
id	S
1	{ (Google, 1) }
2	{ (AT&T, 0.70), (ATT, 0.30) }
⋮	⋮

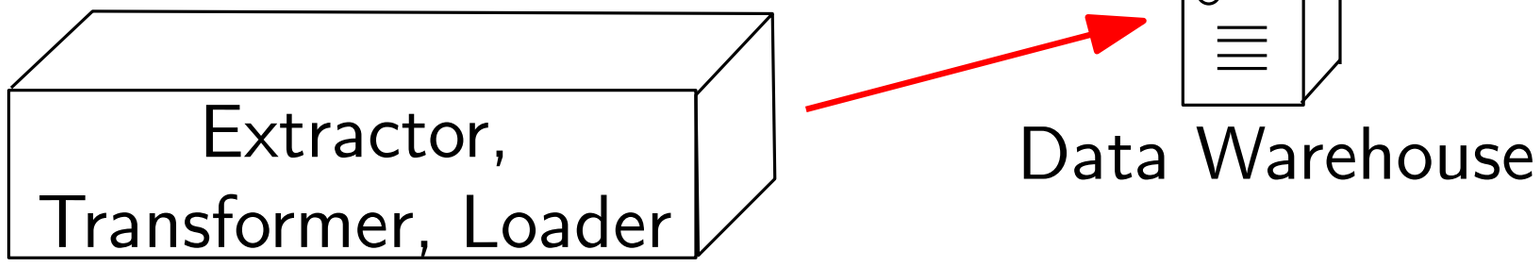
Probabilistic Strings: Data Integration

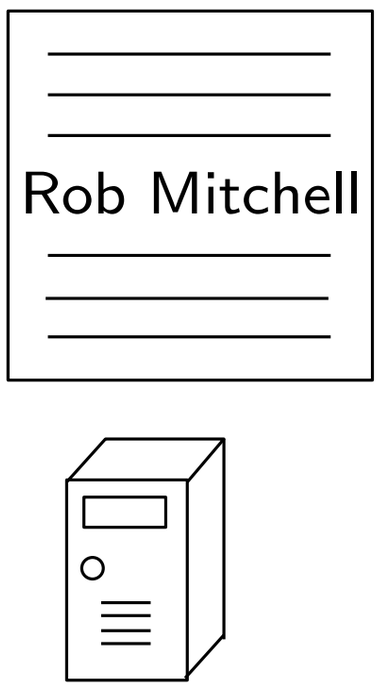


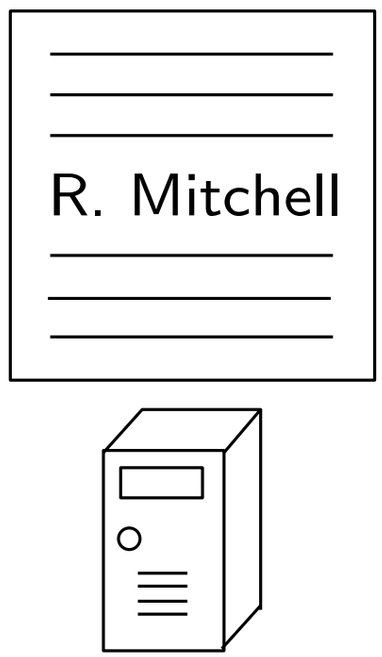
Probabilistic Strings: Data Integration

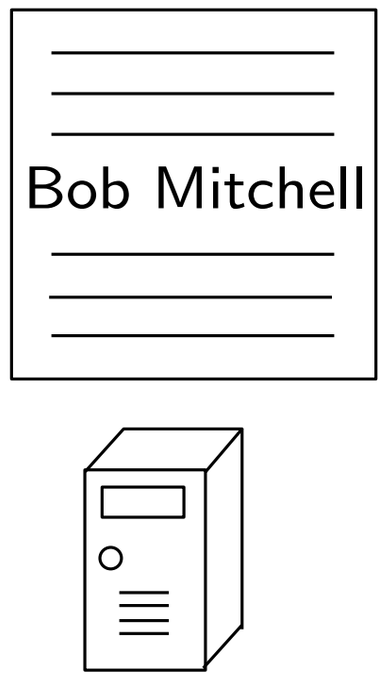


Probabilistic Strings: Data Integration

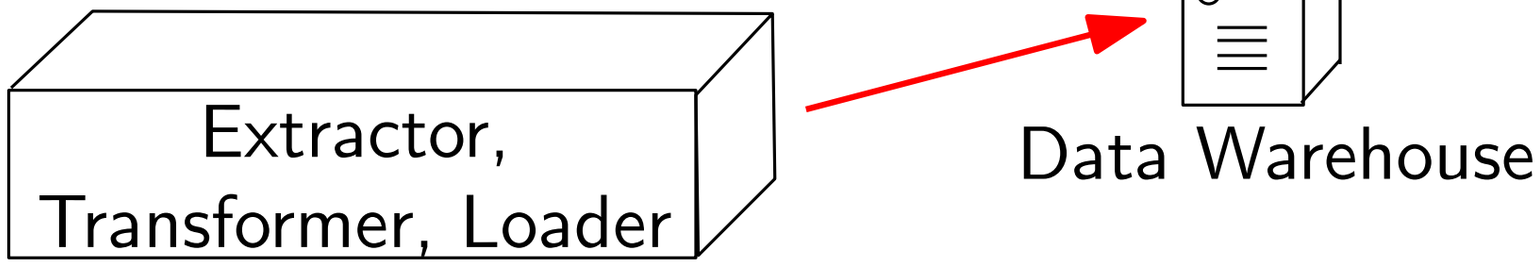


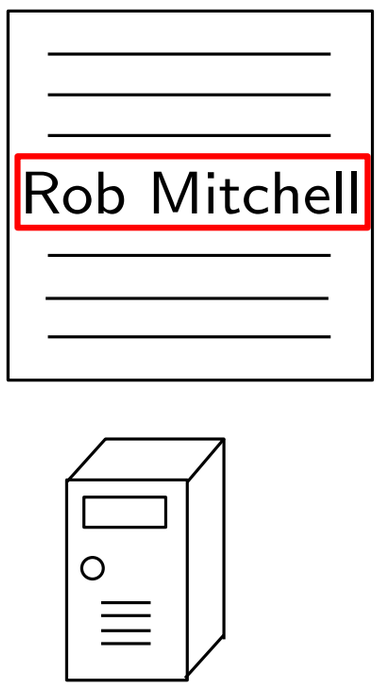


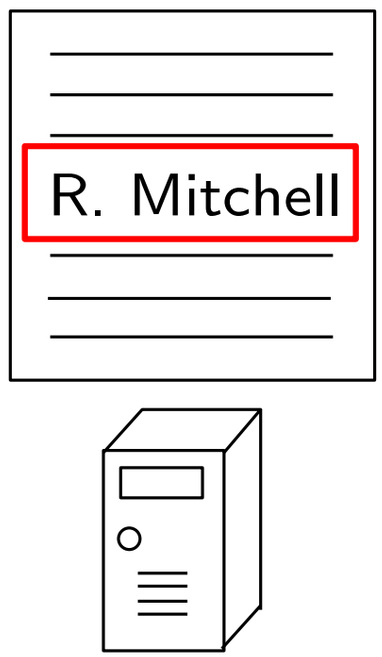


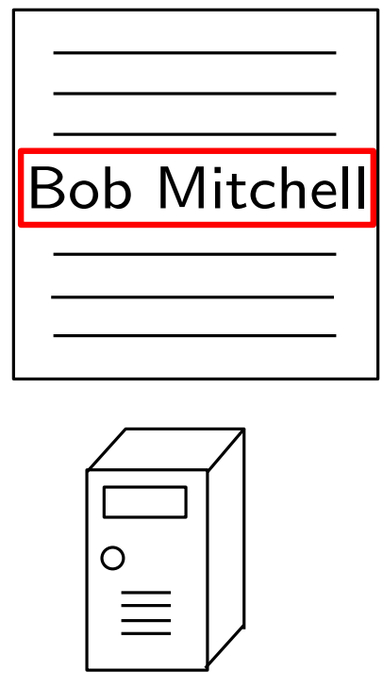


Probabilistic Strings: Data Integration









String-Level Probabilistic Model

String S (σ_i's)	Probability (p_i's)
Bob Mitchell	0.75
Rob Mitchell	0.10
Robert Mitchell	0.05
Robby Mitchell	0.10

String-Level Probabilistic Model

String S (σ_i's)	Probability (p_i's)
Bob Mitchell	0.75
Rob Mitchell	0.10
Robert Mitchell	0.05
Robby Mitchell	0.10

S(1) = (σ_1, p_1) = ("Bob Mitchell", 0.75)
S(2) = (σ_2, p_2) = ("Rob Mitchell", 0.10)
S(3) = (σ_3, p_3) = ("Robert Mitchell", 0.05)
S(4) = (σ_4, p_4) = ("Robby Mitchell", 0.10)

String-Level Probabilistic Model

String S (σ_i's)	Probability (p_i's)
Bob Mitchell	0.75
Rob Mitchell	0.10
Robert Mitchell	0.05
Robby Mitchell	0.10

S(1) = (σ_1, p_1) = ("Bob Mitchell", 0.75)
S(2) = (σ_2, p_2) = ("Rob Mitchell", 0.10)
S(3) = (σ_3, p_3) = ("Robert Mitchell", 0.05)
S(4) = (σ_4, p_4) = ("Robby Mitchell", 0.10)

String-Level Probabilistic Model

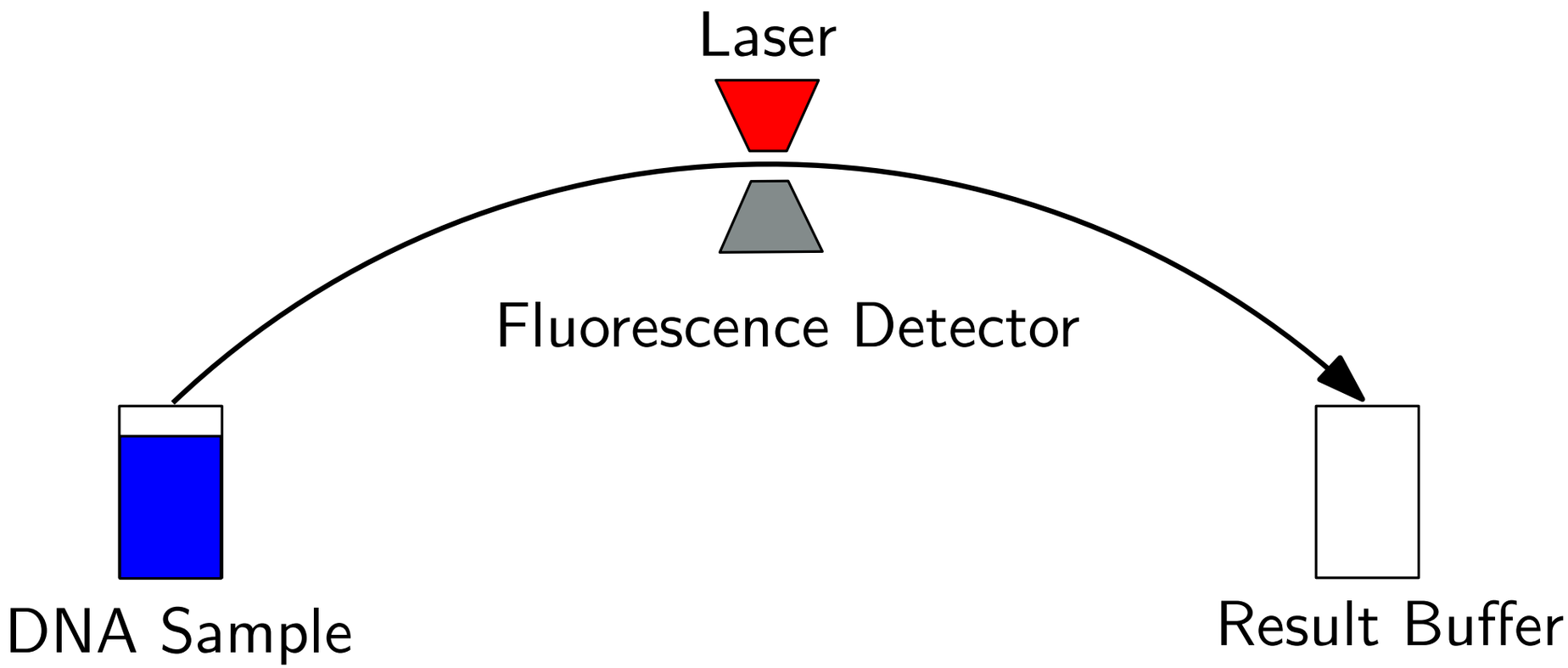
String S (σ_i's)	Probability (p_i's)
Bob Mitchell	0.75
Rob Mitchell	0.10
Robert Mitchell	0.05
Robby Mitchell	0.10

S(1) = (σ_1, p_1) = ("Bob Mitchell", 0.75)
S(2) = (σ_2, p_2) = ("Rob Mitchell", 0.10)
S(3) = (σ_3, p_3) = ("Robert Mitchell", 0.05)
S(4) = (σ_4, p_4) = ("Robby Mitchell", 0.10)

$S = \{(\sigma_1, p_1), (\sigma_2, p_2), \ldots, (\sigma_m, p_m)\}$
where $\sum_{i=1}^{m} p_i = 1$ and $\sigma_i \in \Sigma^*$

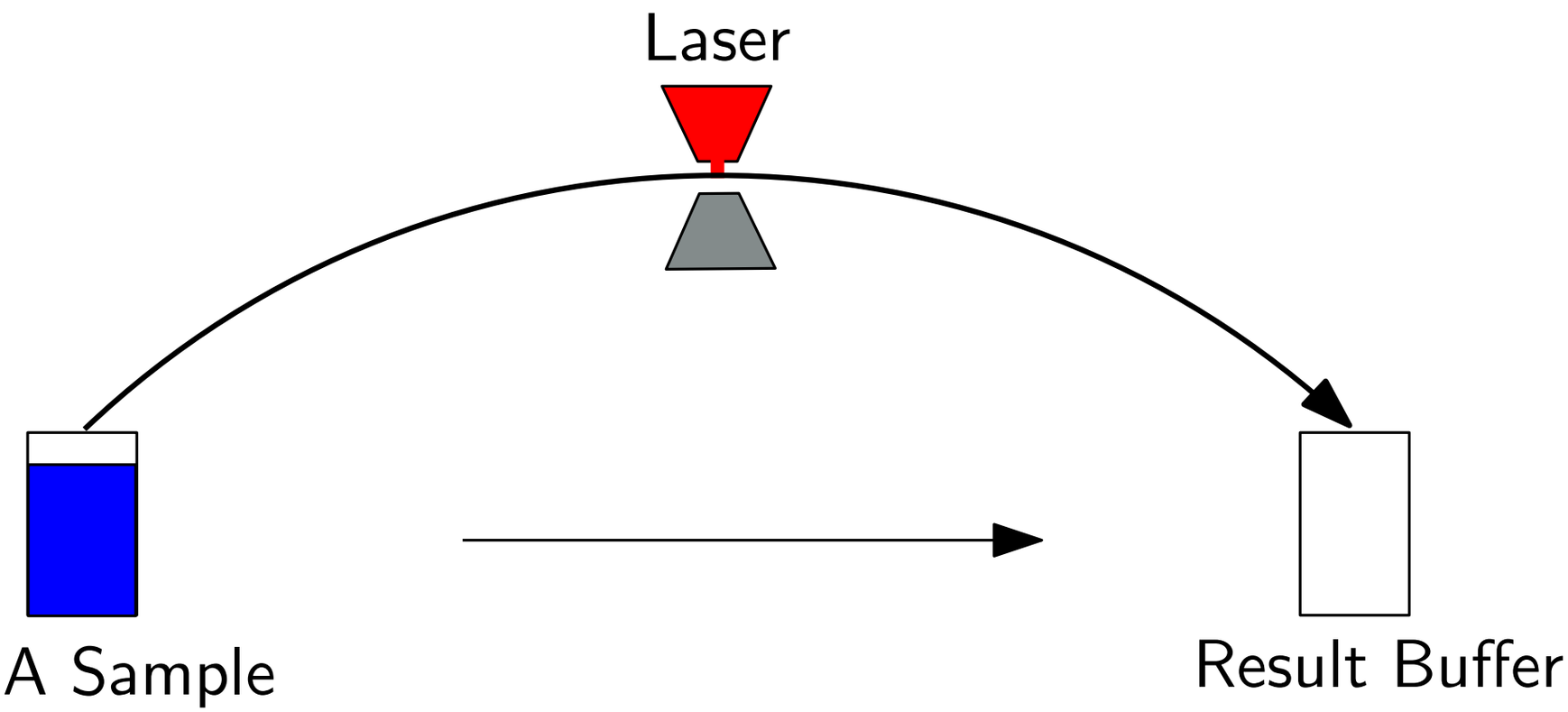
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



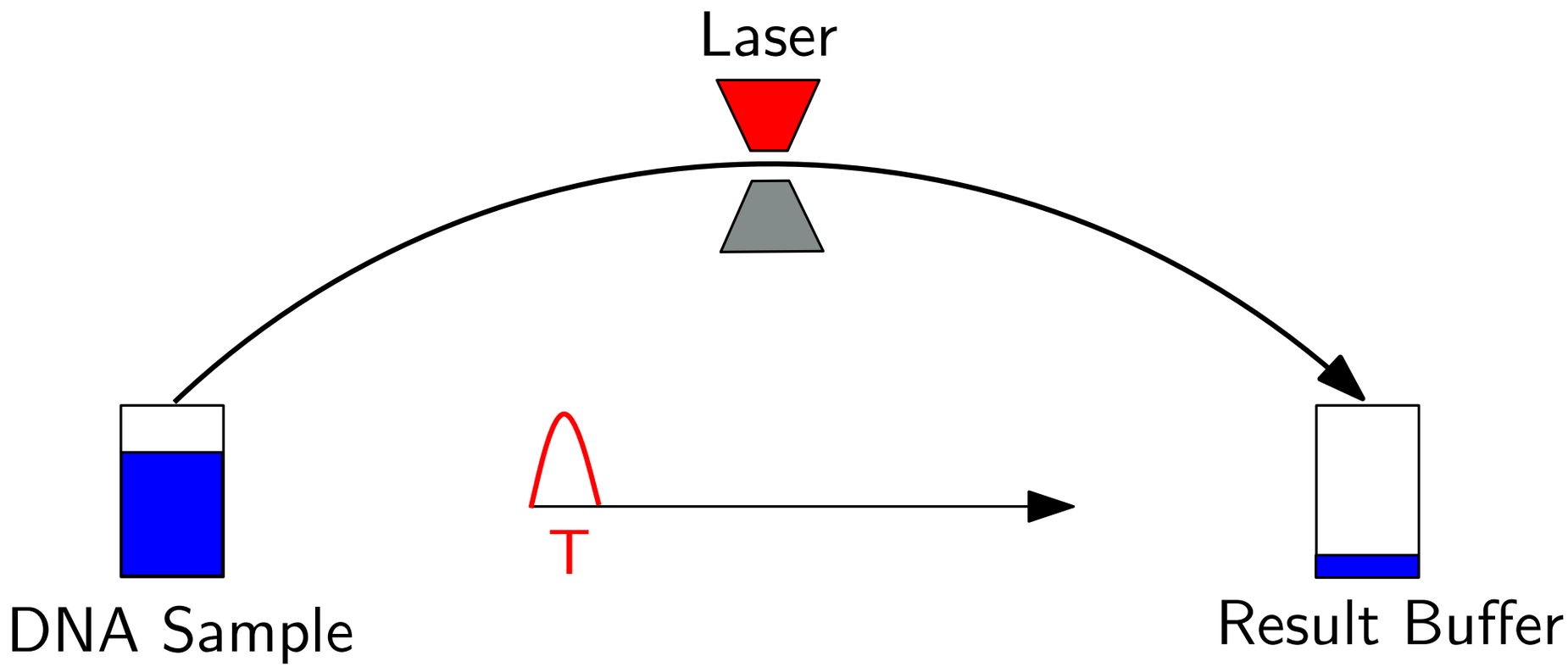
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



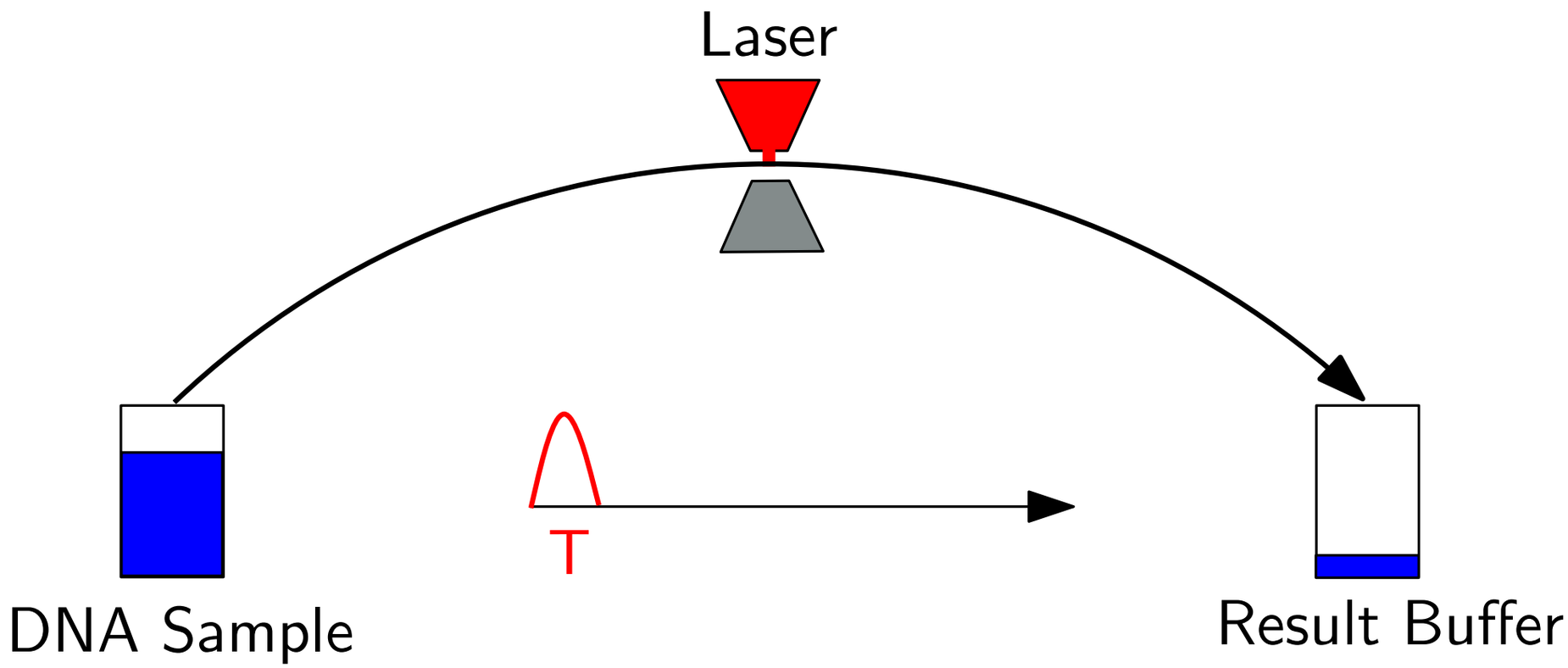
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



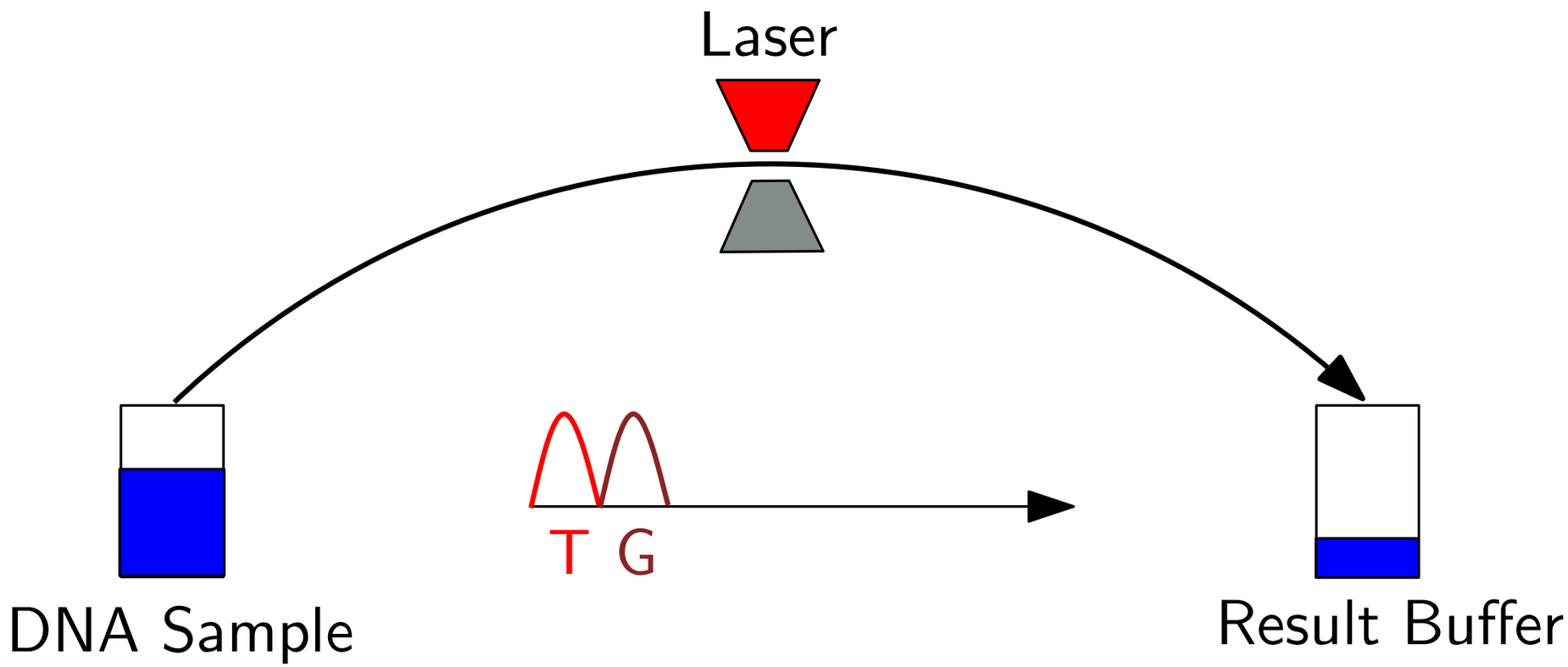
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



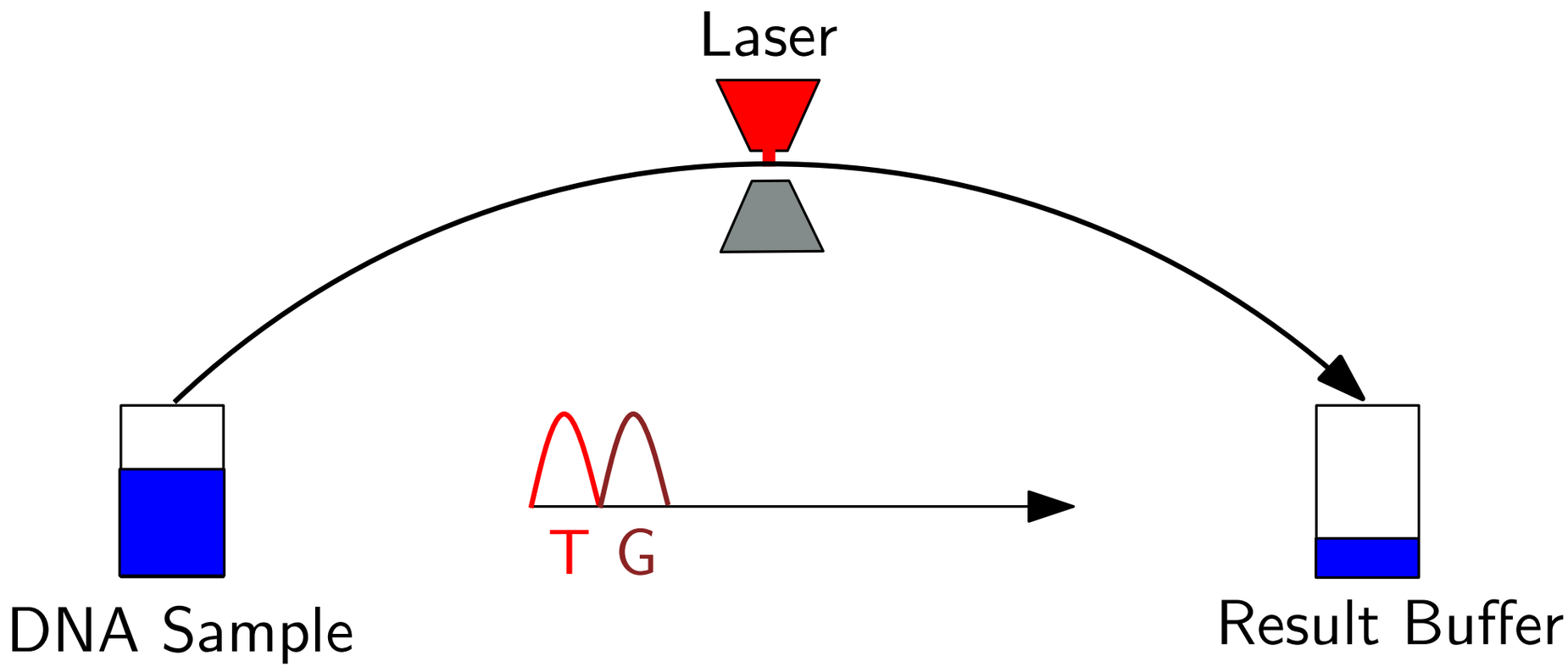
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



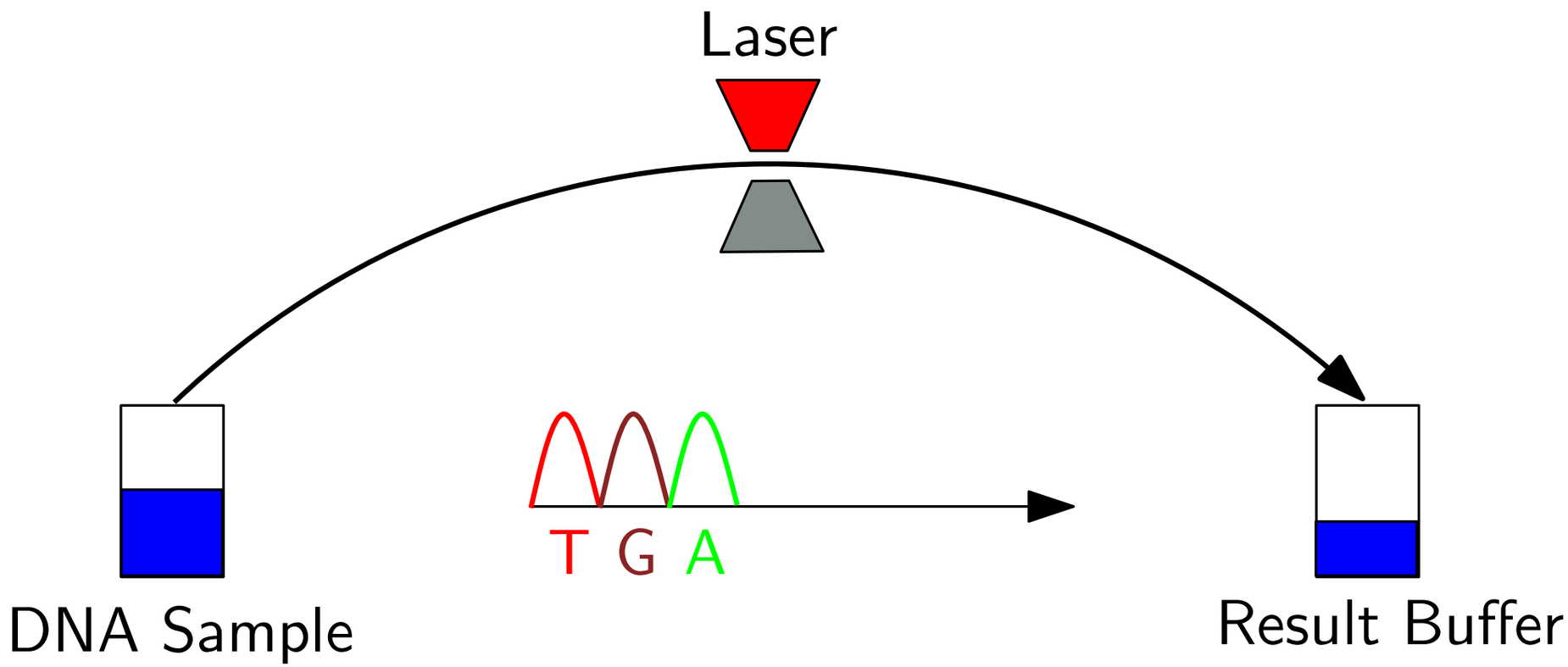
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



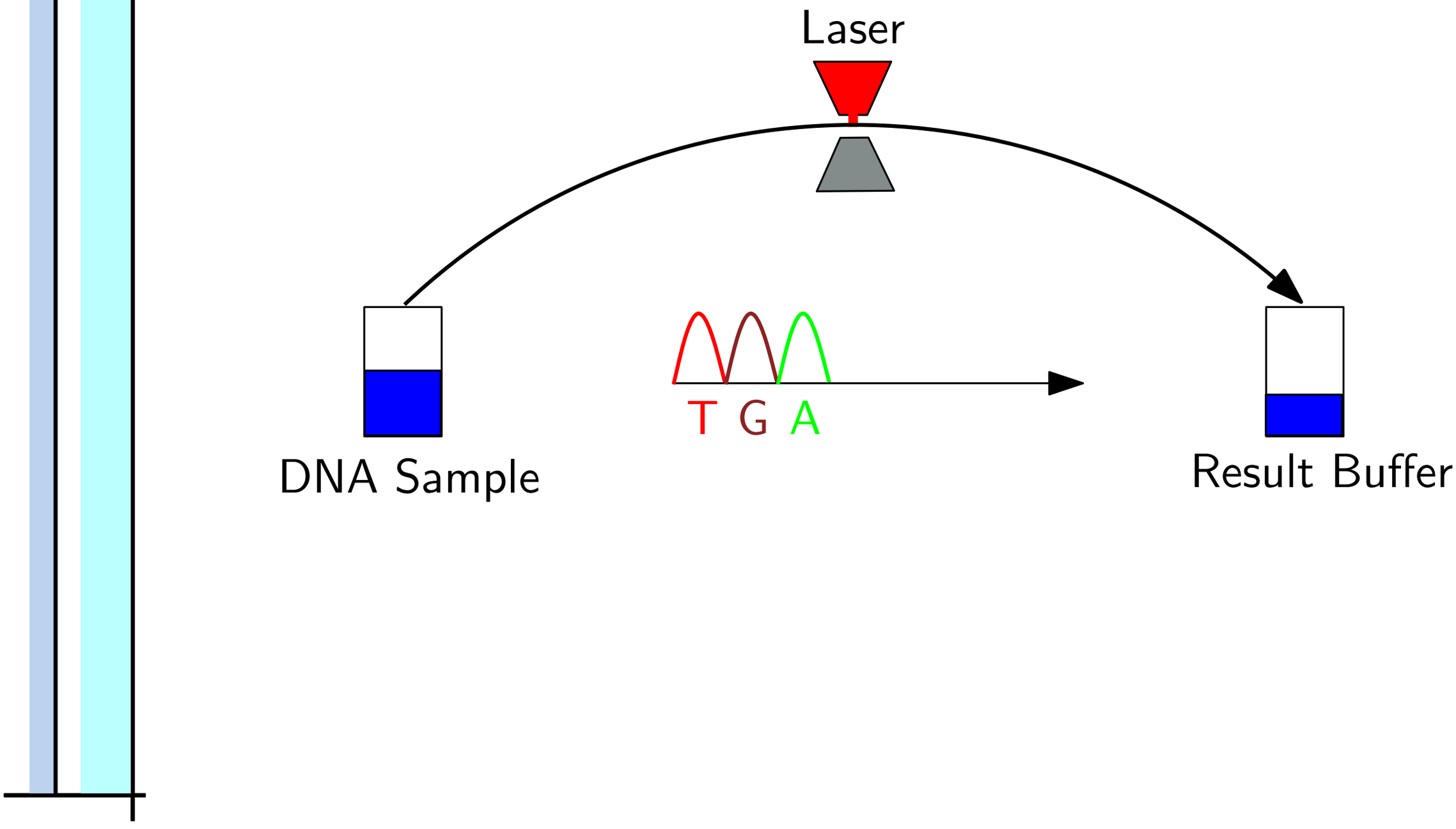
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



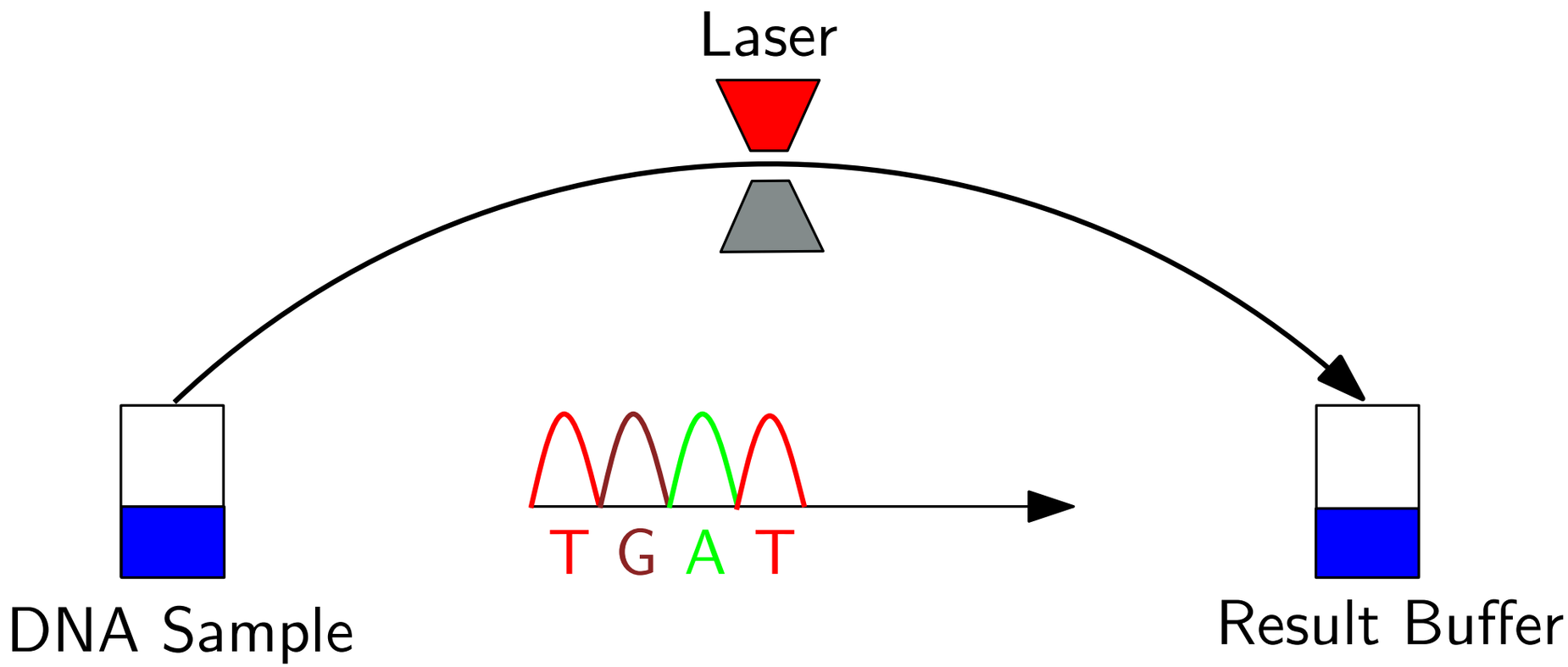
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



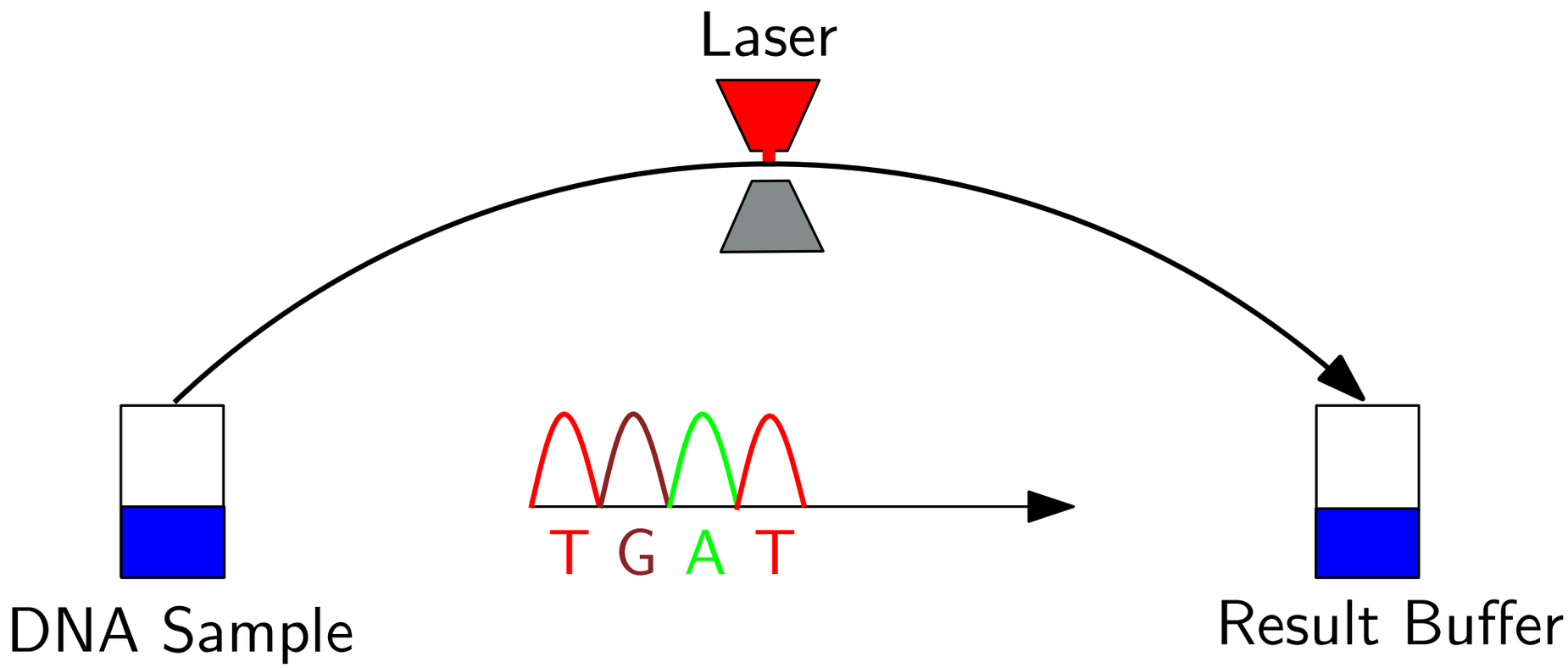
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



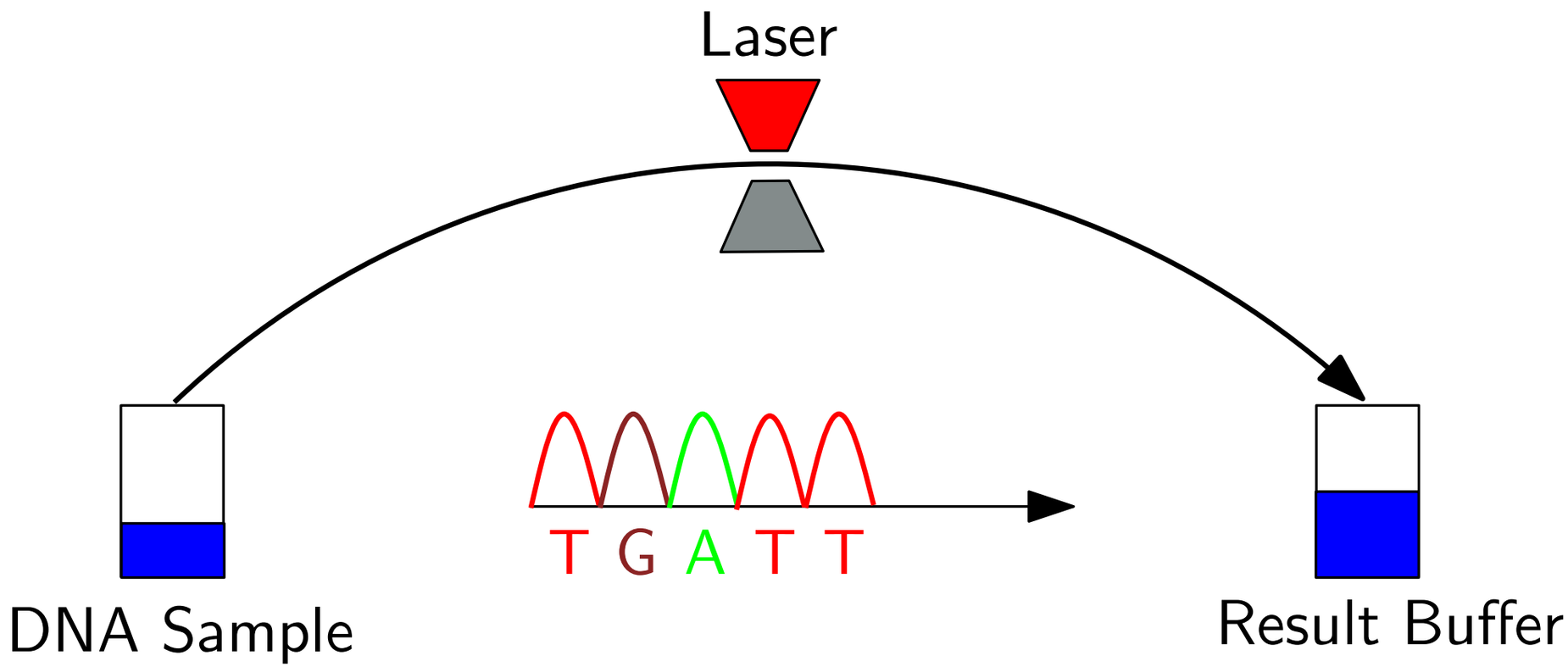
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



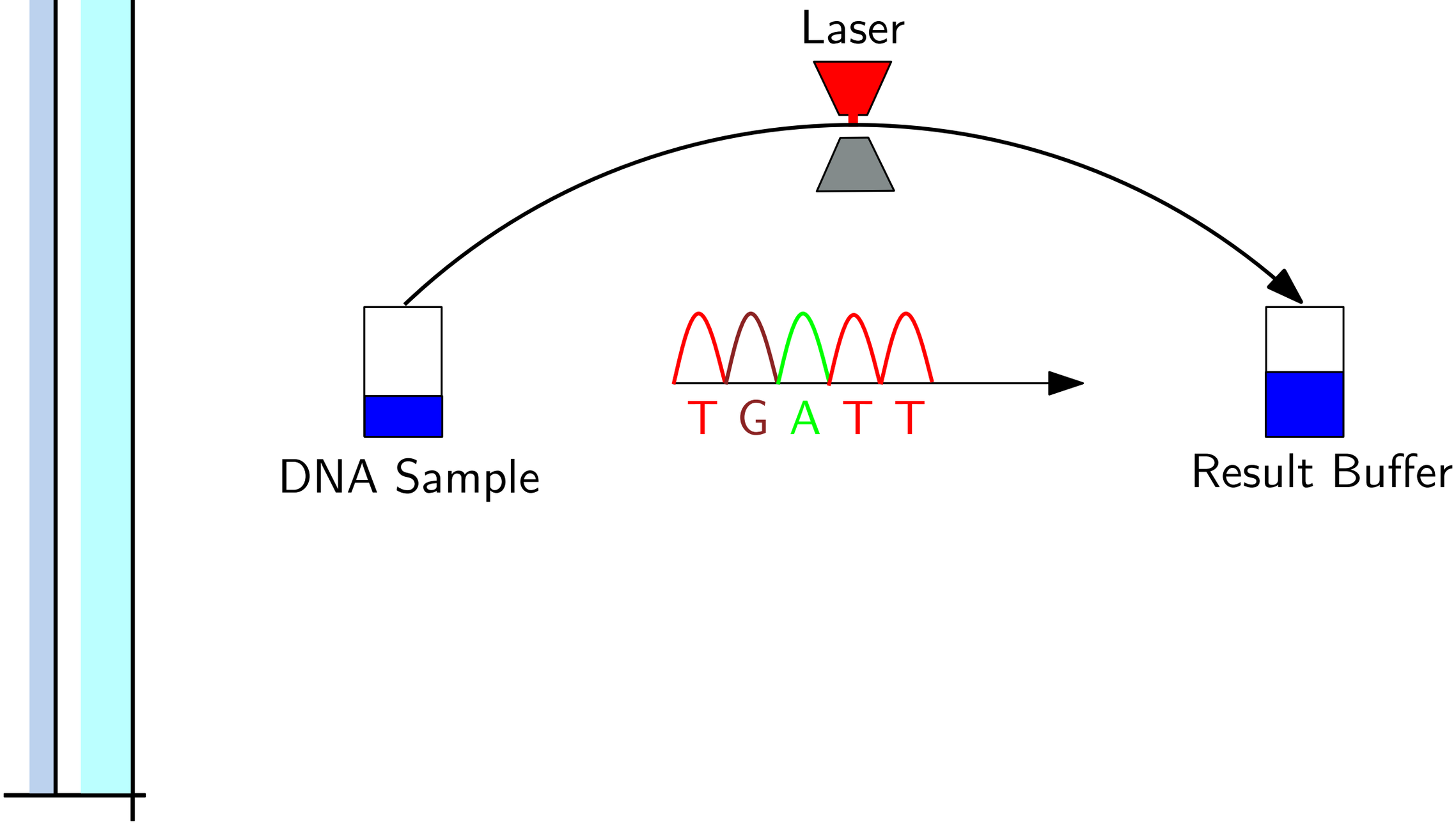
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



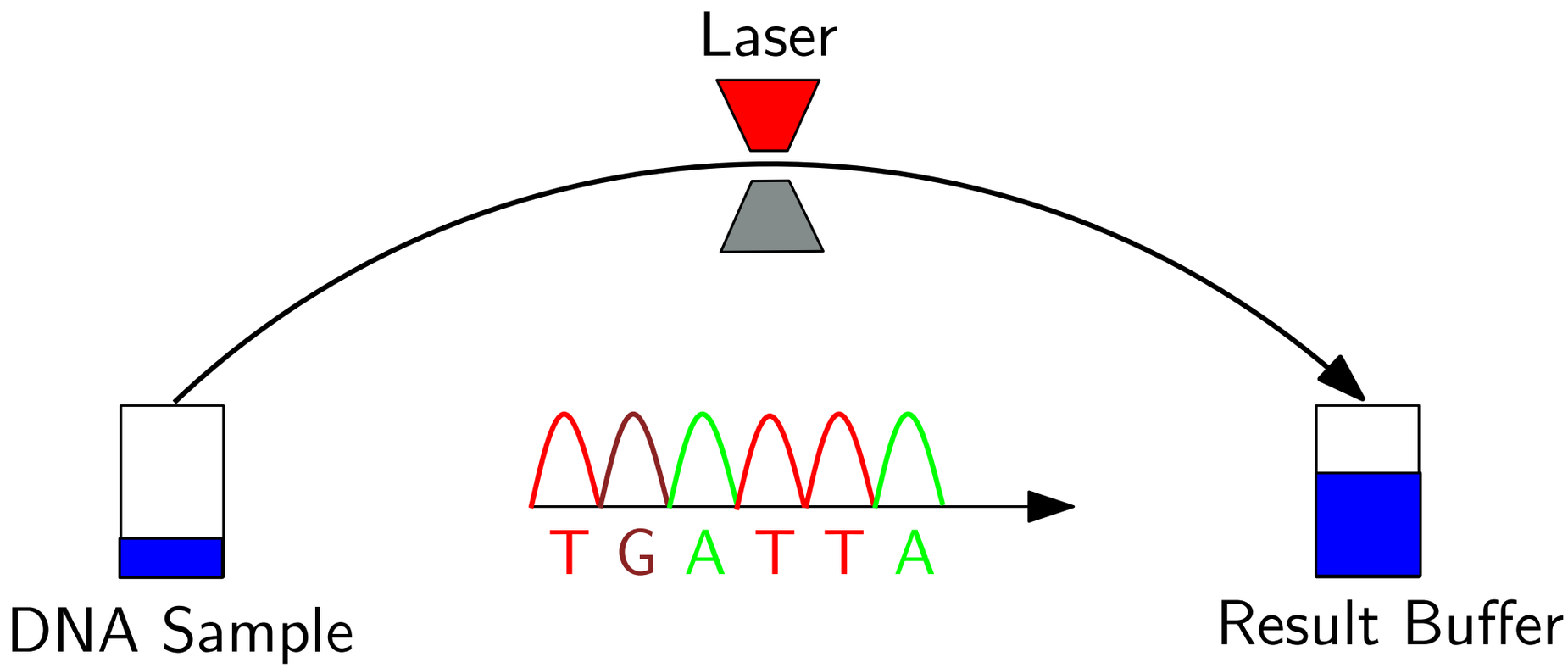
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



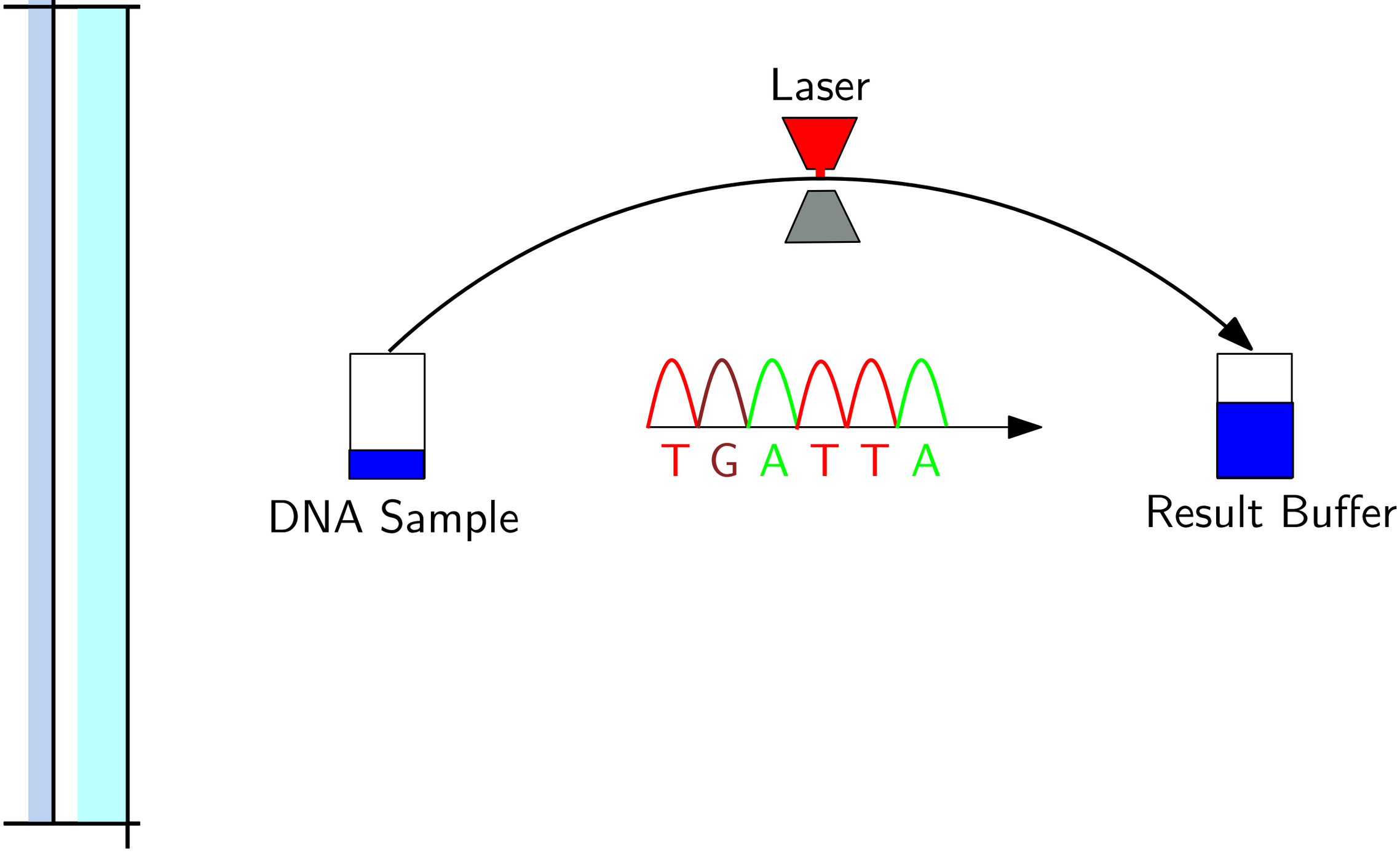
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



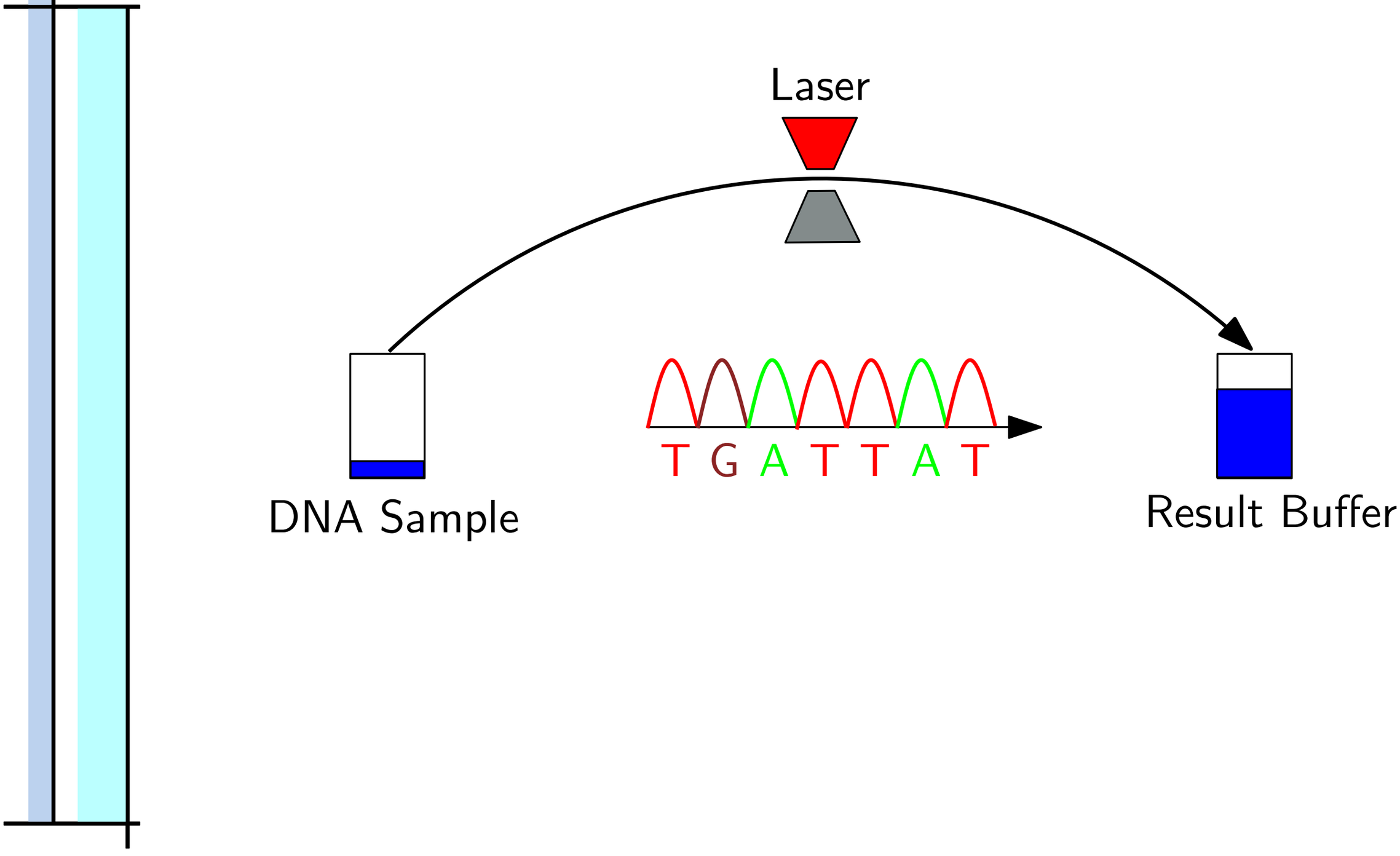
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



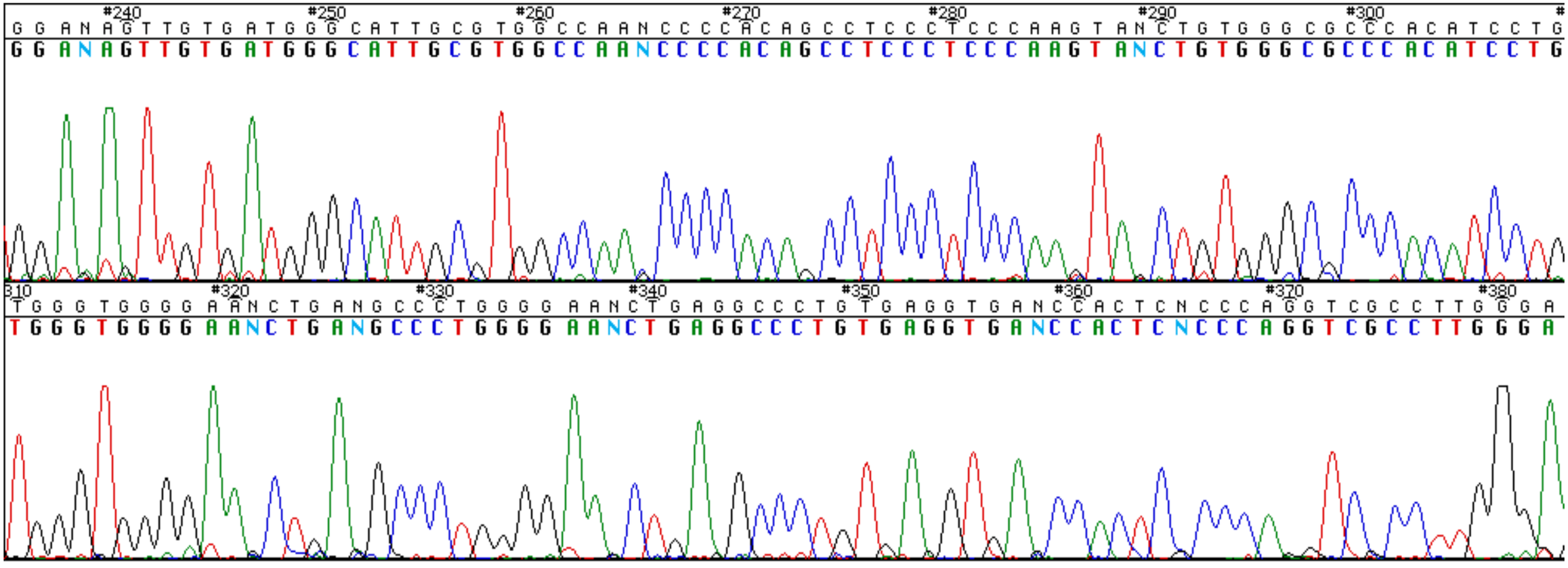
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing



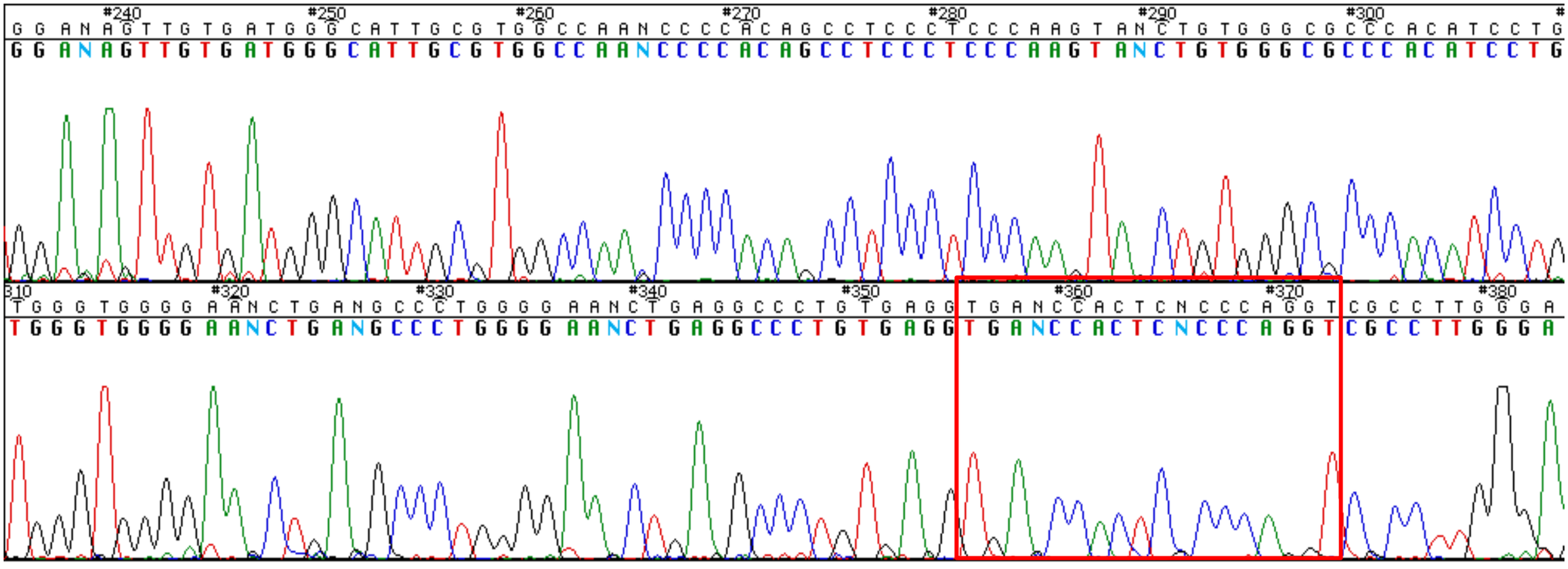
Probabilistic Strings: Scientific Applications

DNA Sequencing: Automated Dye-terminator Sequencing

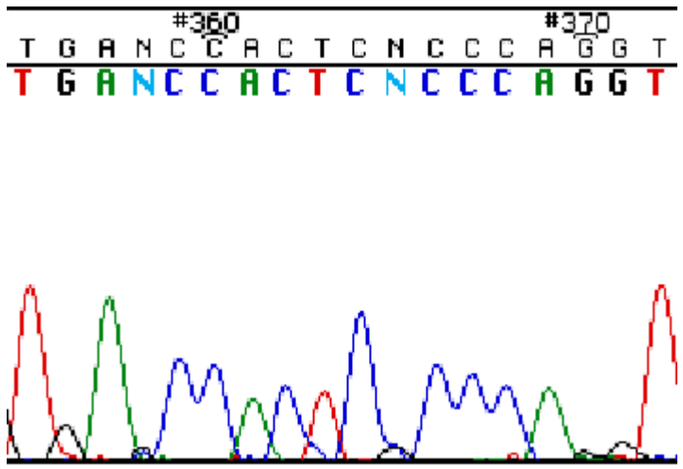


Probabilistic Strings: Scientific Applications

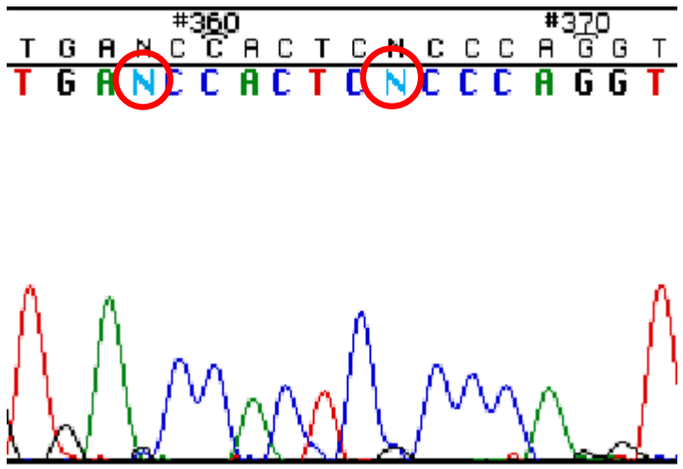
DNA Sequencing: Automated Dye-terminator Sequencing



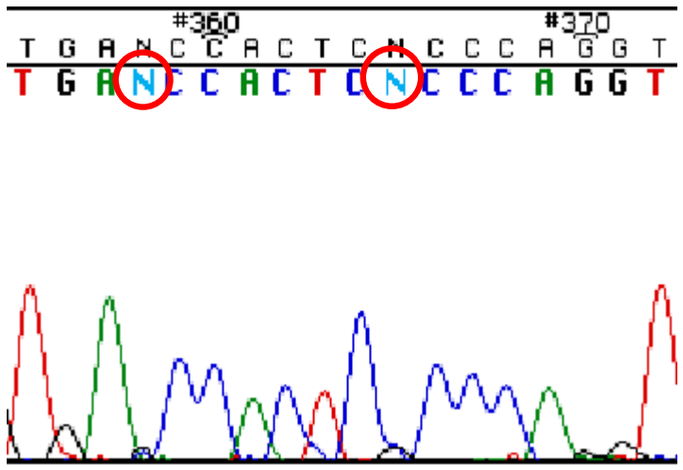
Probabilistic Strings: Scientific Applications



Probabilistic Strings: Scientific Applications

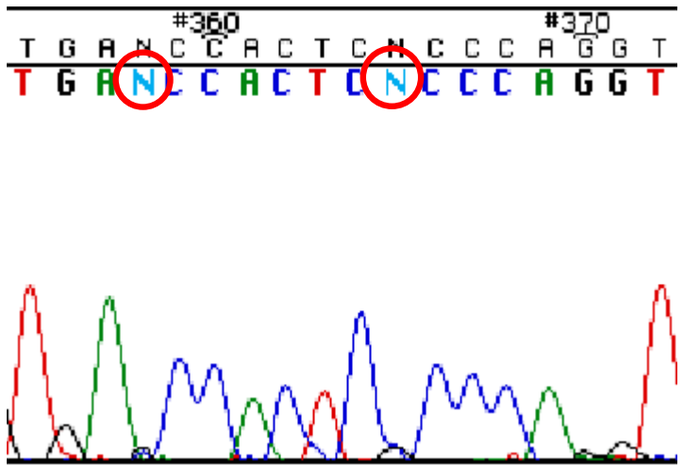


Probabilistic Strings: Scientific Applications



- The International Union of Pure and Applied Chemistry has issued the following convention for naming ambiguous portions of a sequence.
  - A = adenine
  - C = cytosine
  - G = guanine
  - T = thymine
  - R = G A (purine)
  - Y = T C (pyrimidine)
  - K = G T (keto)
  - M = A C (amino)
  - S = G C (strong bonds)
  - W = A T (weak bonds)
  - B = G T C (all but A)
  - D = G A T (all but C)
  - H = A C T (all but G)
  - V = G C A (all but T)
  - N = A G C T (any)

Probabilistic Strings: Scientific Applications



- The International Union of Pure and Applied Chemistry has issued the following convention for naming ambiguous portions of a sequence.
  - A = adenine
  - C = cytosine
  - G = guanine
  - T = thymine
  - R = G A (purine)
  - Y = T C (pyrimidine)
  - K = G T (keto)
  - M = A C (amino)
  - S = G C (strong bonds)
  - W = A T (weak bonds)
  - B = G T C (all but A)
  - D = G A T (all but C)
  - H = A C T (all but G)
  - V = G C A (all but T)
  - N = A G C T (any)

Character-Level Probabilistic Model

DNA Sequence	Probability
TATTTCGTTGCTGCTGCGGAGC	0.75
TATATCGTTGCGGCTGCGGAGC	0.10
TATTTCGTTGCGGCTGCGGAGC	0.05
TATATCGTTGCTGCTGCGGAGC	0.10

Character-Level Probabilistic Model

DNA Sequence	Probability
TATTTCGTTGCTGCTGCGGAGC	0.75
TATATCGTTGCGGCTGCGGAGC	0.10
TATTTCGTTGCGGCTGCGGAGC	0.05
TATATCGTTGCTGCTGCGGAGC	0.10

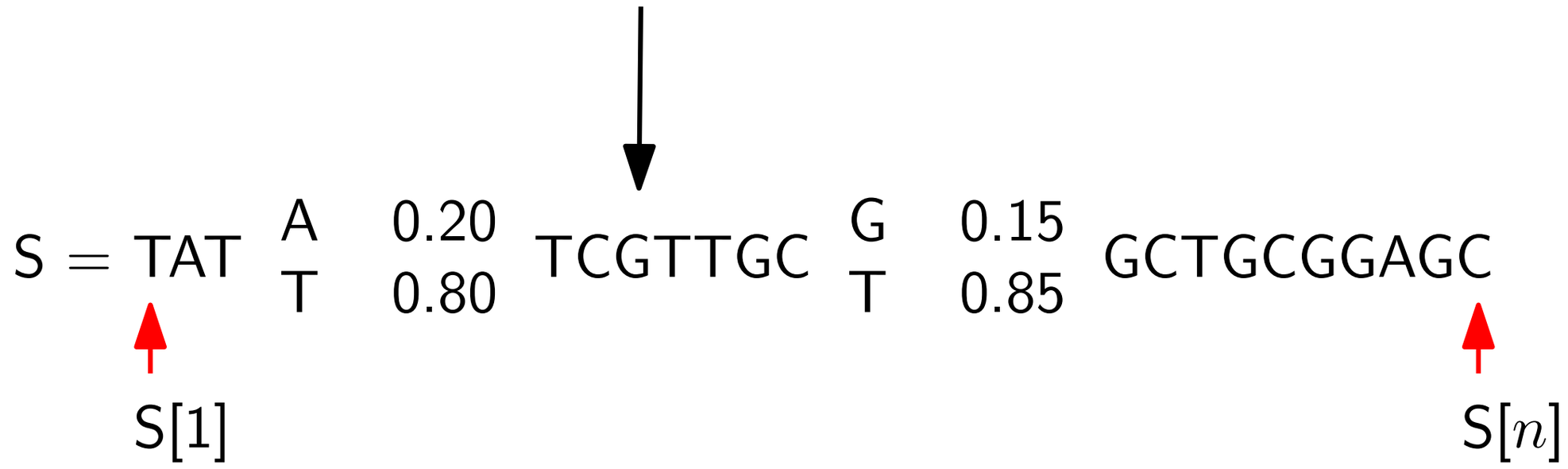
Character-Level Probabilistic Model

DNA Sequence	Probability
TATTTCGTTGCTGCTGCGGAGC	0.75
TATATCGTTGCGGCTGCGGAGC	0.10
TATTTCGTTGCGGCTGCGGAGC	0.05
TATATCGTTGCTGCTGCGGAGC	0.10

$$S = \text{TAT} \begin{array}{ll} A & 0.20 \\ T & 0.80 \end{array} \text{TCGTTGC} \begin{array}{ll} G & 0.15 \\ T & 0.85 \end{array} \text{GCTGCGGAGC}$$

Character-Level Probabilistic Model

DNA Sequence	Probability
TATTTCGTTGCTGCTGCGGAGC	0.75
TATATCGTTGCGGCTGCGGAGC	0.10
TATTTCGTTGCGGCTGCGGAGC	0.05
TATATCGTTGCTGCTGCGGAGC	0.10



Character-Level Probabilistic Model

DNA Sequence	Probability
TATTTCGTTGCTGCTGCGGAGC	0.75
TATATCGTTGCGGCTGCGGAGC	0.10
TATTTCGTTGCGGCTGCGGAGC	0.05
TATATCGTTGCTGCTGCGGAGC	0.10

$$S = \text{TAT} \begin{array}{|ll} \text{A} & 0.20 \\ \text{T} & 0.80 \end{array} \text{TCGTTGC} \begin{array}{ll} \text{G} & 0.15 \\ \text{T} & 0.85 \end{array} \text{GCTGCGGAGC}$$

$$S[4] = \{(A, 0.20), (T, 0.80)\}$$

Character-Level Probabilistic Model

DNA Sequence	Probability
TATTTCGTTGCTGCTGCGGAGC	0.75
TATATCGTTGCGGCTGCGGAGC	0.10
TATTTCGTTGCGGCTGCGGAGC	0.05
TATATCGTTGCTGCTGCGGAGC	0.10

$$\text{S} = \text{TAT} \begin{matrix}\text{A} & 0.20 \\ \text{T} & 0.80\end{matrix} \text{TCGTTGC} \begin{matrix}\text{G} & 0.15 \\ \text{T} & 0.85\end{matrix} \text{GCTGCGGAGC}$$

$S = S[1] \ldots S[n]$

where $S[i] = \{(c_{i,1}, p_{i,1}), \ldots, (c_{i,\eta_i}, p_{i,\eta_i})\}$

$c_{i,j} \in \Sigma$, $p_{i,j} \in (0, 1]$ and $\sum_{j=1}^{\eta_i} p_{i,j} = 1$.

Character-Level Probabilistic Model

- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time

Character-Level Probabilistic Model

- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time
- Consider the '_' wildcard in SQL

Character-Level Probabilistic Model

- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time
- Consider the '_' wildcard in SQL

“advis_r”

Character-Level Probabilistic Model

- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time
- Consider the '_' wildcard in SQL

“advis_r”

“adviser” “advisor” “advisur”

Character-Level Probabilistic Model

- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time
- Consider the '_' wildcard in SQL

“advis_r”

“adviser” “advisor” “advisur”

Character-Level Probabilistic Model

- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time
- Consider the '_' wildcard in SQL

"advis_r"

"adviser" ✓ "advisor" ✓ "advisur"

Character-Level Probabilistic Model

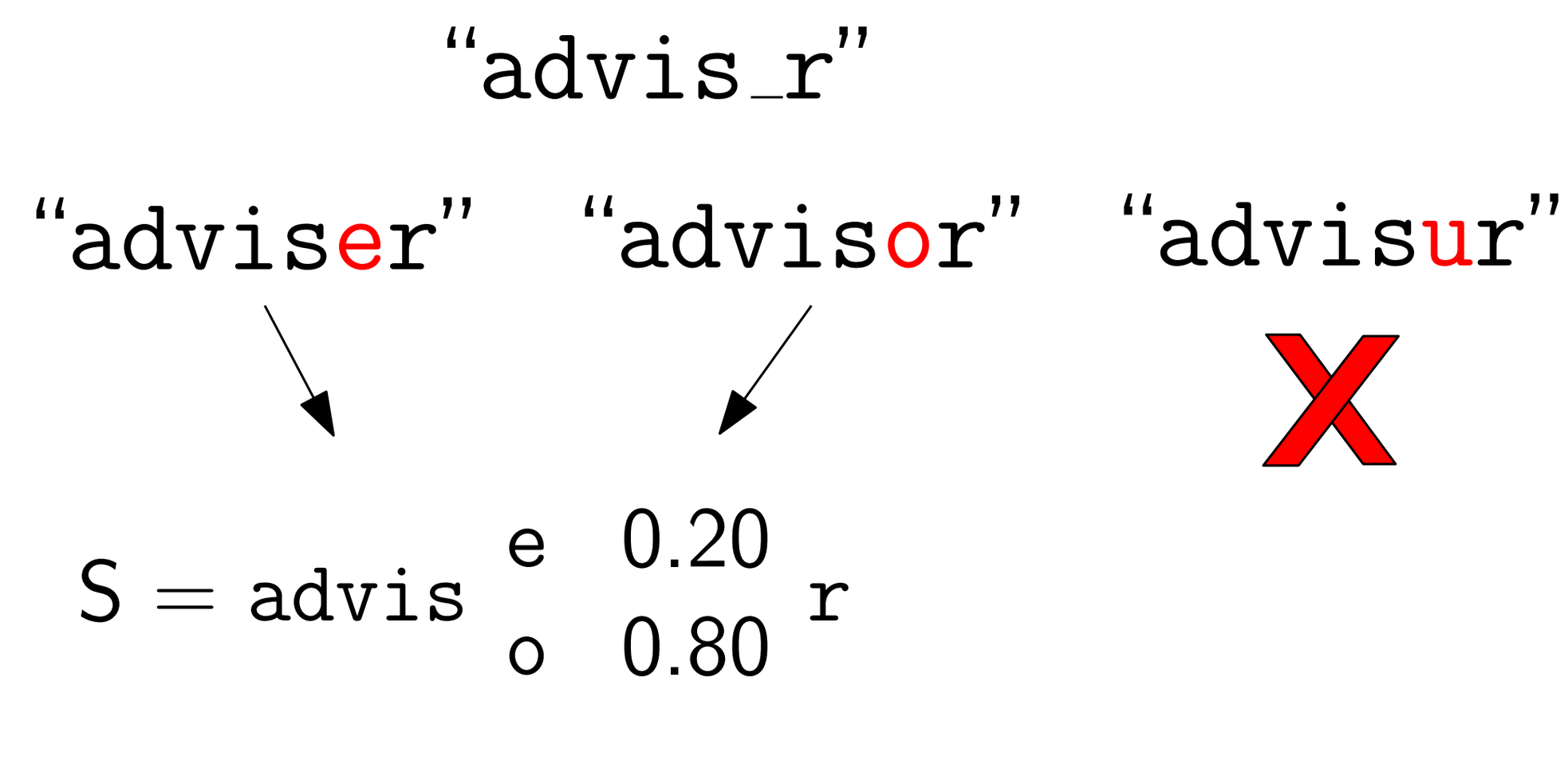
- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time
- Consider the '_' wildcard in SQL

“advis_r”

“adviser” ✓ “advisor” ✓ “advisur” ✗

Character-Level Probabilistic Model

- Uncertainty at specific positions in a string has been supported in traditional relational databases for a long time
- Consider the '_' wildcard in SQL



Edit Distance: A Classical Metric for Deterministic String Similarity Join

- The edit distance between two strings is the minimum number of *insertions* , *deletions* , and *substitutions* necessary to transform one string into the other

Edit Distance: A Classical Metric for Deterministic String Similarity Join

- The edit distance between two strings is the minimum number of *insertions* , *deletions* , and *substitutions* necessary to transform one string into the other
- Consider theater and theatre:
  - The edit distance is $d((theater), (theatre)) = 2$

Edit Distance: A Classical Metric for Deterministic String Similarity Join

- The edit distance between two strings is the minimum number of *insertions* , *deletions* , and *substitutions* necessary to transform one string into the other

- Consider theater and theatre:
  - The edit distance is $d((theater), (theatre)) = 2$

  How can we measure the similarity between probabilistic strings?

Edit Distance: A Classical Metric for Deterministic String Similarity Join

- The edit distance between two strings is the minimum number of *insertions* , *deletions* , and *substitutions* necessary to transform one string into the other

- Consider theater and theatre:
  - The edit distance is $d((theater), (theatre)) = 2$

  How can we measure the similarity between probabilistic strings?

  Key idea: Use the expected edit distance!

Possible Worlds Semantics

S_1	
$\sigma_{1,i}$	$p_{1,i}$
cat	0.50
kitty	0.50

S_2	
$\sigma_{2,i}$	$p_{2,i}$
dog	0.75
doggy	0.10
puppy	0.15

Possible Worlds Semantics

S_1	
$\sigma_{1,i}$	$p_{1,i}$
cat	0.50
kitty	0.50

S_2	
$\sigma_{2,i}$	$p_{2,i}$
dog	0.75
doggy	0.10
puppy	0.15

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

Possible Worlds Semantics

S_1	
$\sigma_{1,i}$	$p_{1,i}$
cat	0.50
kitty	0.50

S_2	
$\sigma_{2,i}$	$p_{2,i}$
dog	0.75
doggy	0.10
puppy	0.15

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

Possible Worlds Semantics

S_1	
$\sigma_{1,i}$	$p_{1,i}$
cat	0.50
kitty	0.50

S_2	
$\sigma_{2,i}$	$p_{2,i}$
dog	0.75
doggy	0.10
puppy	0.15

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

Possible Worlds Semantics

S_1	
$\sigma_{1,i}$	$p_{1,i}$
cat	0.50
kitty	0.50

S_2	
$\sigma_{2,i}$	$p_{2,i}$
dog	0.75
doggy	0.10
puppy	0.15

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

Possible Worlds Semantics

S_1	
$\sigma_{1,i}$	$p_{1,i}$
cat	0.50
kitty	0.50

S_2	
$\sigma_{2,i}$	$p_{2,i}$
dog	0.75
doggy	0.10
puppy	0.15

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + 0.075 \cdot 5 + 0.375 \cdot 5 + 0.05 \cdot 4 + 0.075 \cdot 4$$

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + 0.075 \cdot 5 + 0.375 \cdot 5 + 0.05 \cdot 4 + 0.075 \cdot 4$$

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + 0.075 \cdot 5 + 0.375 \cdot 5 + 0.05 \cdot 4 + 0.075 \cdot 4$$

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + \boxed{0.075 \cdot 5} + 0.375 \cdot 5 + 0.05 \cdot 4 + 0.075 \cdot 4$$

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + 0.075 \cdot 5 + \boxed{0.375 \cdot 5} + 0.05 \cdot 4 + 0.075 \cdot 4$$

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + 0.075 \cdot 5 + 0.375 \cdot 5 + \boxed{0.05 \cdot 4} + 0.075 \cdot 4$$

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + 0.075 \cdot 5 + 0.375 \cdot 5 + 0.05 \cdot 4 + 0.075 \cdot 4$$

Expected Edit Distance

$$EED = \widehat{d}(S_1, S_2) = \sum_{s \in \Omega} w(s) \cdot d(s).$$

Ω					
$\sigma_{1,i}$	$p_{1,i}$	$\sigma_{2,j}$	$p_{2,j}$	w(s)	$d(\sigma_{1,i}, \sigma_{2,j})$
cat	0.50	dog	0.75	0.375	3
cat	0.50	doggy	0.10	0.05	5
cat	0.50	puppy	0.15	0.075	5
kitty	0.50	dog	0.75	0.375	5
kitty	0.50	doggy	0.10	0.05	4
kitty	0.50	puppy	0.15	0.075	4

$$\widehat{d}(S_1, S_2) = 0.375 \cdot 3 + 0.05 \cdot 5 + 0.075 \cdot 5 + 0.375 \cdot 5 + 0.05 \cdot 4 + 0.075 \cdot 4$$

$$\widehat{d}(S_1, S_2) = 4.125$$

Probabilistic String Similiarity Join

R	
id	S
1	{ (Microsoft, 0.90), (Microsoft Inc., 0.10) }
2	{ (Yahoo, 0.80), (Yahoo!, 0.20) }
⋮	⋮

$\bowtie_S$

T	
id	S
1	(Google, 1)
2	{ (AT&T, 0.98), (ATT, 0.02) }
⋮	⋮

Probabilistic String Similiarity Join

R	
id	S
1	{ (Microsoft, 0.90), (Microsoft Inc., 0.10) }
2	{ (Yahoo, 0.80), (Yahoo!, 0.20) }
⋮	⋮

$\bowtie_S$

T	
id	S
1	(Google, 1)
2	{ (AT&T, 0.98), (ATT, 0.02) }
⋮	⋮

- A join on R and T on probabilistic string attribute S returns all pairs of records (r_i, t_j) s.t. $r_i \in R$, $t_j \in T$ and $\widehat{d}(r_i.S, t_j.S) \leq \tau$

Probabilistic String Similiarity Join

R	
id	S
1	{ (Microsoft, 0.90), (Microsoft Inc., 0.10) }
2	{ (Yahoo, 0.80), (Yahoo!, 0.20) }
⋮	⋮

$\bowtie_S$

T	
id	S
1	(Google, 1)
2	{ (AT&T, 0.98), (ATT, 0.02) }
⋮	⋮

- A join on R and T on probabilistic string attribute S returns all pairs of records (r_i, t_j) s.t. $r_i \in R$, $t_j \in T$ and $\widehat{d}(r_i.S, t_j.S) \leq \tau$
- Our goal is to find efficient means to optimize a probabilistic string join in existing relational databases

Relational Representation for String-Level Model

a **string-level probabilistic** q**-gram** is a quadruple (i, p, ℓ, g)

i is the choice index (cid) $\quad$ ℓ is the start position

p is the choice probability $\quad$ g is the q-gram starting at ℓ

Relational Representation for String-Level Model

a **string-level probabilistic** q**-gram** is a quadruple (i, p, ℓ, g)

i is the choice index (cid) $\quad$ ℓ is the start position

p is the choice probability $\quad$ g is the q-gram starting at ℓ

probabilistic strings

id	S
1	{(add, 0.8), (plus, 0.2)}
2	{(up, 0.9), (op, 0.1)}

relational representation

id	cid	A	p	len
1	1	add	0.8	3.2
1	2	plus	0.2	3.2
2	1	up	0.9	2
2	2	op	0.1	2

q-grams

id	cid	ℓ	g
1	1	1	#a
1	1	2	ad
1	1	3	dd
1	1	4	d$
1	2	1	#p
1	2	2	pl
1	2	3	lu
⋮	⋮	⋮	⋮

Relational Representation for String-Level Model

a **string-level probabilistic q-gram** is a quadruple (i, p, ℓ, g)

i is the choice index (cid) $\quad$ ℓ is the start position

p is the choice probability $\quad$ g is the q-gram starting at ℓ

probabilistic strings

id	S
1	{(add, 0.8), (plus, 0.2)}
2	{(up, 0.9), (op, 0.1)}

relational representation

id	cid	A	p	len
1	1	add	0.8	3.2
1	2	plus	0.2	3.2
2	1	up	0.9	2
2	2	op	0.1	2

q-grams

id	cid	ℓ	g
1	1	1	#a
1	1	2	ad
1	1	3	dd
1	1	4	d$
1	2	1	#p
1	2	2	pl
1	2	3	lu
⋮	⋮	⋮	⋮

Relational Representation for String-Level Model

a **string-level probabilistic q-gram** is a quadruple (i, p, ℓ, g)

i is the choice index (cid) $\quad$ ℓ is the start position

p is the choice probability $\quad$ g is the q-gram starting at ℓ

probabilistic strings

id	S
1	{(add, 0.8), (plus, 0.2)}
2	{(up, 0.9), (op, 0.1)}

relational representation

id	cid	A	p	len
1	1	add	0.8	3.2
1	2	plus	0.2	3.2
2	1	up	0.9	2
2	2	op	0.1	2

q-grams

id	cid	ℓ	g
1	1	1	#a
1	1	2	ad
1	1	3	dd
1	1	4	d$
1	2	1	#p
1	2	2	pl
1	2	3	lu
⋮	⋮	⋮	⋮

Relational Representation for String-Level Model

a **string-level probabilistic q-gram** is a quadruple (i, p, ℓ, g)

i is the choice index (cid) $\quad$ ℓ is the start position

p is the choice probability $\quad$ g is the q-gram starting at ℓ

probabilistic strings

id	S
1	{(add, 0.8), (plus, 0.2)}
2	{(up, 0.9), (op, 0.1)}

relational representation

id	cid	A	p	len
1	1	add	0.8	3.2
1	2	plus	0.2	3.2
2	1	up	0.9	2
2	2	op	0.1	2

q-grams

id	cid	ℓ	g
1	1	1	#a
1	1	2	ad
1	1	3	dd
1	1	4	d$
1	2	1	#p
1	2	2	pl
1	2	3	lu
⋮	⋮	⋮	⋮

Relational Representation for String-Level Model

a **string-level probabilistic q-gram** is a quadruple (i, p, ℓ, g)

i is the choice index (cid) $\quad$ ℓ is the start position

p is the choice probability $\quad$ g is the q-gram starting at ℓ

probabilistic strings

id	S
1	{(add, 0.8), (plus, 0.2)}
2	{(up, 0.9), (op, 0.1)}

relational representation

id	cid	A	p	len
1	1	add	0.8	3.2
1	2	plus	0.2	3.2
2	1	up	0.9	2
2	2	op	0.1	2

q-grams

id	cid	ℓ	g
1	1	1	#a
1	1	2	ad
1	1	3	dd
1	1	4	d$
1	2	1	#p
1	2	2	pl
1	2	3	lu
⋮	⋮	⋮	⋮

Relational Representation for String-Level Model

a **string-level probabilistic q-gram** is a quadruple (i, p, ℓ, g)

i is the choice index (cid) $\quad$ ℓ is the start position

p is the choice probability $\quad$ g is the q-gram starting at ℓ

probabilistic strings

id	S
1	{(add, 0.8), (plus, 0.2)}
2	{(up, 0.9), (op, 0.1)}

relational representation

id	cid	A	p	len
1	1	add	0.8	3.2
1	2	plus	0.2	3.2
2	1	up	0.9	2
2	2	op	0.1	2

q-grams

id	cid	ℓ	g
1	1	1	#a
1	1	2	ad
1	1	3	dd
1	1	4	d$
1	2	1	#p
1	2	2	pl
1	2	3	lu
⋮	⋮	⋮	⋮

String-Level Probabilistic String Similiarity Join

R				
id	cid	S	p	len
1	1	Microsoft	0.90	9.5
1	2	Microsoft Inc.	0.10	9.5
2	1	Yahoo	0.80	5.2
2	2	Yahoo!	0.20	5.2
⋮	⋮	⋮	⋮	⋮

T				
id	cid	S	p	len
1	1	Google	1	6
2	1	AT&T	0.98	4.1
2	2	AT&T Labs	0.02	4.1
⋮	⋮	⋮	⋮	⋮

String-Level Probabilistic String Similiarity Join

R				
id	cid	S	p	len
1	1	Microsoft	0.90	9.5
1	2	Microsoft Inc.	0.10	9.5
2	1	Yahoo	0.80	5.2
2	2	Yahoo!	0.20	5.2
⋮	⋮	⋮	⋮	⋮

T				
id	cid	S	p	len
1	1	Google	1	6
2	1	AT&T	0.98	4.1
2	2	AT&T Labs	0.02	4.1
⋮	⋮	⋮	⋮	⋮

- S-BJ:
  SELECT R.id,T.id FROM R,T GROUP BY R.id,T.id
  HAVING SUM(R.p*T.p*d(R.A,T.A)) $\leq$ τ

String-Level Probabilistic String Similiarity Join

R				
id	cid	S	p	len
1	1	Microsoft	0.90	9.5
1	2	Microsoft Inc.	0.10	9.5
2	1	Yahoo	0.80	5.2
2	2	Yahoo!	0.20	5.2
⋮	⋮	⋮	⋮	⋮

T				
id	cid	S	p	len
1	1	Google	1	6
2	1	AT&T	0.98	4.1
2	2	AT&T Labs	0.02	4.1
⋮	⋮	⋮	⋮	⋮

- S-BJ:
  `SELECT R.id,T.id FROM R,T GROUP BY R.id,T.id`
  `HAVING SUM(R.p*T.p*d(R.A,T.A))` $\leq \tau$

  Too expensive!!!!

String-Level Probabilistic String Similiarity Join

- Solution: We derive probabilistic q-gram based lower-bound pruning techniques.

String-Level Probabilistic String Similiarity Join

- Solution: We derive probabilistic q-gram based lower-bound pruning techniques.
- Our techniques are implementable in existing relational databases with pure SQL and are several of magnitude more efficient than S-BJ.

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7|T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7|T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7|T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7|T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7|T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7|T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7|T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8 C0.2⋆⋆G0.7 T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8 C0.2⋆⋆G0.7 T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7 T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7 T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Relational Representation for Character-Level Model

A **character-level probabilistic q-gram** is a pair $(\ell, S[\ell..\ell+q-1])$
ℓ is the beginning position of the q-gram
$S[\ell..\ell+q-1]$ is the *probabilistic substring* $S[\ell]\ldots S[\ell+q-1]$

probabilistic strings

id	S
1	{(A, 0.8), (C,0.2)}, {(G, 0.7), (T, 0.3)}
2	{(A, 1)}, {(G, 0.6), (T,0.4)}
3	{(C, 1)}, {(A, 1)}, {(G, 1)}

relational representation

id	A	len
1	⋆A0.8|C0.2⋆⋆G0.7 T0.3⋆	2
2	A⋆G0.6|T0.4⋆	2
3	CAG	3

q-grams

id	p	ℓ	g
1	0.80	1	#A
1	0.20	1	#C
1	0.56	2	AG
1	0.24	2	AT
1	0.14	2	CG
1	0.06	2	CT
1	0.70	3	G$
1	0.30	3	T$
⋮	⋮	⋮	⋮

Character-Level Probabilistic String Similarity Join

R		
id	A	len
1	AT ⋆A0.8|C0.2⋆GAT	6
1	AT ⋆C0.2|G0.3|T0.5⋆CG	5
⋮	⋮	⋮

T		
id	A	len
1	T ⋆G0.95|T0.05⋆TAC	5
1	⋆G0.3|T0.2|A0.5⋆CGTA	5
⋮	⋮	⋮

Character-Level Probabilistic String Similarity Join

R		
id	A	len
1	AT ⋆A0.8|C0.2⋆GAT	6
1	AT ⋆C0.2|G0.3|T0.5⋆CG	5
⋮	⋮	⋮

T		
id	A	len
1	T ⋆G0.95|T0.05⋆TAC	5
1	⋆G0.3|T0.2|A0.5⋆CGTA	5
⋮	⋮	⋮

- C-BJ:
  `SELECT R.id,T.id FROM R,T WHERE ed(R.A,T.A)` $\leq \tau$

Character-Level Probabilistic String Similarity Join

R		
id	A	len
1	AT ⋆A0.8|C0.2⋆GAT	6
1	AT ⋆C0.2|G0.3|T0.5⋆CG	5
⋮	⋮	⋮

T		
id	A	len
1	T ⋆G0.95|T0.05⋆TAC	5
1	⋆G0.3|T0.2|A0.5⋆CGTA	5
⋮	⋮	⋮

- C-BJ:
  `SELECT R.id,T.id FROM R,T WHERE ed(R.A,T.A)` $\leq \tau$

  Too expensive!!!!

Character-Level Probabilistic String Similarity Join

- Solution: We derive probabilistic q-gram lower and upper-bound pruning techniques.

Character-Level Probabilistic String Similarity Join

- Solution: We derive probabilistic q-gram lower and upper-bound pruning techniques.
- Our techniques are implementable in existing relational databases with pure SQL and are orders of magnitude more efficient than C-BJ.

Character-Level Probabilistic String Similarity Join

- Solution: We derive probabilistic q-gram lower and upper-bound pruning techniques.
- Our techniques are implementable in existing relational databases with pure SQL and are orders of magnitude more efficient than C-BJ.
- We also derive DP-based lower and upper-bound pruning techniques to further improve query efficiency.

Experiments: Setup

- Experimental setup
  - Microsoft SQL Server 2008 Enterprise Edition
  - Windows XP
  - Intel Xeon E5405 @ 2.00 GHz
  - 2GB memory

Experiments: Algorithms

- String-Level
  - String Basic Join (S-BJ): Pure SQL
  - String Prune Join (S-PJ): Positional, needs UDF
  - String Prune Join 2 (S-PJ2): Positional, Pure SQL

Experiments: Algorithms

- String-Level
  - String Basic Join (S-BJ): Pure SQL
  - String Prune Join (S-PJ): Positional, needs UDF
  - String Prune Join 2 (S-PJ2): Positional, Pure SQL
- Character-Level
  - Character Basic Join (C-BJ)
  - Character q-Gram Prune Join (C-qPJ): Uses all q-gram upper and lower bounds
  - Character Prune Join (C-PJ): Uses all q-gram and DP upper and lower bounds

Experiments: Data Sets

- We create 2 datasets for string-level and 2 for character-level models respectively based on 3 real data sources:
    - Author Names from DBLP
    - Category-Link string field from Wikipedia category-link table
    - Genome sequence database from the Rhodococcus project.

Experiments: Data Sets

- We create 2 datasets for string-level and 2 for character-level models respectively based on 3 real data sources:
  - Author Names from DBLP
  - Category-Link string field from Wikipedia category-link table
  - Genome sequence database from the Rhodococcus project.
- String-level model:
  - Author dataset *(Author1)*
  - Category-Link dataset *(Category)*

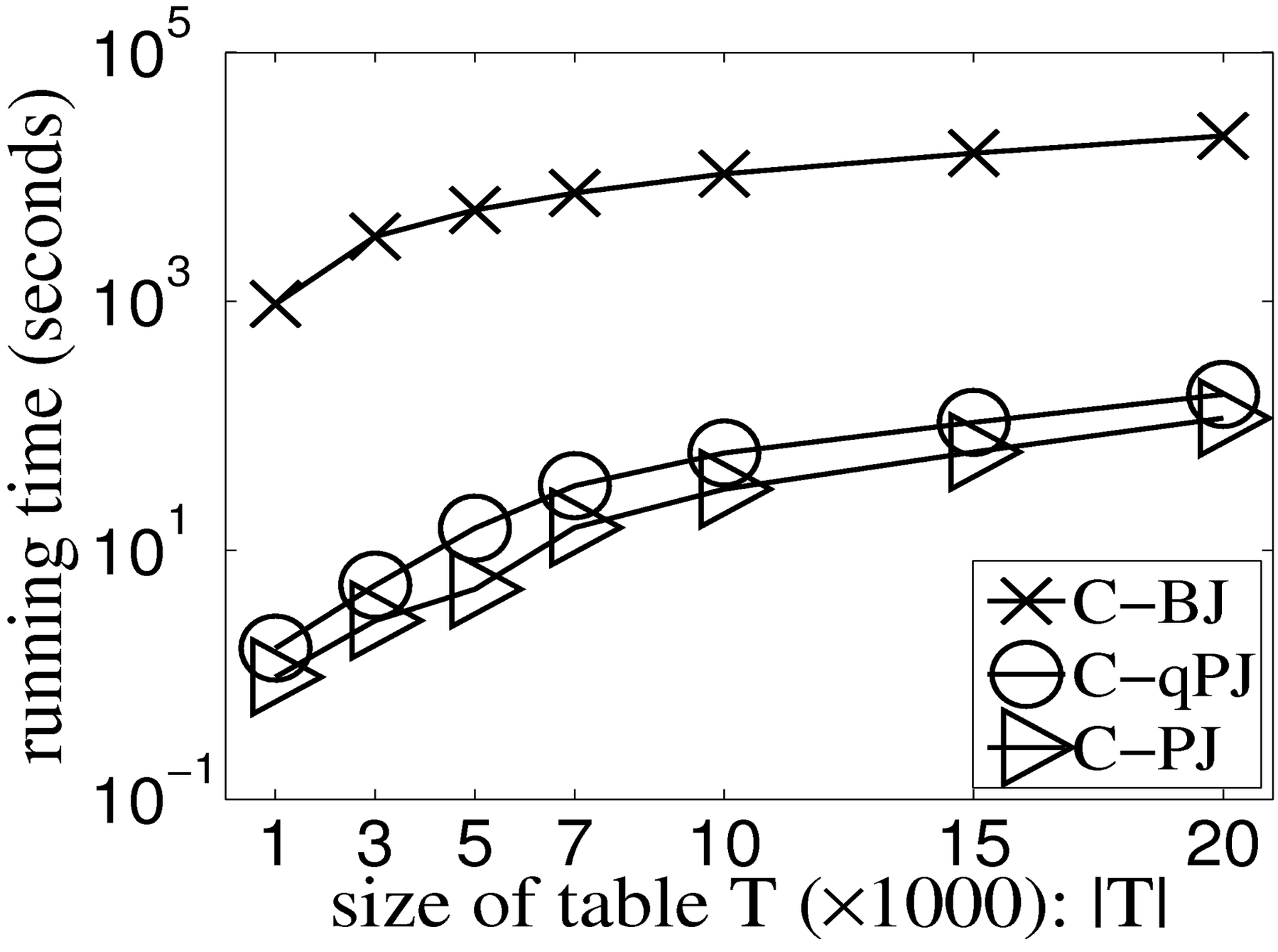
Experiments: Data Sets

- We create 2 datasets for string-level and 2 for character-level models respectively based on 3 real data sources:
  - Author Names from DBLP
  - Category-Link string field from Wikipedia category-link table
  - Genome sequence database from the Rhodococcus project.
- Character-level model:
  - Author dataset *(Author2)*
  - Genome dataset *(Genome)*

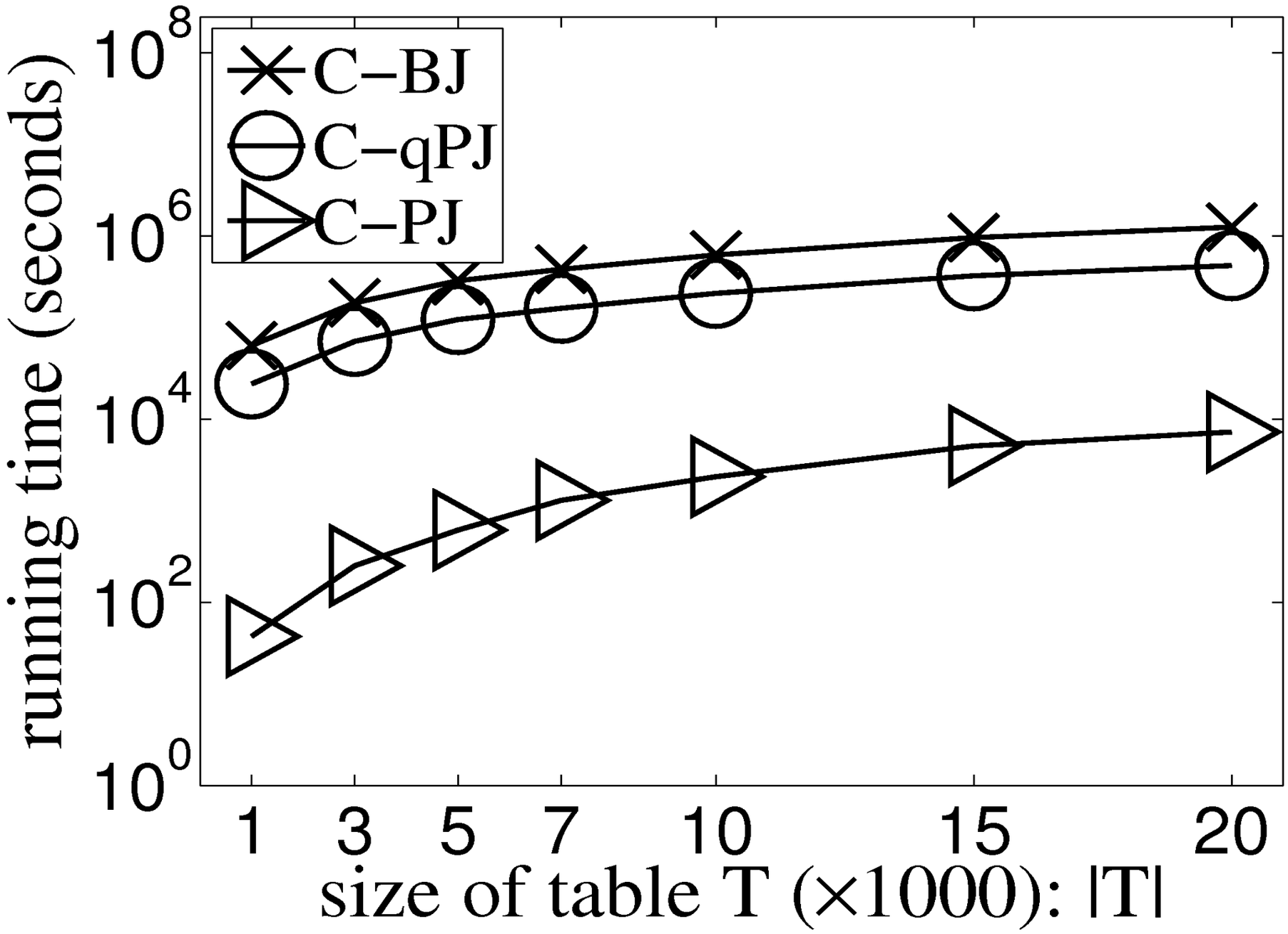
Experiments: Character-Level Default Parameters

Symbol	Definition	Default Value
χ	max(S[i])	6
θ	Max percent uncertain positions	20%
ω	Self-concatentations	0
μ_C	Average string length	Author2 14.3 Genome 14.9
$\|R\|$	Size table R	1,000
$\|T\|$	Size table T	10,000
q	q-gram	2
τ	$\textsc{Eed}$ threshold	2

Character-level model, *Author2* dataset



Character-level model, *Genome* dataset



Conclusions

- We study the problem of efficient string joins in probabilistic string databases utilizing the expected edit distance

Conclusions

- We study the problem of efficient string joins in probabilistic string databases utilizing the expected edit distance
- We introduce efficient and effective techniques for both string-level and character-level models which are implementable in existing relational databases

Conclusions

- We study the problem of efficient string joins in probabilistic string databases utilizing the expected edit distance
- We introduce efficient and effective techniques for both string-level and character-level models which are implementable in existing relational databases
- There are many interesting unexplored problems including methods to derive a representation in either model from a data source, indexing probabilisitc strings for similarity search, selectivity estimations, and keyword searches in probabilistic string databases

Conclusions

- We study the problem of efficient string joins in probabilistic string databases utilizing the expected edit distance
- We introduce efficient and effective techniques for both string-level and character-level models which are implementable in existing relational databases
- There are many interesting unexplored problems including methods to derive a representation in either model from a data source, indexing probabilisitc strings for similarity search, selectivity estimations, and keyword searches in probabilistic string databases
- Other similarity measures other than the edit distance may be desirable, such as Jaccard's Distance

The End

THANK YOU

Q and A

- The entire source code is available from a link at

  http://ww2.cs.fsu.edu/~jestes

Edit Distance: A Classical Metric for Deterministic String Similarity Join

$$d[i,j] = \min \begin{cases} d[i,j-1]+1, & \text{insertion} \\ d[i-1,j]+1, & \text{deletion} \\ d[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]), & \text{substitution} \end{cases}$$

in which

$$c(\sigma_1[i],\sigma_2[j]) = \begin{cases} 1, \sigma_1[i] \neq \sigma_2[j], \\ 0, \sigma_1[i] = \sigma_2[j], \end{cases}$$

Edit Distance: A Classical Metric for Deterministic String Similarity Join

$$d[i,j] = \min \begin{cases} d[i,j-1]+1, & \text{insertion} \\ d[i-1,j]+1, & \text{deletion} \\ d[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]), & \text{substitution} \end{cases}$$

in which

$$c(\sigma_1[i],\sigma_2[j]) = \begin{cases} 1, \sigma_1[i] \neq \sigma_2[j], \\ 0, \sigma_1[i] = \sigma_2[j], \end{cases}$$

Note that $d(\sigma_1,\sigma_2) = d[|\sigma_1|,|\sigma_2|]$

Edit Distance: A Classical Metric for Deterministic String Similarity Join

$$d[i,j] = \min \begin{cases} d[i,j-1]+1, & \text{insertion} \\ d[i-1,j]+1, & \text{deletion} \\ d[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]), & \text{substitution} \end{cases}$$

in which

$$c(\sigma_1[i],\sigma_2[j]) = \begin{cases} 1, \sigma_1[i] \neq \sigma_2[j], \\ 0, \sigma_1[i] = \sigma_2[j], \end{cases}$$

Note that $d(\sigma_1,\sigma_2) = d[|\sigma_1|,|\sigma_2|]$

How can we measure the similarity between probabilistic strings?

Edit Distance: A Classical Metric for Deterministic String Similarity Join

$$d[i,j] = \min \begin{cases} d[i,j-1]+1, & \text{insertion} \\ d[i-1,j]+1, & \text{deletion} \\ d[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]), & \text{substitution} \end{cases}$$

in which

$$c(\sigma_1[i],\sigma_2[j]) = \begin{cases} 1, \sigma_1[i] \neq \sigma_2[j], \\ 0, \sigma_1[i] = \sigma_2[j], \end{cases}$$

Note that $d(\sigma_1,\sigma_2) = d[|\sigma_1|,|\sigma_2|]$

Key idea: Use the expected edit distance!

Background: Naive Deterministic String Similarity Join

R	
id	A
1	sony
2	Microsoft
3	Toshiba
⋮	⋮

$$\bowtie_{d(A,B)\leq\tau}$$

T	
id	B
1	Micrsft
2	Sony
3	Apple
⋮	⋮

Background: Naive Deterministic String Similarity Join

R	
id	A
1	sony
2	Microsoft
3	Toshiba
⋮	⋮

$\bowtie_{d(A,B)\leq\tau}$

T	
id	B
1	Micrsft
2	Sony
3	Apple
⋮	⋮

SELECT R.id, T.id FROM R, T WHERE d(A,B)$\leq\tau$

Background: Naive Deterministic String Similarity Join

R	
id	A
1	sony
2	Microsoft
3	Toshiba
⋮	⋮

$\bowtie_{d(A,B)\leq\tau}$

T	
id	B
1	Micrsft
2	Sony
3	Apple
⋮	⋮

`SELECT R.id, T.id FROM R, T WHERE d(A,B)` $\leq \tau$

The query results in a nested loops join invoking d(A,B) for every pair of $R.A$, $T.B$

Background: Naive Deterministic String Similarity Join

R	
id	A
1	sony
2	Microsoft
3	Toshiba
⋮	⋮

$\bowtie_{d(A,B)\leq\tau}$

T	
id	B
1	Micrsft
2	Sony
3	Apple
⋮	⋮

`SELECT R.id, T.id FROM R, T WHERE d(A,B)` $\leq \tau$

The query results in a nested loops join invoking d(A,B) for every pair of $R.A$, $T.B$

Too expensive!!!

Background: Improving Deterministic String Similarity Join Using q-Grams

- q-grams are strings of length q produced by sliding a window of length q over the characters of a string σ

Background: Improving Deterministic String Similarity Join Using q-Grams

- q-grams are strings of length q produced by sliding a window of length q over the characters of a string σ
- It has been shown in prior research that utilizing q-grams may greatly improve the efficiency of string similarity joins.

Background: Improving Deterministic String Similarity Join Using q-Grams

- q-grams are strings of length q produced by sliding a window of length q over the characters of a string σ
- It has been shown in prior research that utilizing q-grams may greatly improve the efficiency of string similarity joins.
  - The RDBMS can perform pruning on tuples though the use of q-grams

Background: Improving Deterministic String Similarity Join Using q-Grams

- As an example of how to generate q-grams, consider generating 2-grams for the string *advisor*

  $\sigma =$ "advisor"

Background: Improving Deterministic String Similarity Join Using q-Grams

- As an example of how to generate q-grams, consider generating 2-grams for the string *advisor*

σ = "advisor"

σ = "#advisor$"

Background: Improving Deterministic String Similarity Join Using q-Grams

- As an example of how to generate q-grams, consider generating 2-grams for the string *advisor*

σ = "advisor"

↓

σ = "#advisor$"

↓

$G_\sigma = \{(1, \#a), (2, ad), (3, dv), (4, vi), (5, is), (6, so), (7, or), (8, r\$)\}$

Background: Improving Deterministic String Similarity Join Using q-Grams

- As an example of how to generate q-grams, consider generating 2-grams for the string *advisor*

σ = "advisor"

↓

σ = "#advisor$"

↓

$G_\sigma = \{(1, \#a), (2, ad), (3, dv), (4, vi), (5, is), (6, so), (7, or), (8, r\$)\}$

- A q-gram is a pair (ℓ, g)
  ℓ is the beginning position of the q-gram
  g is the substring of length q in σ beginning from ℓ.

Background: Improving Deterministic String Similarity Join Using q-Grams

- Given $q_1 = (\ell_1, g_1)$ and $q_2 = (\ell_2, g_2)$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Given $q_1 = (\ell_1, g_1)$ and $q_2 = (\ell_2, g_2)$
  - $q_1 = q_2$ if $g_1 = g_2$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Given $q_1 = (\ell_1, g_1)$ and $q_2 = (\ell_2, g_2)$
  - $q_1 = q_2$ if $g_1 = g_2$
  - $G_{\sigma_1} \cap G_{\sigma_2} = \{(q_1, q_2) \mid q_1 = q_2, q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}\}$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Given $q_1 = (\ell_1, g_1)$ and $q_2 = (\ell_2, g_2)$
  - $q_1 = q_2$ if $g_1 = g_2$
  - $G_{\sigma_1} \cap G_{\sigma_2} = \{(q_1, q_2) \mid q_1 = q_2, q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}\}$
  - $q_1 \overset{k}{=} q_2$ if $g_1 = g_2$ and $|\ell_1 - \ell_2| \leq k$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Given $q_1 = (\ell_1, g_1)$ and $q_2 = (\ell_2, g_2)$
  - $q_1 = q_2$ if $g_1 = g_2$
  - $G_{\sigma_1} \cap G_{\sigma_2} = \{(q_1, q_2) \mid q_1 = q_2, q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}\}$
- $q_1 \stackrel{k}{=} q_2$ if $g_1 = g_2$ and $|\ell_1 - \ell_2| \leq k$
- $G_{\sigma_1} \cap_k G_{\sigma_2} = \{(q_1, q_2) \mid q_1 \stackrel{k}{=} q_2, q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}\}$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Prior work has shown:

Background: Improving Deterministic String Similarity Join Using q-Grams

- Prior work has shown:

  Lemma 1 (q-gram filtering) *For two strings σ_1 and σ_2,*

$$|G_{\sigma_1} \cap G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Prior work has shown:

  Lemma 1 (q-gram filtering) *For two strings σ_1 and σ_2,*

$$|G_{\sigma_1} \cap G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Prior work has shown:

  Lemma 1 (q-gram filtering) *For two strings σ_1 and σ_2,*

$$|G_{\sigma_1} \cap G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  Lemma 2 (positional q-gram filtering) *For strings σ_1 and σ_2,*

$$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Prior work has shown:

  Lemma 1 (q-gram filtering) *For two strings σ_1 and σ_2,*

  $$|G_{\sigma_1} \cap G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  Lemma 2 (positional q-gram filtering) *For strings σ_1 and σ_2,*

  $$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  Assuming $d(\sigma_1, \sigma_2) \leq \tau$ we get

  $$|G_{\sigma_1} \cap_{\tau} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

Background: Improving Deterministic String Similarity Join Using q-Grams

- Prior work has shown:

  Lemma 1 (q-gram filtering) *For two strings σ_1 and σ_2,*

  $$|G_{\sigma_1} \cap G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  Lemma 2 (positional q-gram filtering) *For strings σ_1 and σ_2,*

  $$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  Assuming $d(\sigma_1, \sigma_2) \leq \tau$ we get

  $$|G_{\sigma_1} \cap_{\tau} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  Lemma 3 (length filtering) $d(\sigma_1, \sigma_2) \geq ||\sigma_1| - |\sigma_2||$

String-Level Probabilistic String Similiarity Join

- The set of probabilistic q-grams for $S(i)$ is $G_{S(i)} = \{(i, p_i, \ell_j, g_j)\}$, for all j s.t. $(\ell_j, g_j) \in G_{\sigma_i}$

String-Level Probabilistic String Similiarity Join

- The set of probabilistic q-grams for $S(i)$ is $G_{S(i)} = \{(i, p_i, \ell_j, g_j)\}$, for all j s.t. $(\ell_j, g_j) \in G_{\sigma_i}$
- $G_S = G_{S(1)} \cup \cdots \cup G_{S(m)}$

String-Level Probabilistic String Similiarity Join

- The set of probabilistic q-grams for $S(i)$ is $G_{S(i)} = \{(i, p_i, \ell_j, g_j)\}$, for all j s.t. $(\ell_j, g_j) \in G_{\sigma_i}$
- $G_S = G_{S(1)} \cup \cdots \cup G_{S(m)}$
- $\rho_1 = \{i, p_i, \ell_x, g_x\}$ from G_{S_1} and $\rho_2 = \{j, p_j, \ell_y, g_y\}$ from G_{S_2}

String-Level Probabilistic String Similiarity Join

- The set of probabilistic q-grams for $S(i)$ is $G_{S(i)} = \{(i, p_i, \ell_j, g_j)\}$, for all j s.t. $(\ell_j, g_j) \in G_{\sigma_i}$
- $G_S = G_{S(1)} \cup \cdots \cup G_{S(m)}$
- $\rho_1 = \{i, p_i, \ell_x, g_x\}$ from G_{S_1} and $\rho_2 = \{j, p_j, \ell_y, g_y\}$ from G_{S_2}
  - $\rho_1 = \rho_2$ if $g_x = g_y$

String-Level Probabilistic String Similiarity Join

- The set of probabilistic q-grams for $S(i)$ is $G_{S(i)} = \{(i, p_i, \ell_j, g_j)\}$, for all j s.t. $(\ell_j, g_j) \in G_{\sigma_i}$
- $G_S = G_{S(1)} \cup \cdots \cup G_{S(m)}$
- $\rho_1 = \{i, p_i, \ell_x, g_x\}$ from G_{S_1} and $\rho_2 = \{j, p_j, \ell_y, g_y\}$ from G_{S_2}
  - $\rho_1 = \rho_2$ if $g_x = g_y$
  - $\rho_1 \overset{k}{=} \rho_2$ if $g_x = g_y$ and $|\ell_x - \ell_y| \leq k$

String-Level Probabilistic String Similiarity Join

- The set of probabilistic q-grams for $S(i)$ is $G_{S(i)} = \{(i, p_i, \ell_j, g_j)\}$, for all j s.t. $(\ell_j, g_j) \in G_{\sigma_i}$
- $G_S = G_{S(1)} \cup \cdots \cup G_{S(m)}$
- $\rho_1 = \{i, p_i, \ell_x, g_x\}$ from G_{S_1} and $\rho_2 = \{j, p_j, \ell_y, g_y\}$ from G_{S_2}
  - $\rho_1 = \rho_2$ if $g_x = g_y$
  - $\rho_1 \overset{k}{=} \rho_2$ if $g_x = g_y$ and $|\ell_x - \ell_y| \leq k$
  - $G_{S_1} \cap G_{S_2} = \{(\rho_1, \rho_2) \mid \rho_1 = \rho_2, \rho_1 \in G_{S_1}, \rho_2 \in G_{S_2}\}$

String-Level Probabilistic String Similiarity Join

- The set of probabilistic q-grams for $S(i)$ is $G_{S(i)} = \{(i, p_i, \ell_j, g_j)\}$, for all j s.t. $(\ell_j, g_j) \in G_{\sigma_i}$
- $G_S = G_{S(1)} \cup \cdots \cup G_{S(m)}$
- $\rho_1 = \{i, p_i, \ell_x, g_x\}$ from G_{S_1} and $\rho_2 = \{j, p_j, \ell_y, g_y\}$ from G_{S_2}
  - $\rho_1 = \rho_2$ if $g_x = g_y$
  - $\rho_1 \overset{k}{=} \rho_2$ if $g_x = g_y$ and $|\ell_x - \ell_y| \leq k$
  - $G_{S_1} \cap G_{S_2} = \{(\rho_1, \rho_2) \mid \rho_1 = \rho_2, \rho_1 \in G_{S_1}, \rho_2 \in G_{S_2}\}$
  - $G_{S_1} \cap_k G_{S_2} = \{(\rho_1, \rho_2) \mid \rho_1 \overset{k}{=} \rho_2, \rho_1 \in G_{S_1}, \rho_2 \in G_{S_2}\}$

String-Level Probabilistic Strings: Length Filtering

- We may directly extend the **length filtering** lemma

$$d(\sigma_1, \sigma_2) \geq ||\sigma_1| - |\sigma_2||$$

to obtain,

Lemma 4 *For any string-level probabilistic strings S_1 and S_2:*

$$\widehat{d}(S_1, S_2) \geq \sum_{s \in \Omega} w(s) ||\sigma_{1,i}| - |\sigma_{2,j}||.$$

String-Level Probabilistic Strings: Length Filtering

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq \sum_{s \in \Omega} w(s) ||\sigma_{1,i}| - |\sigma_{2,j}||.$$

String-Level Probabilistic Strings: Length Filtering

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq \sum_{s \in \Omega} w(s) ||\sigma_{1,i}| - |\sigma_{2,j}||.$$

- ```
  1  SELECT R.id,T.id FROM R,T
  2  GROUP BY R.id, T.id
  3  HAVING SUM(R.p*T.p*ABS(|R.A|-|T.A|))≤ τ
  ```
  ```

String-Level Probabilistic Strings: q-Gram Lower Bounds

- Inspired by deterministic **q-gram filtering**,

$$d(\sigma_1, \sigma_2) \geq 1 + \frac{\max(|\sigma_1|, |\sigma_2|)}{q} - \frac{|G_{\sigma_1} \cap G_{\sigma_2}| + 1}{q}$$

we can derive a similar result for string-level probabilistic q-grams,

Theorem 1 *For any string-level probabilistic strings S_1 and S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1, \rho_2) \in G_{S_1} \cap G_{S_2}} p(\rho_1) p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- Inspired by deterministic **q-gram filtering**,

$$d(\sigma_1, \sigma_2) \geq 1 + \frac{\max(|\sigma_1|, |\sigma_2|)}{q} - \frac{|G_{\sigma_1} \cap G_{\sigma_2}| + 1}{q}$$

we can derive a similar result for string-level probabilistic q-grams,

Theorem 1 *For any string-level probabilistic strings S_1 and S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1, \rho_2) \in G_{S_1} \cap G_{S_2}} p(\rho_1) p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- Inspired by deterministic **q-gram filtering**,

$$d(\sigma_1, \sigma_2) \geq 1 + \frac{\max(|\sigma_1|, |\sigma_2|)}{q} - \frac{|G_{\sigma_1} \cap G_{\sigma_2}| + 1}{q}$$

we can derive a similar result for string-level probabilistic q-grams,

Theorem 1 *For any string-level probabilistic strings S_1 and S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1, \rho_2) \in G_{S_1} \cap G_{S_2}} p(\rho_1) p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- The next lemma will be useful in deriving $\textsc{Eed}$ lower bounds.

Lemma 5 *For two non-negative random variables X, Y:*

$$\mathbf{E}[\max(X, Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y]).$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- The next lemma will be useful in deriving $\textsc{Eed}$ lower bounds.

  Lemma 5 *For two non-negative random variables X, Y:*

  $$\mathbf{E}[\max(X, Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y]).$$

- **Proof**

$$\mathbf{E}[\max(X, Y)] \quad = \quad \mathbf{E}\left(\frac{X + Y + |X - Y|}{2}\right)$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- The next lemma will be useful in deriving $\textsc{Ed}$ lower bounds.

  Lemma 5 *For two non-negative random variables X, Y:*

  $$\mathbf{E}[\max(X, Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y]).$$

- **Proof**

$$\begin{aligned} \mathbf{E}[\max(X, Y)] &= \mathbf{E}\left(\frac{X + Y + |X - Y|}{2}\right) \\ &= \frac{\mathbf{E}[X] + \mathbf{E}[Y]}{2} + \frac{\mathbf{E}[|X - Y|]}{2} \end{aligned}$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- The next lemma will be useful in deriving $\textsc{Eed}$ lower bounds.

  Lemma 5 *For two non-negative random variables X, Y:*

  $$\mathbf{E}[\max(X,Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y]).$$

- **Proof**

$$\begin{aligned}
\mathbf{E}[\max(X,Y)] &= \mathbf{E}\left(\frac{X+Y+|X-Y|}{2}\right) \\
&= \frac{\mathbf{E}[X]+\mathbf{E}[Y]}{2} + \frac{\mathbf{E}[|X-Y|]}{2} \\
&\geq \frac{\mathbf{E}[X]+\mathbf{E}[Y]}{2} + \frac{\mathbf{E}[X-Y]}{2} = \mathbf{E}[X].
\end{aligned}$$

Similarly, $\mathbf{E}[\max(X,Y)] \geq \mathbf{E}[Y]$. $\square$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

- **Proof** Let $s \in \Omega$ and $s = (S_1(i), S_2(j))$.
  By the basic **q-gram filtering** lemma we have,

$$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

$$\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

$$\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

  $$\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$$

- Since $\sum_{s\in\Omega} w(s) = 1$, the RHS equals

$$\begin{aligned} \sum_{s\in\Omega} \quad & w(s)\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q\sum_{s\in\Omega} w(s)d(s) + q \\ = \quad & \mathbf{E}[\max(|S_1|, |S_2|)] - 1 - q \cdot \widehat{d}(S_1, S_2) + q \end{aligned}$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

$$\boxed{\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).}$$

- Since $\sum_{s\in\Omega} w(s) = 1$, the RHS equals

$$\begin{aligned} \sum_{s\in\Omega} \quad & w(s)\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q\sum_{s\in\Omega} w(s)d(s) + q \\ = \quad & \mathbf{E}[\max(|S_1|, |S_2|)] - 1 - q \cdot \widehat{d}(S_1, S_2) + q \end{aligned}$$

$$\mathbf{E}[\max(X, Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y])$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

  $$\boxed{\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).}$$

- Since $\sum_{s\in\Omega} w(s) = 1$, the RHS equals

$$\sum_{s\in\Omega} \quad w(s)\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q\sum_{s\in\Omega} w(s)d(s) + q$$

$$= \quad \boxed{\mathbf{E}[\max(|S_1|, |S_2|)]} - 1 - q \cdot \widehat{d}(S_1, S_2) + q$$

$$\mathbf{E}[\max(X, Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y])$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

$$\boxed{\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).}$$

- Since $\sum_{s\in\Omega} w(s) = 1$, the RHS equals

$$\begin{aligned}
\sum_{s\in\Omega} \quad & w(s)\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q\sum_{s\in\Omega} w(s)d(s) + q \\
= \quad & \boxed{\mathbf{E}[\max(|S_1|, |S_2|)]} - 1 - q \cdot \widehat{d}(S_1, S_2) + q \\
\geq \quad & \max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - 1 - q(\widehat{d}(S_1, S_2) - 1),
\end{aligned}$$

$$\mathbf{E}[\max(X, Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y])$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

  $$\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$$

- Since $\sum_{s\in\Omega} w(s) = 1$, the RHS equals

$$\begin{aligned} \sum_{s\in\Omega} \quad & w(s)\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q\sum_{s\in\Omega} w(s)d(s) + q \\ = \quad & \mathbf{E}[\max(|S_1|, |S_2|)] - 1 - q \cdot \widehat{d}(S_1, S_2) + q \\ \geq \quad & \max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - 1 - q(\widehat{d}(S_1, S_2) - 1), \end{aligned}$$

$$\mathbf{E}[\max(X, Y)] \geq \max(\mathbf{E}[X], \mathbf{E}[Y])$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$$

  $$\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$

  $\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$

- The LHS is

$$\begin{aligned} \textstyle\sum_{s\in\Omega} w(s) \quad & |G_{S_1(i)} \cap G_{S_2(j)}| \\ = \quad & \sum_{s\in\Omega} \sum_{(\rho_1,\rho_2)\in G_{S_1(i)}\cap G_{S_2(j)}} w(s) \end{aligned}$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$

  $\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$

- The LHS is

$$
\begin{aligned}
\textstyle\sum_{s\in\Omega} w(s) \quad & |G_{S_1(i)} \cap G_{S_2(j)}| \\
= \quad & \sum_{s\in\Omega} \sum_{(\rho_1,\rho_2)\in G_{S_1(i)}\cap G_{S_2(j)}} w(s) \\
= \quad & \sum_{s\in\Omega} \sum_{(\rho_1,\rho_2)\in G_{S_1(i)}\cap G_{S_2(j)}} p(\rho_1)p(\rho_2)
\end{aligned}
$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- $\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| \geq$

  $\sum_{s\in\Omega} w(s)(\max(|\sigma_{1,i}|, |\sigma_{2,j}|) - 1 - q(d(s) - 1)).$

- The LHS is

$$
\begin{aligned}
\textstyle\sum_{s\in\Omega} w(s) \quad & |G_{S_1(i)} \cap G_{S_2(j)}| \\
= & \sum_{s\in\Omega} \sum_{(\rho_1,\rho_2)\in G_{S_1(i)}\cap G_{S_2(j)}} w(s) \\
= & \sum_{s\in\Omega} \sum_{(\rho_1,\rho_2)\in G_{S_1(i)}\cap G_{S_2(j)}} p(\rho_1)p(\rho_2) \\
= & \sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2). \quad \square
\end{aligned}
$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

-
```
1  SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
2  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
3     AND R_q.cid=R.cid AND T_q.cid=T.cid
4  GROUP BY R.id, T.id, R.len, T.len
5  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
```

String-Level Probabilistic Strings: q-Gram Lower Bounds

- **Theorem 1** *For any string-level probabilistic strings S_1 and S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- **Theorem 1** *For any string-level probabilistic strings S_1 and S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

- ```
1 SELECT R.id AS rid, T.id AS tid FROM R,T,Rq,Tq
2 WHERE Rq.g=Tq.g AND Rq.id=R.id AND Tq.id=T.id
3   AND Rq.cid=R.cid AND Tq.cid=T.cid
4 GROUP BY R.id, T.id, R.len, T.len
5 HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
```
```

String-Level Probabilistic Strings: q-Gram Lower Bounds

- **Theorem 1** *For any string-level probabilistic strings S_1 and S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

- ```
  1  SELECT R.id AS rid, T.id AS tid FROM R,T,Rq,Tq
  2  WHERE Rq.g=Tq.g AND Rq.id=R.id AND Tq.id=T.id
  3     AND Rq.cid=R.cid AND Tq.cid=T.cid
  4  GROUP BY R.id, T.id, R.len, T.len
  5  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
  ```

- We also derive SQL lower bounds which utilize positional and length information
  ```

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Inspired by deterministic **positional q-gram filtering**,

$$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

we could attempt to derive a similar result for probabilistic strings

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Inspired by deterministic **positional q-gram filtering**,

$$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  we could attempt to derive a similar result for probabilistic strings

- However, we have a problem as $d(s)$ may be different for different worlds

$$\begin{aligned} & \sum_{s \in \Omega} w(s) |G_{S_1(i)} \cap_{d(s)} G_{S_2(j)}| \\ \geq \quad & \max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - 1 - q(\widehat{d}(S_1, S_2) - 1). \end{aligned}$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Inspired by deterministic **positional q-gram filtering**,

$$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  we could attempt to derive a similar result for probabilistic strings

- However, we have a problem as $d(s)$ may be different for different worlds

$$\begin{aligned} & \sum_{s \in \Omega} w(s) |G_{S_1(i)} \cap_{d(s)} G_{S_2(j)}| \\ \geq \quad & \max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - 1 - q(\widehat{d}(S_1, S_2) - 1). \end{aligned}$$

- We can enlarge the LHS to make it easier to compute

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Assume $\widehat{d}(S_1, S_2) \leq \tau$
  Let $\Omega' \subseteq \Omega$ be the subset of possible worlds s in which $d(s) \geq 2\tau$.
  The probability mass of Ω' is at most $1/2$. (Markov Inequality)

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Assume $\widehat{d}(S_1, S_2) \leq \tau$
  Let $\Omega' \subseteq \Omega$ be the subset of possible worlds s in which $d(s) \geq 2\tau$.
  The probability mass of Ω' is at most $1/2$. (Markov Inequality)
- $$\sum_{s \in \Omega} w(s) |G_{S_1(i)} \cap_{d(s)} G_{S_2(j)}| \leq \sum_{s \in \Omega'} w(s) |G_{S_1(i)} \cap G_{S_2(j)}| +$$
  $$\sum_{s \in \Omega \setminus \Omega'} w(s) |G_{S_1(i)} \cap_{2\tau - 1} G_{S_2(j)}|$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Assume $\widehat{d}(S_1, S_2) \leq \tau$
  Let $\Omega' \subseteq \Omega$ be the subset of possible worlds s in which $d(s) \geq 2\tau$.
  The probability mass of Ω' is at most $1/2$. (Markov Inequality)
- $$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{d(s)} G_{S_2(j)}| \leq \sum_{s\in\Omega'} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| + \sum_{s\in\Omega\setminus\Omega'} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$$

$$= \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}| + \sum_{s\in\Omega'} w(s)(|G_{S_1(i)} \cap G_{S_2(j)}| - |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|).$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

$$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{d(s)} G_{S_2(j)}| \leq \sum_{s\in\Omega'} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| + \sum_{s\in\Omega\setminus\Omega'} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$$

$$= \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}| + \sum_{s\in\Omega'} w(s)(|G_{S_1(i)} \cap G_{S_2(j)}| - |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|).$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

$$\sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{d(s)} G_{S_2(j)}| \leq \sum_{s\in\Omega'} w(s)|G_{S_1(i)} \cap G_{S_2(j)}| + \sum_{s\in\Omega\setminus\Omega'} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$$

$$= \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}| + \sum_{s\in\Omega'} w(s)(|G_{S_1(i)} \cap G_{S_2(j)}| - |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|).$$

- Let $x(s) = |G_{S_1(i)} \cap G_{S_2(j)}| - |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$

  Then we may cast it as a fractional knapsack as:
  Choose $s \in \Omega'$ s.t. $\sum_{s\in\Omega'} w(s) \leq \frac{1}{2}$
  Maximize $\sum_{s\in\Omega'} w(s)x(s)$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Fractional Knapsack Formulation:
  $x(s) = |G_{S_1(i)} \cap G_{S_2(j)}| - |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$
  Item value : $w(s)x(s)$
  Item weight : $w(s)$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Fractional Knapsack Formulation:
  $x(s) = |G_{S_1(i)} \cap G_{S_2(j)}| - |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$
  Item value : $w(s)x(s)$
  Item weight : $w(s)$

  Initialize $\Omega' = \emptyset$.
  Sort $s \in \Omega$ by decreasing $\frac{w(s)x(s)}{x(s)} = x(s)$
  While $\sum_{s \in \Omega'} w(s) > 1/2$
  $\Omega' \leftarrow s \in \Omega$ with next largest $x(s)$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Fractional Knapsack Formulation:
  $x(s) = |G_{S_1(i)} \cap G_{S_2(j)}| - |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$
  Item value : $w(s)x(s)$
  Item weight : $w(s)$

  Initialize $\Omega' = \emptyset$.
  Sort $s \in \Omega$ by decreasing $\frac{w(s)x(s)}{x(s)} = x(s)$
  While $\sum_{s\in\Omega'} w(s) > 1/2$
  $\Omega' \leftarrow s \in \Omega$ with next largest $x(s)$

- Let s' be the last world added to Ω',

$$\mathrm{UB}_\tau = \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}| + \sum_{s\in\Omega'} w(s)x(s) - w(s')x(s')\frac{\sum_{s\in\Omega'} w(s) - 1/2}{w(s')}$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Theorem 2** *For any string-level probabilistic strings S_1 and S_2,*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

where,

$$\begin{aligned}\mathrm{UB}_\tau &= \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}| \\ &\quad + \sum_{s\in\Omega'} w(s)x(s) - w(s')x(s')\frac{\sum_{s\in\Omega'} w(s) - 1/2}{w(s')}.\end{aligned}$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Theorem 2** *For any string-level probabilistic strings S_1 and S_2,*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

where,

$$\begin{aligned} \mathrm{UB}_\tau &= \sum_{s \in \Omega} w(s) |G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}| \\ &\quad + \sum_{s \in \Omega'} w(s)x(s) - w(s')x(s') \frac{\sum_{s \in \Omega'} w(s) - 1/2}{w(s')}. \end{aligned}$$

- UB_τ requires a UDF to compute,
  `ub(`$\mathtt{R}_q$`.cid, R.p, `$\mathtt{R}_q$`.`ℓ`, `$\mathtt{R}_q$`.g, `$\mathtt{T}_q$`.cid, T.p, `$\mathtt{T}_q$`.`ℓ`, `$\mathtt{T}_q$`.g, `τ`)`

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

- S-PJ:

```
1  SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
2  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
3     AND R_q.cid=R.cid AND T_q.cid=T.cid
4  GROUP BY R.id, T.id, R.len, T.len
5     HAVING 1 + 1/q*( max(R.len,T.len) - 1 -
6     ub(R_q.cid,R.p,R_q.ℓ,R_q.g,T_q.cid,T.p,T_q.ℓ,T_q.g,τ))≤ τ
```

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- $$\mathrm{UB}_\tau = \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}| + \sum_{s\in\Omega'} w(s)x(s) - w(s')x(s')\frac{\sum_{s\in\Omega'} w(s) - 1/2}{w(s')}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- $$\mathrm{UB}_\tau = \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$$

$$+ \sum_{s\in\Omega'} w(s)x(s) - w(s')x(s')\frac{\sum_{s\in\Omega'} w(s) - 1/2}{w(s')}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

Loose!!!!

- $$\mathrm{UB}_\tau = \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$$

$$+ \sum_{s\in\Omega'} w(s)x(s) - w(s')x(s')\frac{\sum_{s\in\Omega'} w(s) - 1/2}{w(s')}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

Loose!!!!

- $$\mathrm{UB}_\tau = \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$$
$$+ \sum_{s\in\Omega'} w(s)x(s) - w(s')x(s')\frac{\sum_{s\in\Omega'} w(s) - 1/2}{w(s')}.$$

- **Observation.** For most cases if $\widehat{d}(S_1, S_2) \leq \tau$,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap_{2\tau} G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

Loose!!!!

- $$\mathrm{UB}_\tau = \sum_{s\in\Omega} w(s)|G_{S_1(i)} \cap_{2\tau-1} G_{S_2(j)}|$$
$$+ \sum_{s\in\Omega'} w(s)x(s) - w(s')x(s')\frac{\sum_{s\in\Omega'} w(s) - 1/2}{w(s')}.$$

- **Observation.** For most cases if $\widehat{d}(S_1, S_2) \leq \tau$,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1} \cap_{2\tau} G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Observation.** For most cases if $\widehat{d}(S_1, S_2) \leq \tau$,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap_{2\tau} G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Observation.** For most cases if $\widehat{d}(S_1, S_2) \leq \tau$,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap_{2\tau} G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

- S-PJ2:

```
1  SELECT R.id AS rid, T.id AS tid FROM R,T,Rq,Tq
2  WHERE Rq.g=Tq.g AND ABS(Rq.ℓ-Tq.ℓ)≤ 2τ
3     AND Rq.id=R.id AND Tq.id=T.id
4     AND Rq.cid=R.cid AND Tq.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
```

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Observation.** For most cases if $\widehat{d}(S_1, S_2) \leq \tau$,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap_{2\tau} G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

- S-PJ2:

```
1  SELECT R.id AS rid, T.id AS tid FROM R,T,Rq,Tq
2  WHERE Rq.g=Tq.g AND ABS(Rq.ℓ-Tq.ℓ)≤ 2τ
3      AND Rq.id=R.id AND Tq.id=T.id
4      AND Rq.cid=R.cid AND Tq.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
```

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$
- $\Pr(\gamma_1 = \gamma_2) = \Pr(S_1[i..i+q-1] = S_2[j..j+q-1])$

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$
- $\Pr(\gamma_1 = \gamma_2) = \Pr(S_1[i..i+q-1] = S_2[j..j+q-1])$
  if $(\gamma_1, \gamma_2) \in G_{S_1,S_2}^d$
  $\Pr(\gamma_1 = \gamma_2) = 1$ if $S_1[i..i+q-1] = S_2[j..j+q-1]$
  0 otherwise

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$
- $\Pr(\gamma_1 = \gamma_2) = \Pr(S_1[i..i+q-1] = S_2[j..j+q-1])$
  if $(\gamma_1, \gamma_2) \in G_{S_1,S_2}^d$
  $\Pr(\gamma_1 = \gamma_2) = 1$ if $S_1[i..i+q-1] = S_2[j..j+q-1]$
  0 otherwise
- $\mathrm{off}(\gamma_1, \gamma_2)$ represents the difference in position of two q-grams

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$
- $\Pr(\gamma_1 = \gamma_2) = \Pr(S_1[i..i+q-1] = S_2[j..j+q-1])$
  if $(\gamma_1, \gamma_2) \in G_{S_1,S_2}^d$
  $\Pr(\gamma_1 = \gamma_2) = 1$ if $S_1[i..i+q-1] = S_2[j..j+q-1]$
  0 otherwise
- $\mathrm{off}(\gamma_1, \gamma_2)$ represents the difference in position of two q-grams

$\gamma_1 = (1, \{(\#, 1)\}\{(A, 0.8), (C, 0.2)\})$
$\gamma_2 = (1, \{(\#, 1)\}\{(A, 1)\}).$

$\Pr(\gamma_1 = \gamma_2) = 0.8$
$\mathrm{off}(\gamma_1, \gamma_2) = 0$
$\gamma_2 \in G_{S_2}^d$
$\gamma_1 \notin G_{S_1}^d$

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$
- $\Pr(\gamma_1 = \gamma_2) = \Pr(S_1[i..i+q-1] = S_2[j..j+q-1])$

  if $(\gamma_1, \gamma_2) \in G_{S_1,S_2}^d$

  $\Pr(\gamma_1 = \gamma_2) = 1$ if $S_1[i..i+q-1] = S_2[j..j+q-1]$

  0 otherwise
- $\text{off}(\gamma_1, \gamma_2)$ represents the difference in position of two q-grams

$\gamma_1 = (1, \{(\#, 1)\}\{(A, 0.8), (C, 0.2)\})$
$\gamma_2 = (1, \{(\#, 1)\}\{(A, 1)\}).$

$\Pr(\gamma_1 = \gamma_2) = 0.8$
$\text{off}(\gamma_1, \gamma_2) = 0$
$\gamma_2 \in G_{S_2}^d$
$\gamma_1 \notin G_{S_1}^d$

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$
- $\Pr(\gamma_1 = \gamma_2) = \Pr(S_1[i..i+q-1] = S_2[j..j+q-1])$
  if $(\gamma_1, \gamma_2) \in G_{S_1,S_2}^d$
  $\Pr(\gamma_1 = \gamma_2) = 1$ if $S_1[i..i+q-1] = S_2[j..j+q-1]$
  0 otherwise
- $\mathrm{off}(\gamma_1, \gamma_2)$ represents the difference in position of two q-grams

$\gamma_1 = (1, \{(\#, 1)\}\{(A, 0.8), (C, 0.2)\})$
$\gamma_2 = (1, \{(\#, 1)\}\{(A, 1)\})$.

$\Pr(\gamma_1 = \gamma_2) = 0.8$
$\mathrm{off}(\gamma_1, \gamma_2) = 0$
$\gamma_2 \in G_{S_2}^d$
$\gamma_1 \notin G_{S_1}^d$

Character-Level Probabilistic String Similarity Join

- The set of deterministic q-grams are denoted G_S^d.
- $G_{S_1,S_2}^d = G_{S_1}^d \times G_{S_2}^d$ and $G_{S_1,S_2} = G_{S_1} \times G_{S_2}$
- $\Pr(S_1 = S_2) = \sum_{s \in \Omega, \sigma_1 = \sigma_2} w(s)$.
- $\gamma_1 = (i, S_1[i..i+q-1])$ and $\gamma_2 = (j, S_2[j..j+q-1])$
- $\Pr(\gamma_1 = \gamma_2) = \Pr(S_1[i..i+q-1] = S_2[j..j+q-1])$
  if $(\gamma_1, \gamma_2) \in G_{S_1,S_2}^d$
  $\Pr(\gamma_1 = \gamma_2) = 1$ if $S_1[i..i+q-1] = S_2[j..j+q-1]$
  0 otherwise
- $\mathrm{off}(\gamma_1, \gamma_2)$ represents the difference in position of two q-grams

$\gamma_1 = (1, \{(\#, 1)\}\{(A, 0.8), (C, 0.2)\})$
$\gamma_2 = (1, \{(\#, 1)\}\{(A, 1)\})$.

$\Pr(\gamma_1 = \gamma_2) = 0.8$
$\mathrm{off}(\gamma_1, \gamma_2) = 0$
$\gamma_2 \in G_{S_2}^d$
$\gamma_1 \notin G_{S_1}^d$

Character-Level Probabilistic Strings: Length Filtering

- We may directly extend the **length filtering** lemma

$$d(\sigma_1, \sigma_2) \geq ||\sigma_1| - |\sigma_2||$$

to obtain,

Lemma 6 *For any character-level probabilistic strings* S_1, S_2,

$$\widehat{d}(S_1, S_2) \geq ||S_1| - |S_2||.$$

Character-Level Probabilistic Strings: Length Filtering

- For any character-level probabilistic strings S_1, S_2,

$$\widehat{d}(S_1, S_2) \geq ||S_1| - |S_2||.$$

Character-Level Probabilistic Strings: Length Filtering

- For any character-level probabilistic strings S_1, S_2,

$$\widehat{d}(S_1, S_2) \geq ||S_1| - |S_2||.$$

- ```
  1   SELECT R.id AS rid,T.id AS tid FROM R,T
  2   WHERE ABS(R.len - T.len) ≤ τ
  ```
  ```

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Inspired by deterministic **q-gram filtering**,

$$d(\sigma_1, \sigma_2) \geq 1 + \frac{\max(|\sigma_1|, |\sigma_2|)}{q} - \frac{|G_{\sigma_1} \cap G_{\sigma_2}| + 1}{q}$$

we can derive a similar result for character-level probabilistic q-grams,

Theorem 3 *For any character-level probabilistic strings S_1, S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Inspired by deterministic **q-gram filtering**,

$$d(\sigma_1, \sigma_2) \geq 1 + \frac{\max(|\sigma_1|, |\sigma_2|)}{q} - \frac{|G_{\sigma_1} \cap G_{\sigma_2}| + 1}{q}$$

we can derive a similar result for character-level probabilistic q-grams,

Theorem 3 *For any character-level probabilistic strings S_1, S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Inspired by deterministic **q-gram filtering**,

$$d(\sigma_1, \sigma_2) \geq 1 + \frac{\max(|\sigma_1|, |\sigma_2|)}{q} - \frac{|G_{\sigma_1} \cap G_{\sigma_2}| + 1}{q}$$

we can derive a similar result for character-level probabilistic q-grams,

Theorem 3 *For any character-level probabilistic strings S_1, S_2:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

```
1  SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
2  WHERE R_q.g=T_q.g AND R.id=R_q.id AND T.id=T_q.id
3  GROUP BY R.id, T.id, R.len, T.len
4  HAVING 1 + max(R.len,T.len)/q -
5   SUM( T_q.p*R_q.p)/q - 1/q ≤ τ
```

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Inspired by deterministic **postional q-gram filtering**,

$$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

we may attempt to derive a similar result for probabilistic strings

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Inspired by deterministic **postional q-gram filtering**,

$$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  we may attempt to derive a similar result for probabilistic strings

- However, we have a problem as $d(s)$ may be different for different worlds

$$\sum_{s \in \Omega} w(s)|G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}| \geq \max(|S_1|, |S_2|) - 1 - q(\widehat{d}(S_1, S_2) - 1).$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- Inspired by deterministic **postional q-gram filtering**,

$$|G_{\sigma_1} \cap_{d(\sigma_1,\sigma_2)} G_{\sigma_2}| \geq \max(|\sigma_1|, |\sigma_2|) + q - 1 - qd(\sigma_1, \sigma_2)$$

  we may attempt to derive a similar result for probabilistic strings

- However, we have a problem as $d(s)$ may be different for different worlds

$$\sum_{s \in \Omega} w(s)|G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}| \geq \max(|S_1|, |S_2|) - 1 - q(\widehat{d}(S_1, S_2) - 1).$$

- We can enlarge the LHS to make it easier to compute

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- However, we have a problem as $d(s)$ may be different for different worlds

$$\sum_{s \in \Omega} w(s) |G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}| \geq \max(|S_1|, |S_2|) - 1 - q(\widehat{d}(S_1, S_2) - 1).$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- However, we have a problem as $d(s)$ may be different for different worlds

$$\sum_{s\in\Omega} w(s)|G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}| \geq \max(|S_1|,|S_2|) - 1 - q(\widehat{d}(S_1,S_2)-1).$$

- **Lemma 7** *For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1,S_2)$:*

$$\sum_{s\in\Omega} w(s)|G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}| \leq \sum_{(\gamma_1,\gamma_2)\in(G_{S_1,S_2}-G^d_{S_1,S_2})} \Pr(\gamma_1=\gamma_2) + \sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2},\gamma_1=\gamma_2} \min\left(1, \frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- However, we have a problem as $d(s)$ may be different for different worlds

$$\sum_{s\in\Omega} w(s)|G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}| \geq \max(|S_1|,|S_2|)-1-q(\widehat{d}(S_1,S_2)-1).$$

- **Lemma 7** *For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1,S_2)$:*

$$\sum_{s\in\Omega} w(s)|G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}| \leq \sum_{(\gamma_1,\gamma_2)\in(G_{S_1,S_2}-G^d_{S_1,S_2})} \Pr(\gamma_1=\gamma_2) + \sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2},\gamma_1=\gamma_2} \min\left(1,\frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Proof:** First let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  $\gamma_1 = (i, S_1[i..i+q-1]) \in G_{S_1}$
  $\gamma_2 = (j, S_2[j..j+q-1]) \in G_{S_2}$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Proof:** First let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  $\gamma_1 = (i, S_1[i..i+q-1]) \in G_{S_1}$
  $\gamma_2 = (j, S_2[j..j+q-1]) \in G_{S_2}$

- We will need,

$$\Pr(d(s) \geq \text{off}(\gamma_1, \gamma_2)) \leq \min\left(1, \frac{\mathbf{E}(d(s))}{\text{off}(\gamma_1, \gamma_2)}\right)$$

$$= \min\left(1, \frac{\widehat{d}(S_1, S_2)}{\text{off}(\gamma_1, \gamma_2)}\right) \leq \min\left(1, \frac{\tau}{\text{off}(\gamma_1, \gamma_2)}\right),$$

where we define $\frac{1}{\text{off}(\gamma_1, \gamma_2)} = \infty$ if $\text{off}(\gamma_1, \gamma_2) = 0$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Proof:** First let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  $\gamma_1 = (i, S_1[i..i+q-1]) \in G_{S_1}$
  $\gamma_2 = (j, S_2[j..j+q-1]) \in G_{S_2}$

Markov Inequality

- We will need,

$$\Pr(d(s) \geq \mathrm{off}(\gamma_1, \gamma_2)) \leq \min\left(1, \frac{\mathbf{E}(d(s))}{\mathrm{off}(\gamma_1, \gamma_2)}\right)$$

$$= \min\left(1, \frac{\widehat{d}(S_1, S_2)}{\mathrm{off}(\gamma_1, \gamma_2)}\right) \leq \min\left(1, \frac{\tau}{\mathrm{off}(\gamma_1, \gamma_2)}\right),$$

where we define $\frac{1}{\mathrm{off}(\gamma_1, \gamma_2)} = \infty$ if $\mathrm{off}(\gamma_1, \gamma_2) = 0$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Proof:** First let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  $\gamma_1 = (i, S_1[i..i+q-1]) \in G_{S_1}$
  $\gamma_2 = (j, S_2[j..j+q-1]) \in G_{S_2}$

- We also define $\gamma_1 =_s \gamma_2$ as the event that $S_1[i..i+q-1] = S_2[j..j+q-1]$ in s. We have

$$\Pr(\gamma_1 =_s \gamma_2) = \begin{cases} w(s), & \text{if } \sigma_1[i..i+q-1] = \sigma_2[j..j+q-1]; \\ 0, & \text{otherwise.} \end{cases}$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- $$\sum_{s \in \Omega} w(s) |G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}|$$

$$= \sum_{(\gamma_1, \gamma_2) \in G_{S_1, S_2}} \sum_{s \in \Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1, \gamma_2) \leq d(s))$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

$$\sum_{s\in\Omega} w(s)|G_{\sigma_1} \cap_{d(s)} G_{\sigma_2}|$$

$$= \sum_{(\gamma_1,\gamma_2)\in G_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

$$= \sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

$$+ \sum_{\substack{(\gamma_1,\gamma_2)\in \\ G_{S_1,S_2}-G^d_{S_1,S_2}}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

$$\sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

$$+ \sum_{\substack{(\gamma_1,\gamma_2)\in \\ G_{S_1,S_2}-G^d_{S_1,S_2}}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- $$\sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

  $$+ \sum_{\substack{(\gamma_1,\gamma_2)\in \\ G_{S_1,S_2}-G^d_{S_1,S_2}}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

$$\Pr(d(s) \ge \mathrm{off}(\gamma_1,\gamma_2)) \le \min\left(1, \frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

$$\sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2)\le d(s))$$

$$+ \sum_{(\gamma_1,\gamma_2)\in G_{S_1,S_2}-G^d_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2)\le d(s))$$

$$\Pr(d(s)\ge \mathrm{off}(\gamma_1,\gamma_2)) \le \min\left(1, \frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

$$\sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2}} \boxed{\sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))}$$

$$+ \sum_{(\gamma_1,\gamma_2)\in G_{S_1,S_2}-G^d_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2) \le d(s))$$

$$\le \sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2},\gamma_1=\gamma_2} \min\left(1, \frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)$$

$$+ \sum_{(\gamma_1,\gamma_2)\in (G_{S_1,S_2}-G^d_{S_1,S_2})} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2). \quad \square$$

$$\Pr(d(s) \ge \mathrm{off}(\gamma_1,\gamma_2)) \le \min\left(1, \frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

$$\sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2}} \boxed{\sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2)\le d(s))}$$

$$+ \sum_{(\gamma_1,\gamma_2)\in G_{S_1,S_2}-G^d_{S_1,S_2}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2 \text{ and } \mathrm{off}(\gamma_1,\gamma_2)\le d(s))$$

$$\le \sum_{(\gamma_1,\gamma_2)\in G^d_{S_1,S_2},\gamma_1=\gamma_2} \boxed{\min\left(1,\frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)}$$

$$+ \sum_{(\gamma_1,\gamma_2)\in (G_{S_1,S_2}-G^d_{S_1,S_2})} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2). \quad \square$$

$$\Pr(d(s)\ge \mathrm{off}(\gamma_1,\gamma_2)) \le \min\left(1,\frac{\tau}{\mathrm{off}(\gamma_1,\gamma_2)}\right)$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- **Theorem 4** *For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1, S_2)$:*

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|) - 1}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G^d_{S_1,S_2}, \gamma_1 = \gamma_2} \min\left(1, \frac{\tau}{\text{off}(\gamma_1,\gamma_2)}\right)}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G_{S_1,S_2} - G^d_{S_1,S_2}} \Pr(\gamma_1 = \gamma_2)}{q}.$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1, S_2)$:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|) - 1}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G^d_{S_1,S_2}, \gamma_1 = \gamma_2} \min\left(1, \frac{\tau}{\text{off}(\gamma_1,\gamma_2)}\right)}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G_{S_1,S_2} - G^d_{S_1,S_2}} \Pr(\gamma_1 = \gamma_2)}{q}.$$

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1, S_2)$:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|) - 1}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G^d_{S_1,S_2}, \gamma_1 = \gamma_2} \min\left(1, \frac{\tau}{\text{off}(\gamma_1,\gamma_2)}\right)}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G_{S_1,S_2} - G^d_{S_1,S_2}} \Pr(\gamma_1 = \gamma_2)}{q}.$$

- ```
1 SELECT R.id AS rid,T.id AS tid FROM R,T,Rq,Tq
2 WHERE Rq.g=Tq.g AND R.id=Rq.id AND T.id=Tq.id
3 GROUP BY R.id, T.id, R.len, T.len
4 HAVING 1 + (max(R.len,T.len)-1)/q -
5 SUM(FLOOR(Tq.p*Rq.p)*
6   min(1, τ/max(1,ABS(Rq.ℓ - Tq.ℓ))))/q -
7   SUM(CEILING(1-Rq.p*Tq.p)*Tq.p*Rq.p)/q ≤ τ
```
```

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1, S_2)$:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|) - 1}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G^d_{S_1,S_2}, \gamma_1 = \gamma_2} \min\left(1, \frac{\tau}{\text{off}(\gamma_1,\gamma_2)}\right)}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G_{S_1,S_2} - G^d_{S_1,S_2}} \Pr(\gamma_1 = \gamma_2)}{q}.$$

- ```
1 SELECT R.id AS rid,T.id AS tid FROM R,T,Rq,Tq
2 WHERE Rq.g=Tq.g AND R.id=Rq.id AND T.id=Tq.id
3 GROUP BY R.id, T.id, R.len, T.len
4 HAVING 1 + (max(R.len,T.len)-1)/q -
5 SUM(FLOOR(Tq.p*Rq.p)*
6   min(1, τ/max(1,ABS(Rq.ℓ - Tq.ℓ))))/q -
7   SUM(CEILING(1-Rq.p*Tq.p)*Tq.p*Rq.p)/q ≤ τ
```
```

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1, S_2)$:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|) - 1}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G^d_{S_1,S_2}, \gamma_1 = \gamma_2} \min\left(1, \frac{\tau}{\text{off}(\gamma_1,\gamma_2)}\right)}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G_{S_1,S_2} - G^d_{S_1,S_2}} \Pr(\gamma_1 = \gamma_2)}{q}.$$

```
1  SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
2  WHERE R_q.g=T_q.g AND R.id=R_q.id AND T.id=T_q.id
3  GROUP BY R.id, T.id, R.len, T.len
4  HAVING 1 + (max(R.len,T.len)-1)/q -
5  SUM( FLOOR(T_q.p*R_q.p)*
6      min(1, τ/max(1,ABS(R_q.ℓ - T_q.ℓ))) )/q -
7      SUM( CEILING(1-R_q.p*T_q.p)*T_q.p*R_q.p )/q ≤ τ
```

Character-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2, $\forall \tau \geq \widehat{d}(S_1, S_2)$:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|) - 1}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G^d_{S_1,S_2}, \gamma_1 = \gamma_2} \min\left(1, \frac{\tau}{\text{off}(\gamma_1,\gamma_2)}\right)}{q} - \frac{\sum_{(\gamma_1,\gamma_2) \in G_{S_1,S_2} - G^d_{S_1,S_2}} \Pr(\gamma_1 = \gamma_2)}{q}.$$

- 1 SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
  2 WHERE R_q.g=T_q.g AND R.id=R_q.id AND T.id=T_q.id
  3 GROUP BY R.id, T.id, R.len, T.len
  4 HAVING 1 + (max(R.len,T.len)-1)/q -
  5 SUM(FLOOR(T_q.p*R_q.p)*
  6 min(1, τ/max(1,ABS(R_q.ℓ - T_q.ℓ))))/q -
  7 SUM(CEILING(1-R_q.p*T_q.p)*T_q.p*R_q.p)/q $\leq$ τ

Character-Level Probabilistic Strings: Dynamic Program Lower Bounds

- We can derive a DP based lower-bound on the ED as follows,

$$\widehat{ld}[i,j] = \min \begin{cases} \widehat{ld}[i,j-1] + 1; \\ \widehat{ld}[i-1,j] + 1; \\ \widehat{ld}[i-1,j-1] + lc(S_1[i], S_2[j]), \end{cases}$$

where

$$lc(S_1[i], S_2[j]) = \begin{cases} 1, \Pr(S_1[i] = S_2[j]) = 0; \\ 0, \Pr(S_1[i] = S_2[j]) > 0. \end{cases}$$

Character-Level Probabilistic Strings: Dynamic Program Lower Bounds

- We can derive a DP based lower-bound on the $\textsc{Eed}$ as follows,

$$\widehat{ld}[i,j] = \min \begin{cases} \widehat{ld}[i,j-1] + 1; \\ \widehat{ld}[i-1,j] + 1; \\ \widehat{ld}[i-1,j-1] + lc(S_1[i], S_2[j]), \end{cases}$$

where

$$lc(S_1[i], S_2[j]) = \begin{cases} 1, \Pr(S_1[i] = S_2[j]) = 0; \\ 0, \Pr(S_1[i] = S_2[j]) > 0. \end{cases}$$

- **Theorem 5** *For any two character-level probabilistic strings* $S_1 = S_1[1] \ldots S_1[n_1]$ *and* $S_2 = S_2[1] \ldots S_2[n_2]$,

$$\widehat{ld}[n_1, n_2] = \min_{s \in \Omega}(d(s)) \leq \widehat{d}(S_1, S_2).$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- Let $q_1 = (\ell_1, g_1)$, $q_2 = (\ell_2, g_2)$
  Where $q_1 \in G_{\sigma_1}$ and $q_2 \in G_{\sigma_2}$
  Define $q_1 \equiv q_2 \iff g_1 = g_2$ *and* $\ell_1 = \ell_2$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- Let $q_1 = (\ell_1, g_1)$, $q_2 = (\ell_2, g_2)$
  Where $q_1 \in G_{\sigma_1}$ and $q_2 \in G_{\sigma_2}$
  Define $q_1 \equiv q_2 \iff g_1 = g_2$ *and* $\ell_1 = \ell_2$

- **Lemma 8** *For any two deterministic strings σ_1 and σ_2:*

$$d(\sigma_1, \sigma_2) \leq \max(|\sigma_1|, |\sigma_2|) + q - 1 - \sum_{q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}} (q_1 \equiv q_2)$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- Let $q_1 = (\ell_1, g_1)$, $q_2 = (\ell_2, g_2)$
  Where $q_1 \in G_{\sigma_1}$ and $q_2 \in G_{\sigma_2}$
  Define $q_1 \equiv q_2 \iff g_1 = g_2$ *and* $\ell_1 = \ell_2$

- **Lemma 8** *For any two deterministic strings σ_1 and σ_2:*

$$d(\sigma_1, \sigma_2) \leq \max(|\sigma_1|, |\sigma_2|) + q - 1 - \sum_{q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}} \boxed{(q_1 \equiv q_2)}$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- Let $q_1 = (\ell_1, g_1)$, $q_2 = (\ell_2, g_2)$
  Where $q_1 \in G_{\sigma_1}$ and $q_2 \in G_{\sigma_2}$
  Define $q_1 \equiv q_2 \iff g_1 = g_2$ *and* $\ell_1 = \ell_2$

- **Lemma 8** *For any two deterministic strings σ_1 and σ_2:*

$$d(\sigma_1, \sigma_2) \leq \max(|\sigma_1|, |\sigma_2|) + q - 1 - \sum_{q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}} (q_1 \equiv q_2)$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- Let $q_1 = (\ell_1, g_1)$, $q_2 = (\ell_2, g_2)$
  Where $q_1 \in G_{\sigma_1}$ and $q_2 \in G_{\sigma_2}$
  Define $q_1 \equiv q_2 \iff g_1 = g_2$ *and* $\ell_1 = \ell_2$

- **Lemma 8** *For any two deterministic strings σ_1 and σ_2:*

$$d(\sigma_1, \sigma_2) \leq \max(|\sigma_1|, |\sigma_2|) + q - 1 - \sum_{q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}} (q_1 \equiv q_2)$$

- **Theorem 6** *For any character-level probabilistic strings S_1, S_2:*

$$\widehat{d}(S_1, S_2) \leq \max(|S_1|, |S_2|) + q - 1 - \sum_{(\gamma_1, \gamma_2) \in G_{S_1, S_2}} Pr(\gamma_1 \equiv \gamma_2)$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- Let $q_1 = (\ell_1, g_1)$, $q_2 = (\ell_2, g_2)$
  Where $q_1 \in G_{\sigma_1}$ and $q_2 \in G_{\sigma_2}$
  Define $q_1 \equiv q_2 \iff g_1 = g_2$ *and* $\ell_1 = \ell_2$

- **Lemma 8** *For any two deterministic strings σ_1 and σ_2:*

$$d(\sigma_1, \sigma_2) \leq \max(|\sigma_1|, |\sigma_2|) + q - 1 - \sum_{q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}} (q_1 \equiv q_2)$$

- **Theorem 6** *For any character-level probabilistic strings S_1, S_2:*

$$\widehat{d}(S_1, S_2) \leq \max(|S_1|, |S_2|) + q - 1 - \sum_{(\gamma_1, \gamma_2) \in G_{S_1, S_2}} Pr(\gamma_1 \equiv \gamma_2)$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- Let $q_1 = (\ell_1, g_1)$, $q_2 = (\ell_2, g_2)$
  Where $q_1 \in G_{\sigma_1}$ and $q_2 \in G_{\sigma_2}$
  Define $q_1 \equiv q_2 \iff g_1 = g_2$ *and* $\ell_1 = \ell_2$

- **Lemma 8** *For any two deterministic strings σ_1 and σ_2:*

$$d(\sigma_1, \sigma_2) \leq \max(|\sigma_1|, |\sigma_2|) + q - 1 - \sum_{q_1 \in G_{\sigma_1}, q_2 \in G_{\sigma_2}} (q_1 \equiv q_2)$$

- **Theorem 6** *For any character-level probabilistic strings S_1, S_2:*

$$\widehat{d}(S_1, S_2) \leq \max(|S_1|, |S_2|) + q - 1 - \sum_{(\gamma_1, \gamma_2) \in G_{S_1, S_2}} Pr(\gamma_1 \equiv \gamma_2)$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \leq \max(|S_1|, |S_2|) + q - 1 - \sum_{(\gamma_1, \gamma_2) \in G_{S_1, S_2}} Pr(\gamma_1 \equiv \gamma_2)$$

Character-Level Probabilistic Strings: q-Gram Upper Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \leq \max(|S_1|, |S_2|) + q - 1 - \sum_{(\gamma_1, \gamma_2) \in G_{S_1, S_2}} Pr(\gamma_1 \equiv \gamma_2)$$

```
1  SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
2  WHERE R_q.g=T_q.g AND R_q.ℓ=T_q.ℓ
3     AND R.id=R_q.id AND T.id=T_q.id
4  GROUP BY R.id, T.id, R.len, T.len
5  HAVING max(R.len,T.len)+q-1-SUM(T_q.p*R_q.p)≤τ
```

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

- We can derive a DP-based upper bound on the $\textsc{Eed}$ as follows

$$\widehat{ud}[i,j] = \min \begin{cases} \widehat{ud}[i,j-1] + 1, \\ \widehat{ud}[i-1,j] + 1, \\ \widehat{ud}[i-1,j-1] + \Pr(S_1[i] \neq S_2[j]), \end{cases}$$

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

- We can derive a DP-based upper bound on the $\textsc{Eed}$ as follows

$$\widehat{ud}[i,j] = \min \begin{cases} \widehat{ud}[i,j-1] + 1, \\ \widehat{ud}[i-1,j] + 1, \\ \widehat{ud}[i-1,j-1] + \Pr(S_1[i] \neq S_2[j]), \end{cases}$$

- **Theorem 7** $\widehat{ud}[i,j] \geq \widehat{d}(S_1[1..i], S_2[1..j])$ *for all* i, j.

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

- **Theorem** $\widehat{ud}[i,j] \geq \widehat{d}(S_1[1..i], S_2[1..j])$ for all i, j

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

- **Theorem** $\widehat{ud}[i,j] \geq \widehat{d}(S_1[1..i], S_2[1..j])$ for all i, j

- **Proof:** We will need,

  Lemma 10 *For two non-negative random variables X, Y,*

$$\mathbf{E}[\min(X,Y)] \leq \min(\mathbf{E}[X], \mathbf{E}[Y]).$$

and,

Corollary 2 *For three non-negative random variables X, Y, Z,*

$$\mathbf{E}[\min(X,Y,Z)] \leq \min(\mathbf{E}[X], \mathbf{E}[Y], \mathbf{E}[Z]).$$

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

- **Theorem** $\widehat{ud}[i,j] \geq \widehat{d}(S_1[1..i], S_2[1..j])$ for all i, j

- The theorem holds when $i \cdot j = 0$
  Assume it holds $\forall i' \leq i, j' \leq j$.
  Let $s = \{(\sigma_1, p_1), (\sigma_2, p_2)\}$ be a random world from Ω

$$d_s[i,j] = \min \begin{cases} d_s[i-1,j] + 1, \\ d_s[i,j-1] + 1, \\ d_s[i-1,j-1] + c(\sigma_1[i], \sigma_2[j]). \end{cases}$$

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

$$\widehat{d}(S_1[1..i], S_2[1..j])$$

$$= \mathbf{E}\left[\min\begin{cases} d_s[i-1,j]+1 \\ d_s[i,j-1]+1 \\ d_s[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]) \end{cases}\right]$$

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

$$
\begin{aligned}
& \widehat{d}(S_1[1..i], S_2[1..j]) \\
= \quad & \mathbf{E}\left[\min\begin{cases} d_s[i-1,j]+1 \\ d_s[i,j-1]+1 \\ d_s[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]) \end{cases}\right] \\
\leq \quad & \min\begin{cases} \widehat{d}(S_1[1..i-1], S_2[1..j])+1 \\ \widehat{d}(S_1[1..i], S_2[1..j-1])+1 \qquad \text{(by Corollary 2)} \\ \widehat{d}(S_1[1..i-1], S_2[1..j-1])+\Pr(S_1[i] \neq S_2[j]) \end{cases}
\end{aligned}
$$

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

$$
\begin{aligned}
& \widehat{d}(S_1[1..i], S_2[1..j]) \\
= \quad & \mathbf{E}\left[\min\begin{cases} d_s[i-1,j]+1 \\ d_s[i,j-1]+1 \\ d_s[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]) \end{cases}\right] \\
\leq \quad & \min\begin{cases} \widehat{d}(S_1[1..i-1], S_2[1..j])+1 \\ \widehat{d}(S_1[1..i], S_2[1..j-1])+1 & \text{(by Corollary 2)} \\ \widehat{d}(S_1[1..i-1], S_2[1..j-1])+\Pr(S_1[i]\neq S_2[j]) \end{cases} \\
\leq \quad & \min\begin{cases} \widehat{ud}[i-1,j]+1 \\ \widehat{ud}[i,j-1]+1 & \text{(by induction)} \\ \widehat{ud}[i-1,j-1]+\Pr(S_1[i]\neq S_2[j]) \end{cases}
\end{aligned}
$$

Character-Level Probabilistic Strings: Dynamic Program Upper Bounds

$$
\begin{aligned}
& \widehat{d}(S_1[1..i], S_2[1..j]) \\
= \quad & \mathbf{E}\left[\min\begin{cases} d_s[i-1,j]+1 \\ d_s[i,j-1]+1 \\ d_s[i-1,j-1]+c(\sigma_1[i],\sigma_2[j]) \end{cases}\right] \\
\leq \quad & \min\begin{cases} \widehat{d}(S_1[1..i-1], S_2[1..j])+1 \\ \widehat{d}(S_1[1..i], S_2[1..j-1])+1 & \text{(by Corollary 2)} \\ \widehat{d}(S_1[1..i-1], S_2[1..j-1])+\Pr(S_1[i]\neq S_2[j]) \end{cases} \\
\leq \quad & \min\begin{cases} \widehat{ud}[i-1,j]+1 \\ \widehat{ud}[i,j-1]+1 & \text{(by induction)} \\ \widehat{ud}[i-1,j-1]+\Pr(S_1[i]\neq S_2[j]) \end{cases} \\
= \quad & \widehat{ud}[i,j]. \quad \square
\end{aligned}
$$

Experiments: Generation of Data Sets

- For each string σ_1 in the source database D we find the similar set $A(\sigma_1) = \{\sigma_2 | \sigma_2 \in D \wedge d(\sigma_1, \sigma_2) \leq 3\}$

Experiments: Generation of Data Sets

- For each string σ_1 in the source database D we find the similar set $A(\sigma_1) = \{\sigma_2 | \sigma_2 \in D \wedge d(\sigma_1, \sigma_2) \leq 3\}$
- For the genome data set, we break a long genome sequence into smaller strings with length uniformly in [10, 20]

Experiments: Generation of Data Sets

- For each string σ_1 in the source database D we find the similar set $A(\sigma_1) = \{\sigma_2 | \sigma_2 \in D \wedge d(\sigma_1, \sigma_2) \leq 3\}$
- For the genome data set, we break a long genome sequence into smaller strings with length uniformly in [10, 20]
- For string-level probabilistic strings:
  The choices are in $A(\sigma_1)$ with probability normalized to their frequencies, some strings are replaced with strings outside $A(\sigma_1)$

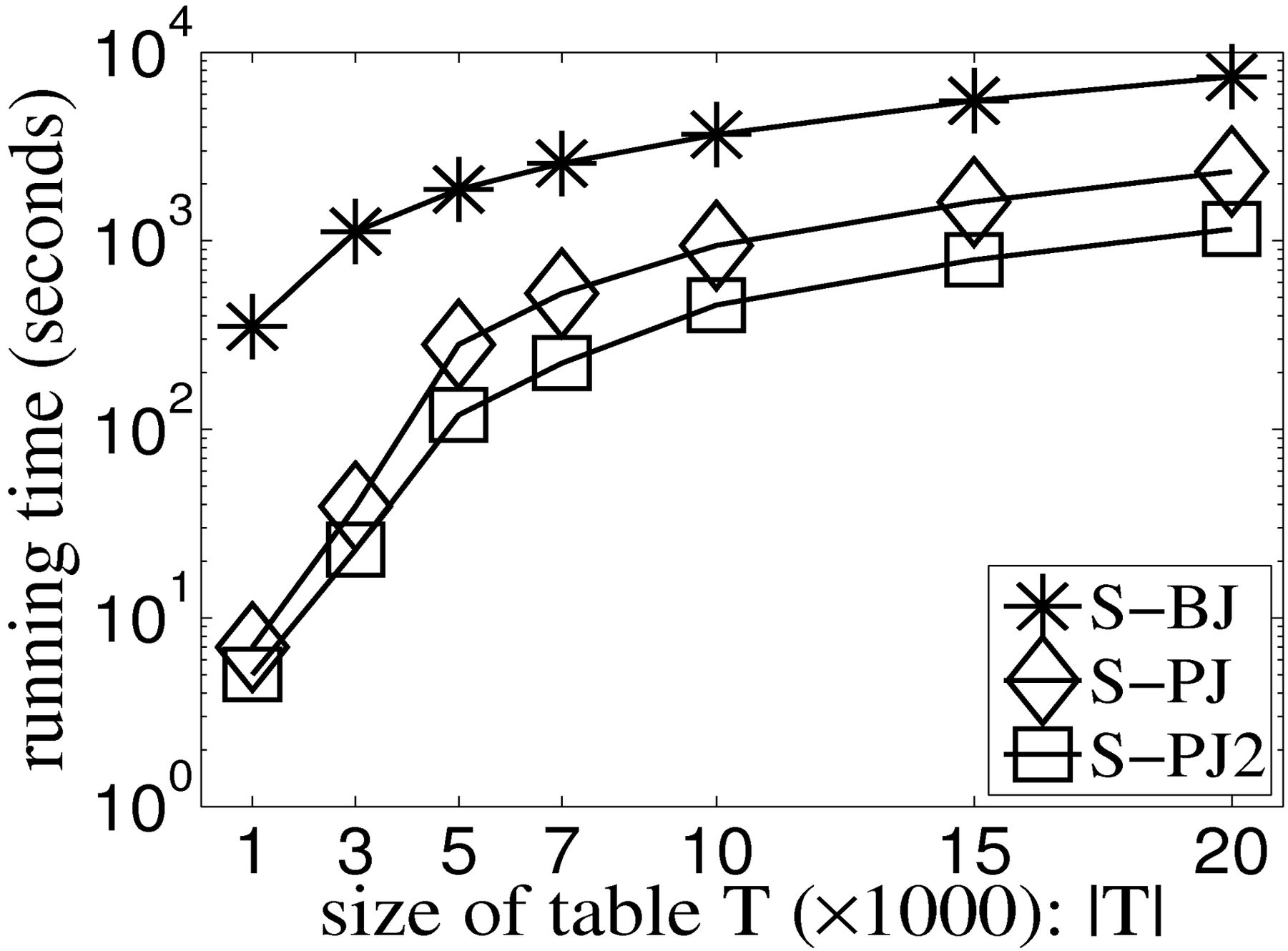
Experiments: Generation of Data Sets

- For each string σ_1 in the source database D we find the similar set $A(\sigma_1) = \{\sigma_2 | \sigma_2 \in D \wedge d(\sigma_1, \sigma_2) \leq 3\}$
- For the genome data set, we break a long genome sequence into smaller strings with length uniformly in [10, 20]
- For string-level probabilistic strings:
  The choices are in $A(\sigma_1)$ with probability normalized to their frequencies, some strings are replaced with strings outside $A(\sigma_1)$
- For a character-level probabilistic string S:
  We select a position $i \in [1, n]$ where $|\sigma_1| = n$ and generate the pdf of $S[i]$ based on the normalized frequency of characters in the ith position of strings in $A(\sigma_1)$.

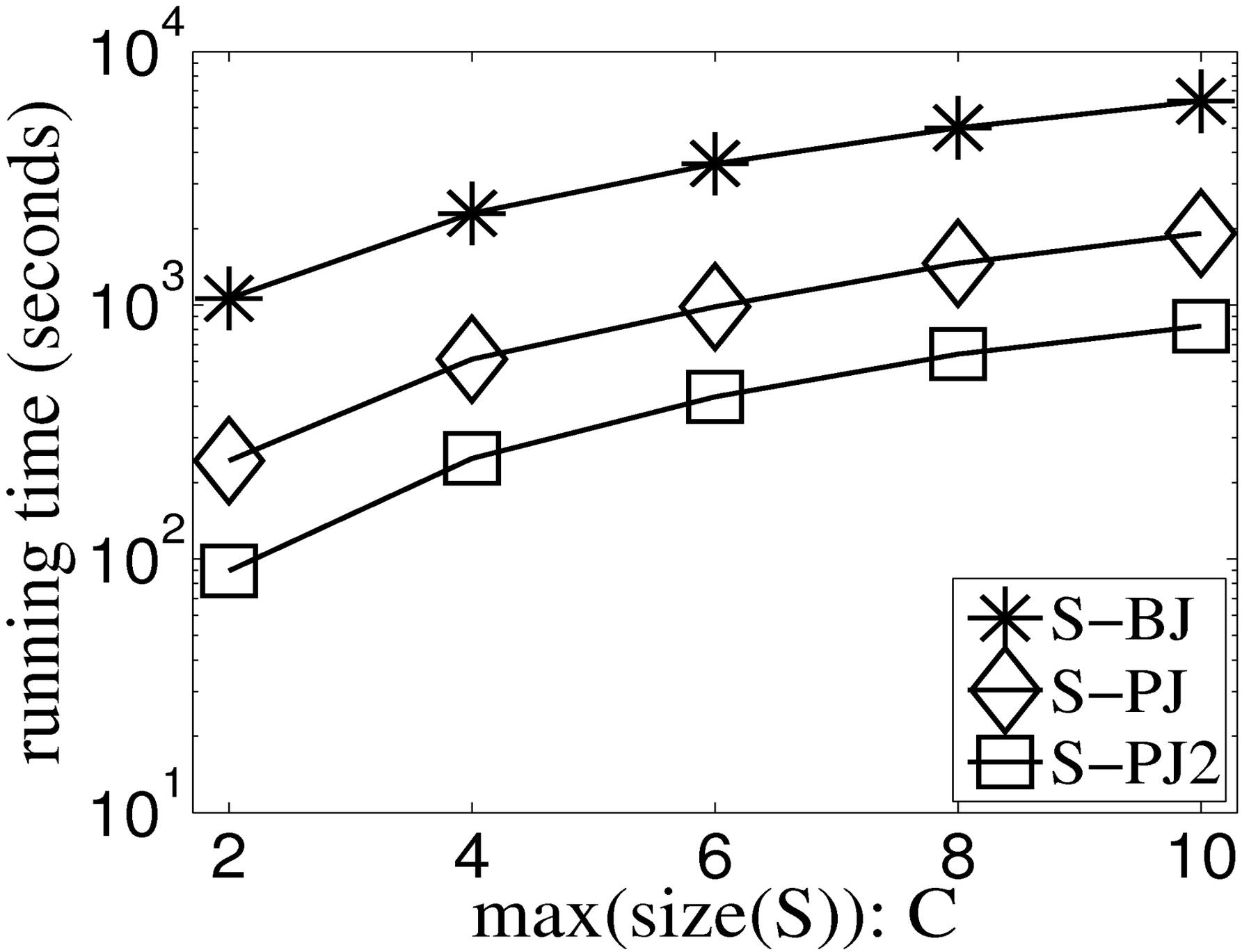
Experiments: String-Level Default Parameters

Symbol	Definition	Default Value
C	max(size(S))	6
ω	Self-concatenations	0
μ_S	Average $\lvert S(i)\rvert$	Author1 13.6 Category 19.9
$\lvert R\rvert$	Size table R	1,000
$\lvert T\rvert$	Size table T	10,000
q	q-gram	2
τ	$\textsc{Ed}$ threshold	2

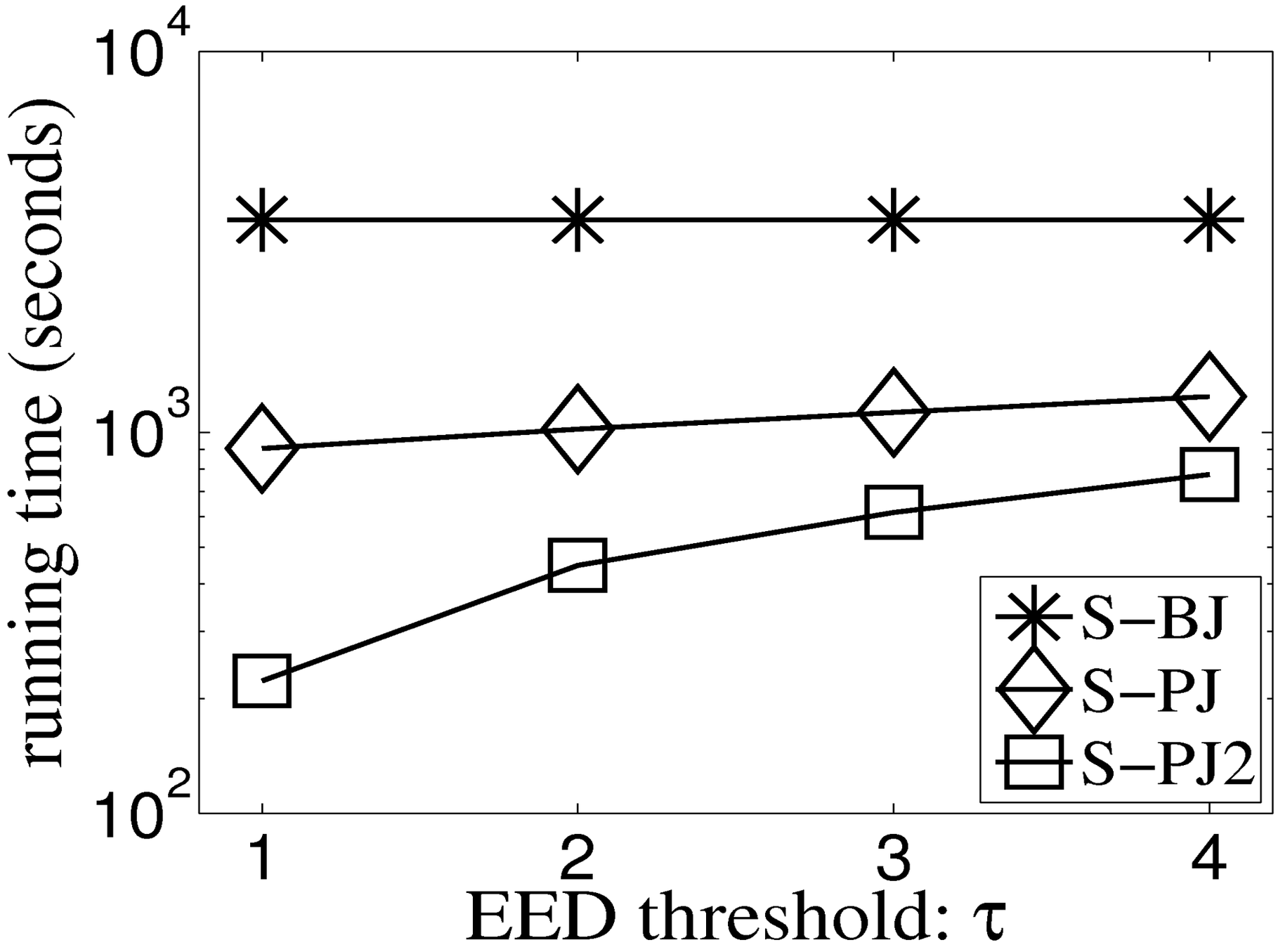
String-level model, *Author1* dataset



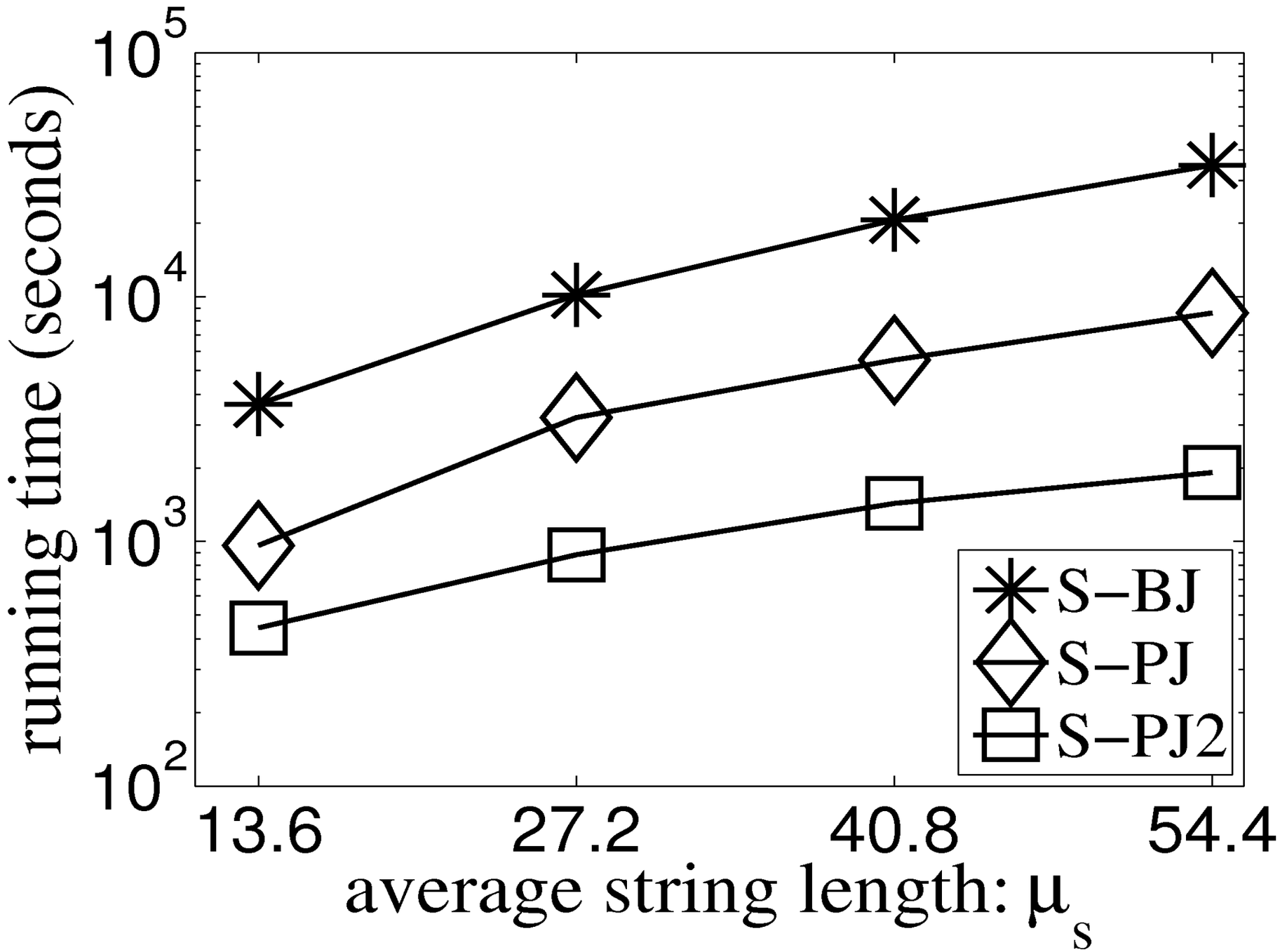
String-level model, *Author1* dataset



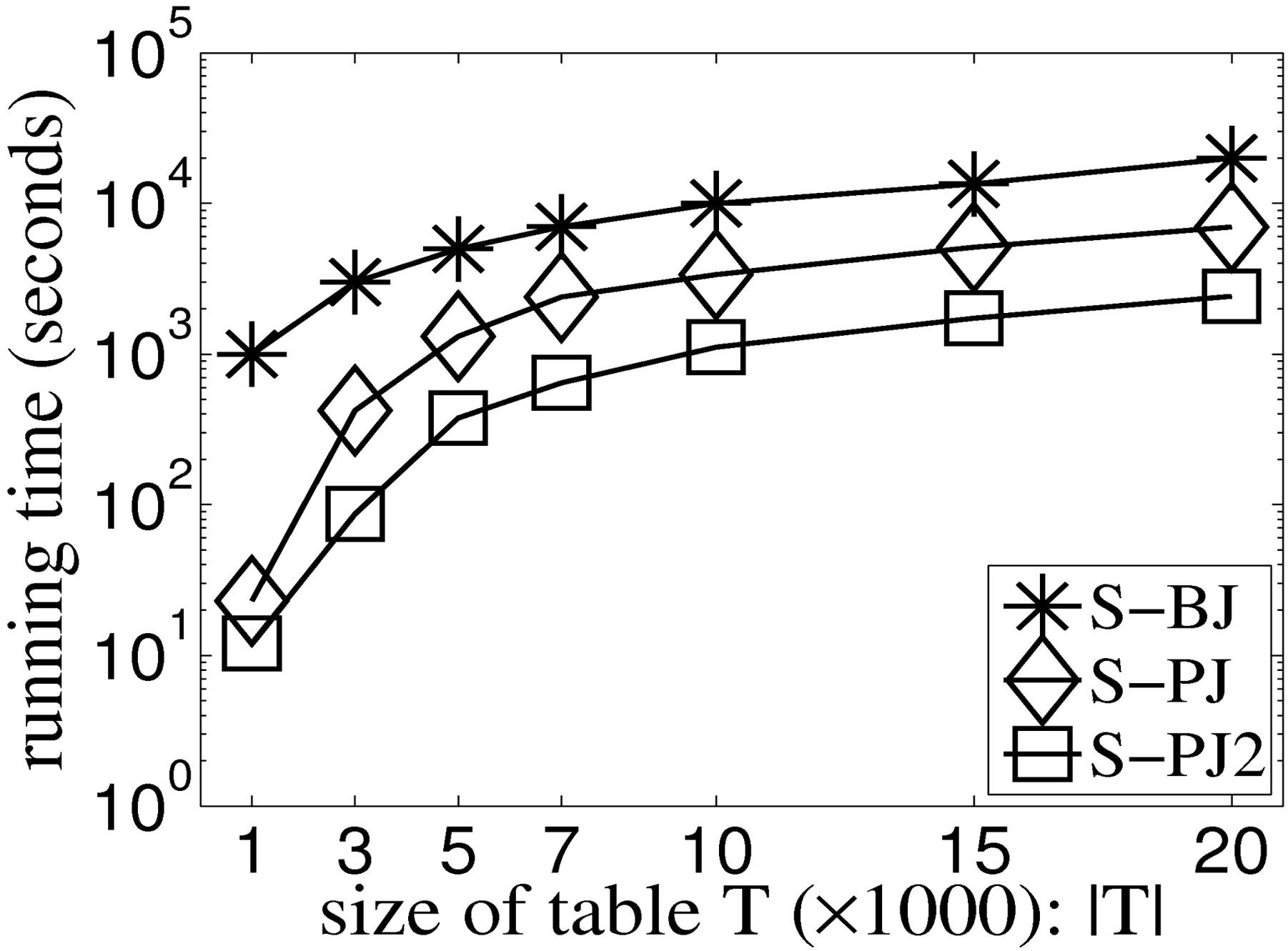
String-level model, *Author1* dataset



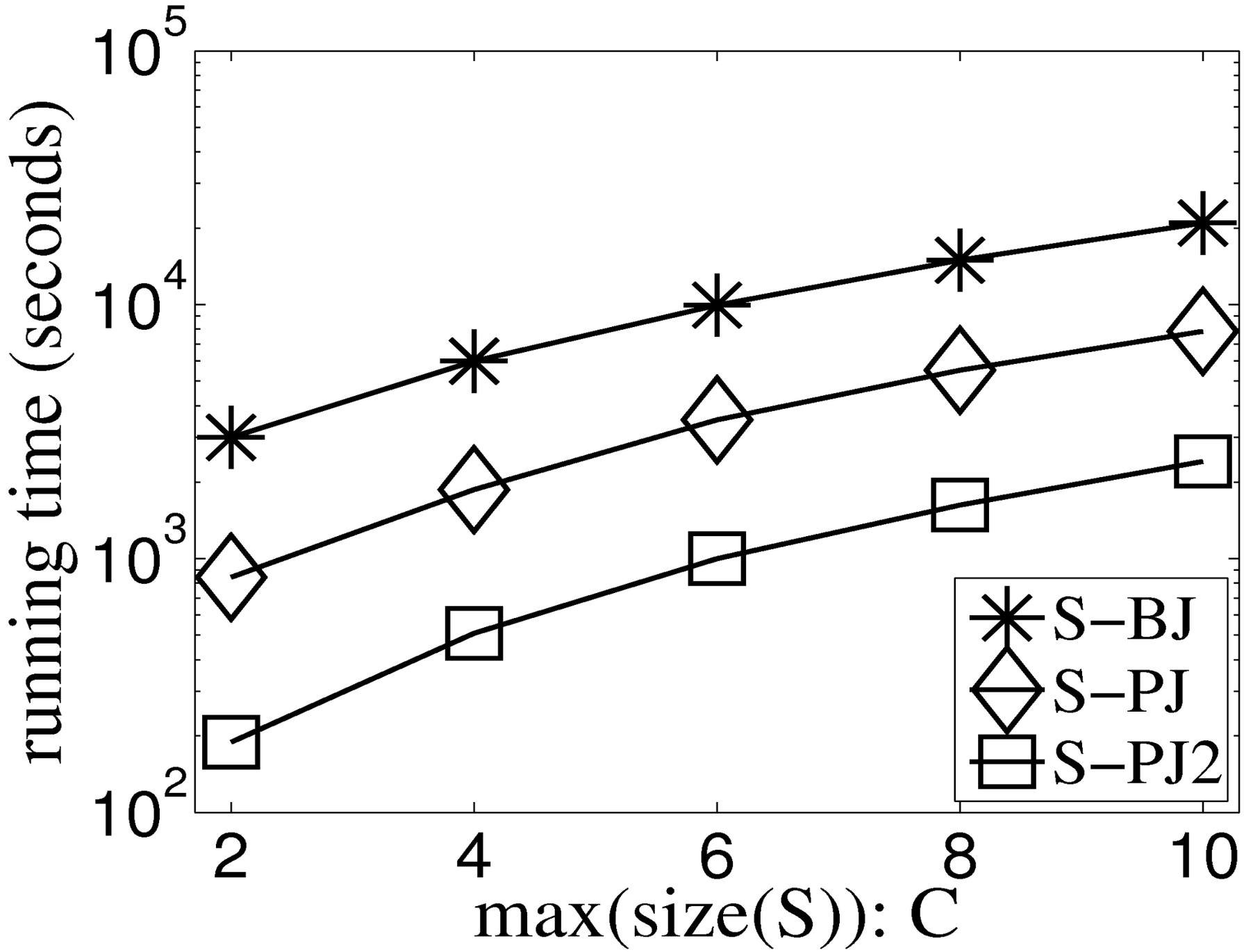
String-level model, *Author1* dataset



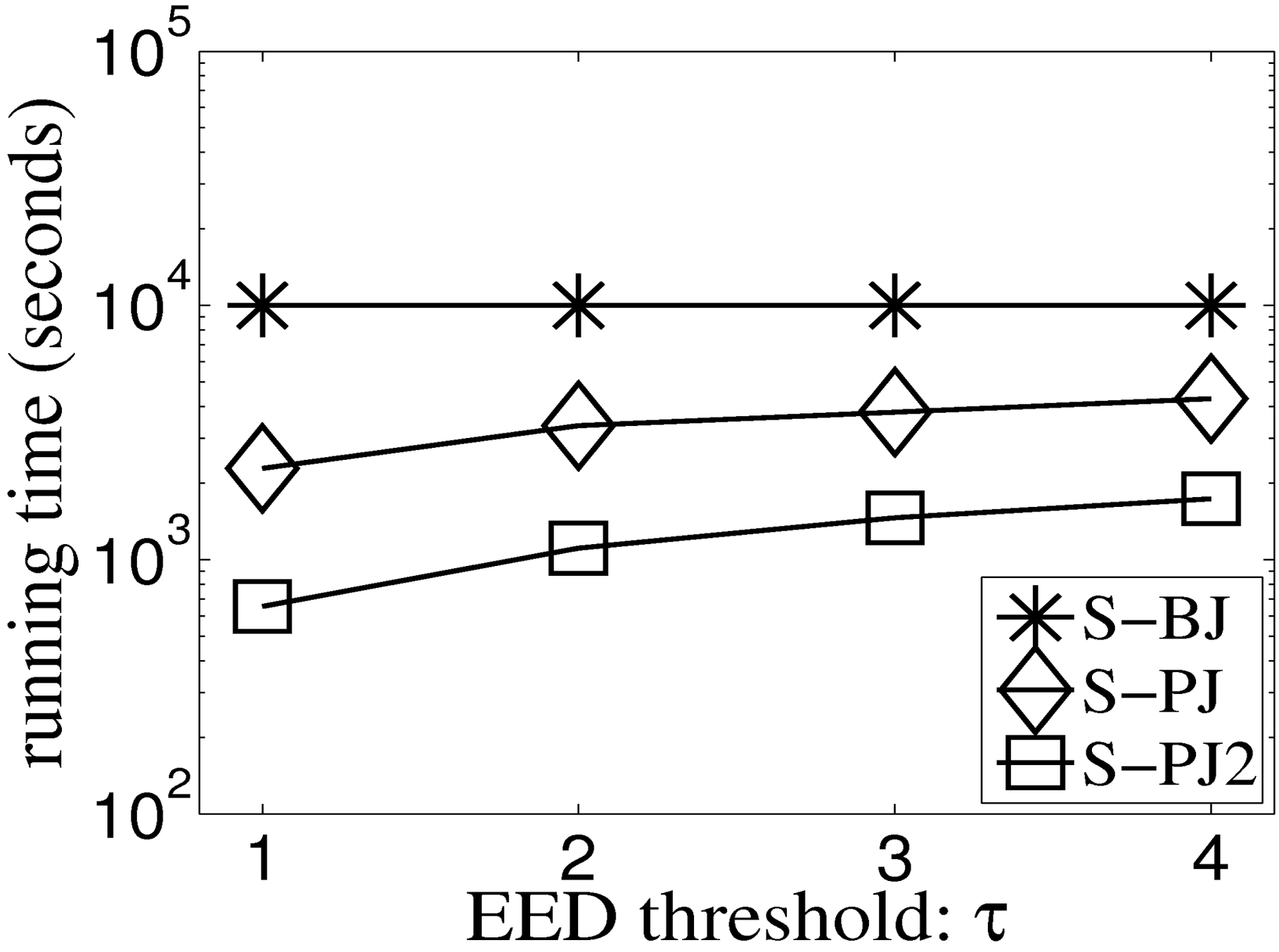
String-level model, *Category* dataset



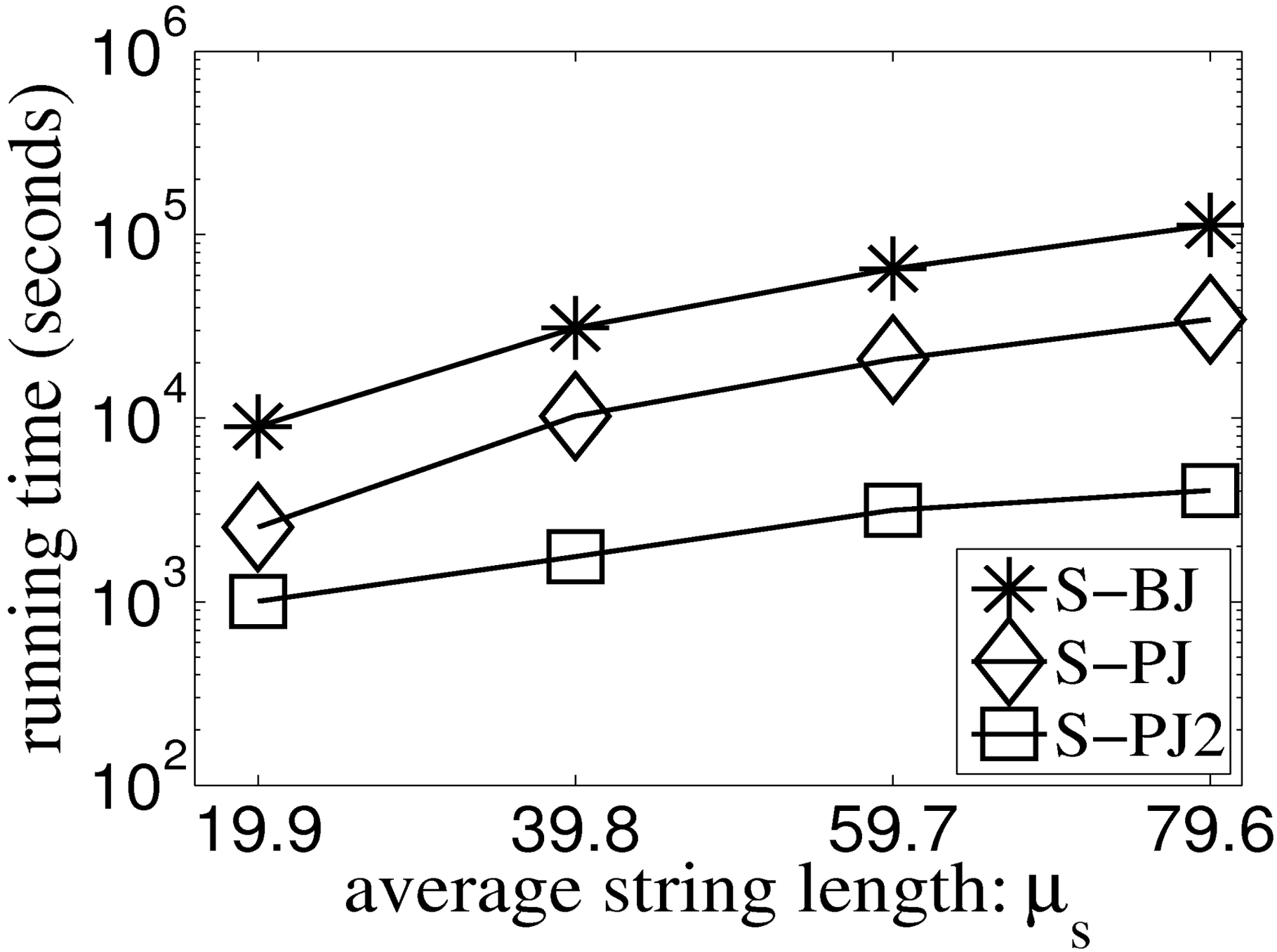
String-level model, *Category* dataset



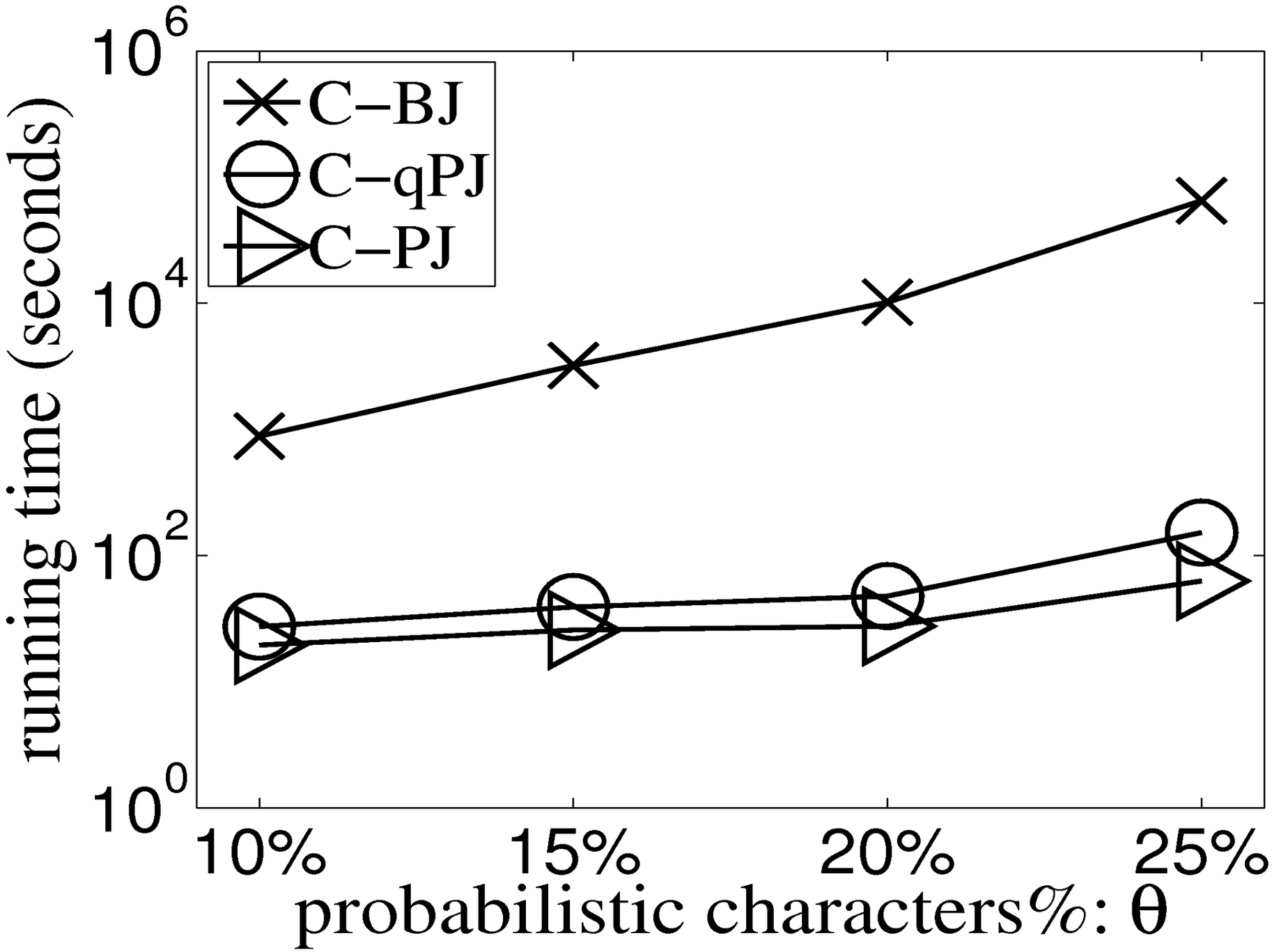
String-level model, *Category* dataset



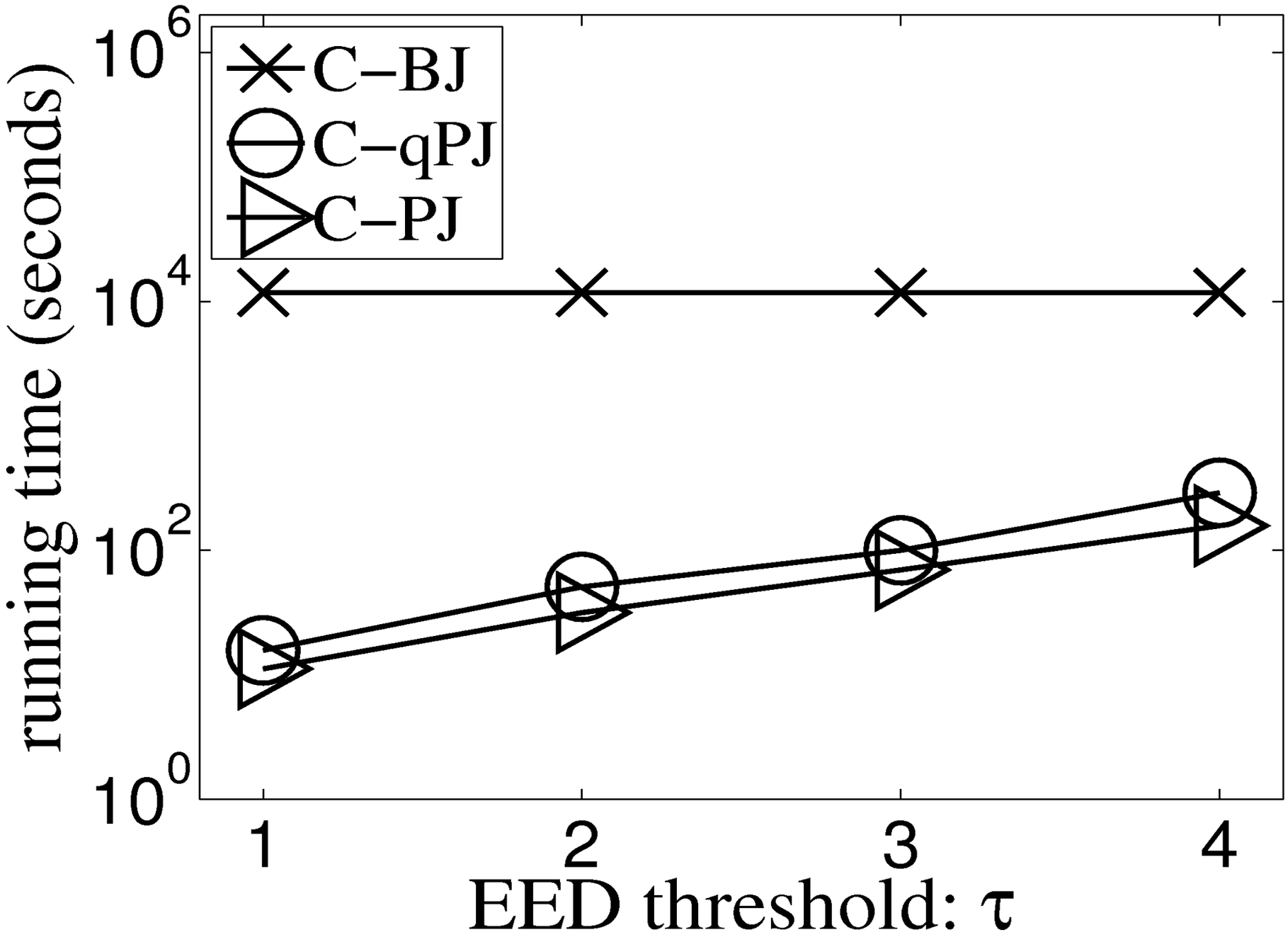
String-level model, *Category* dataset



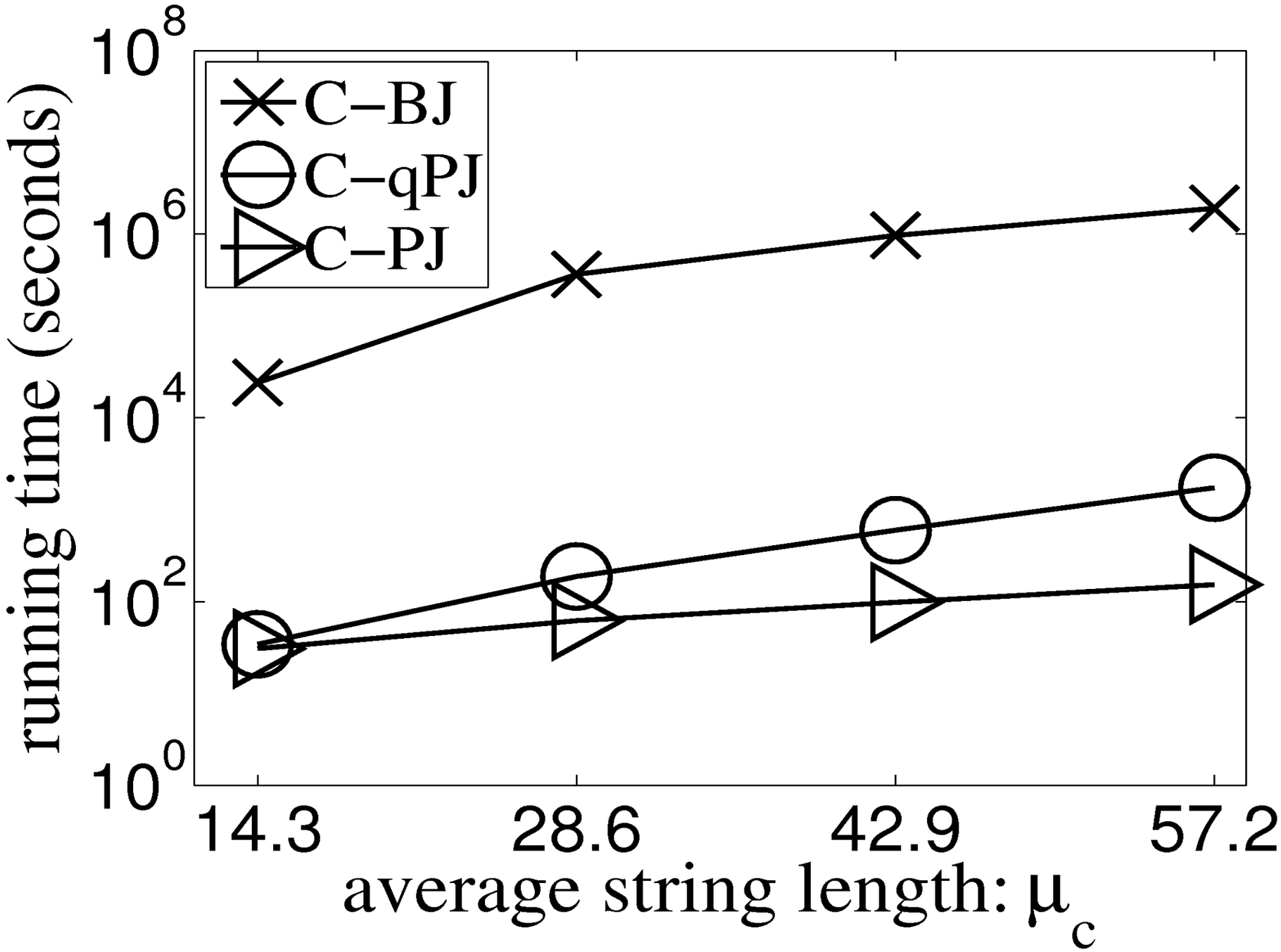
Character-level model, *Author2* dataset



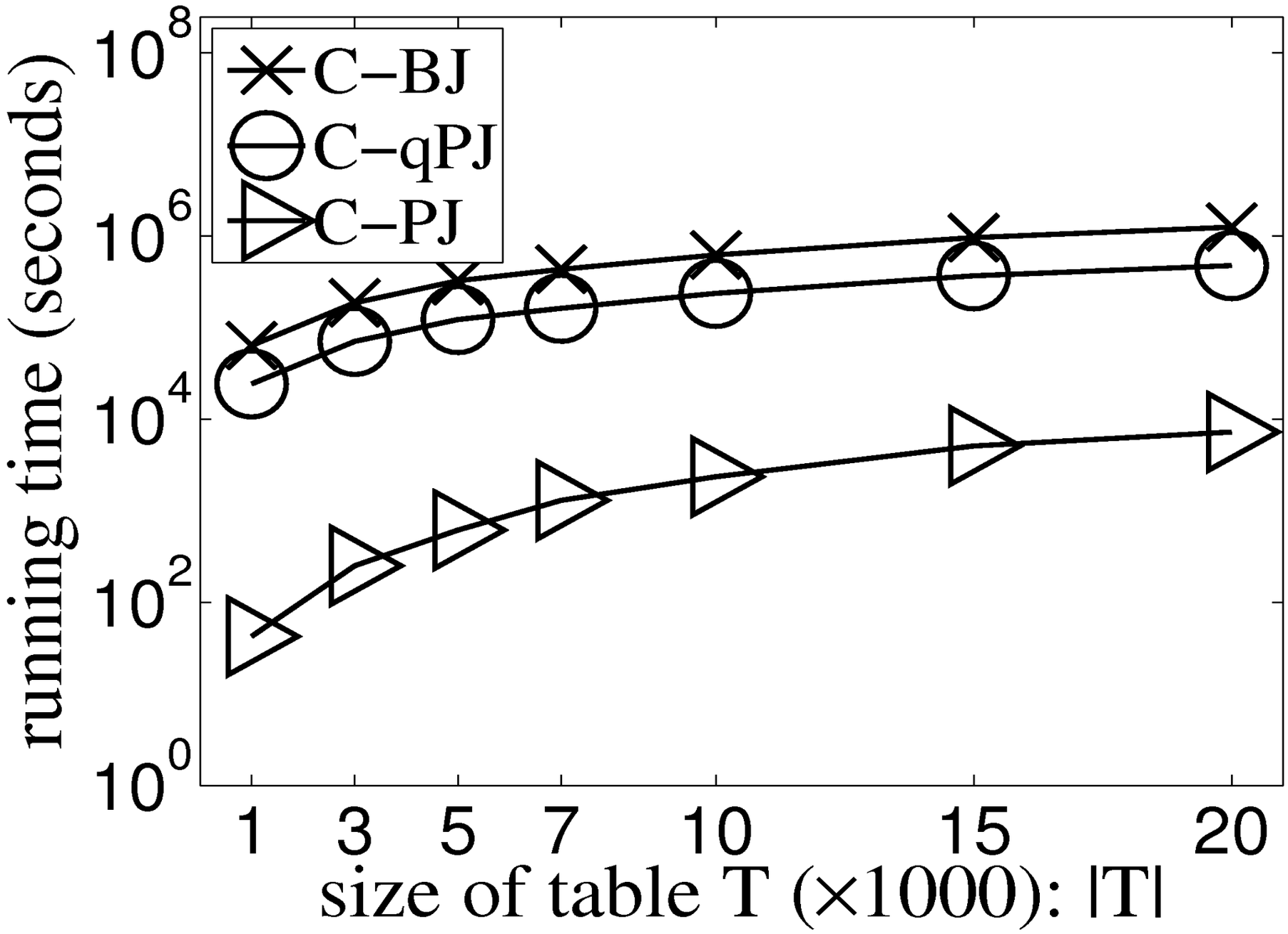
Character-level model, *Author2* dataset



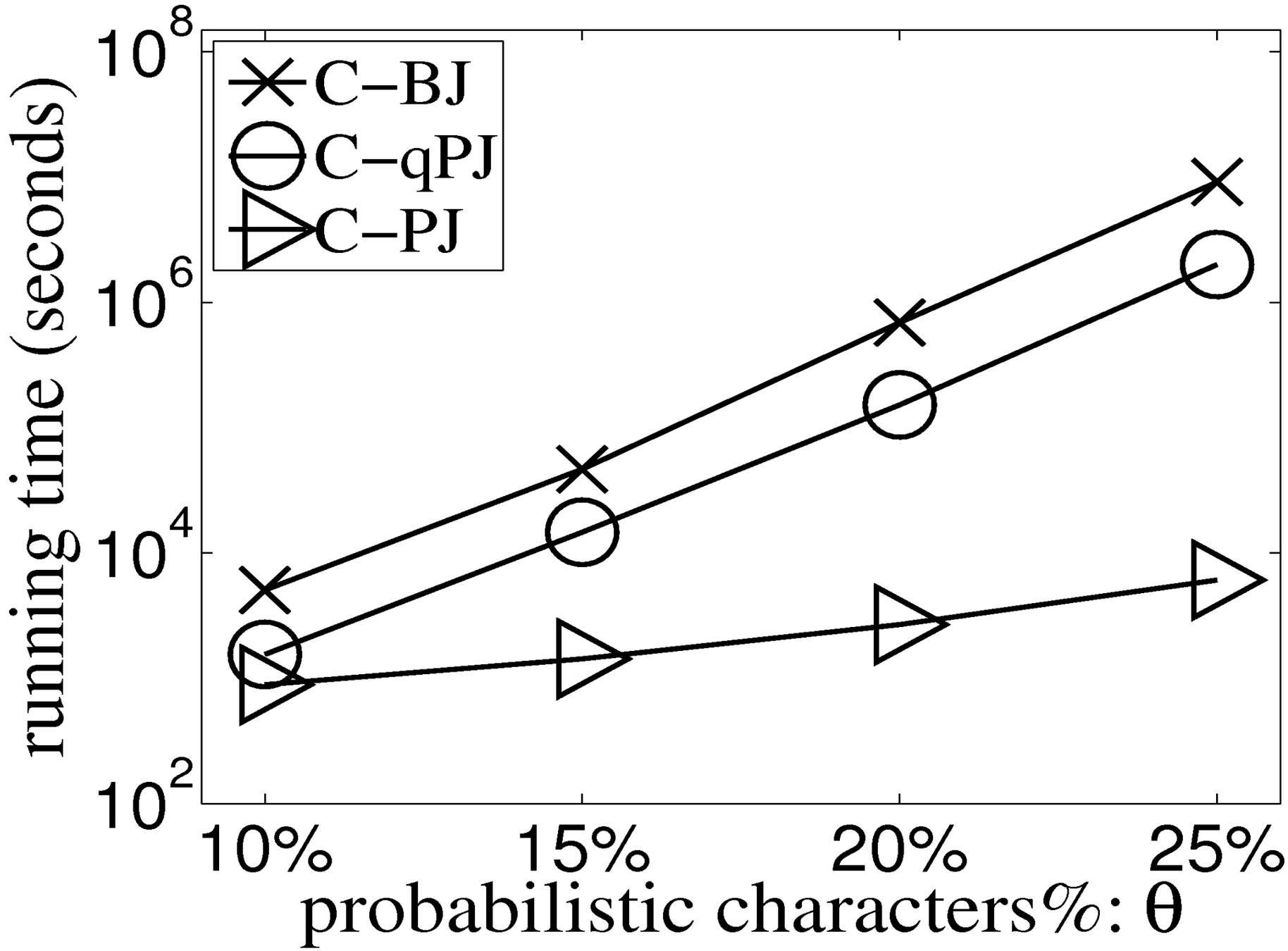
Character-level model, *Author2* dataset



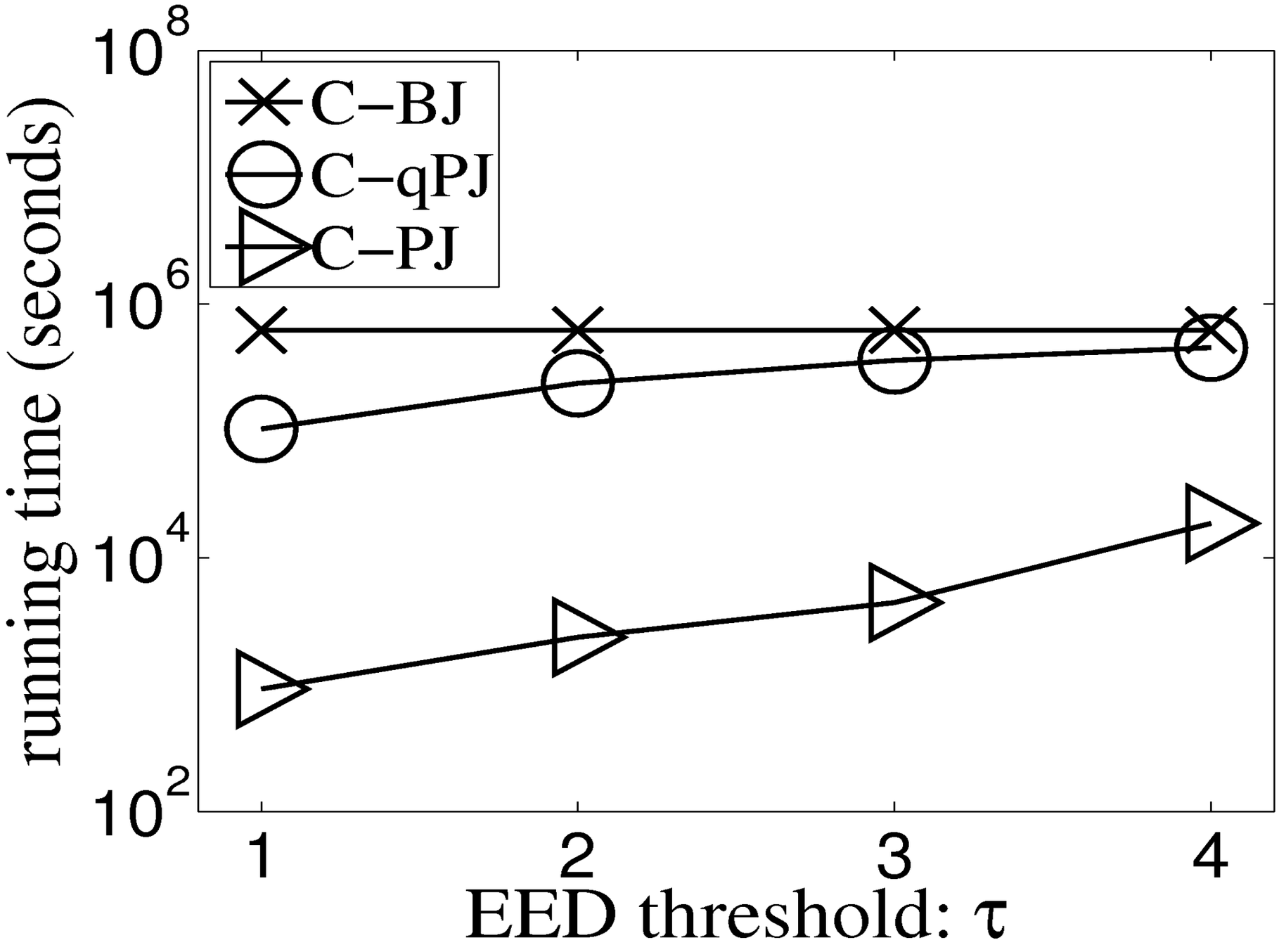
Character-level model, *Genome* dataset



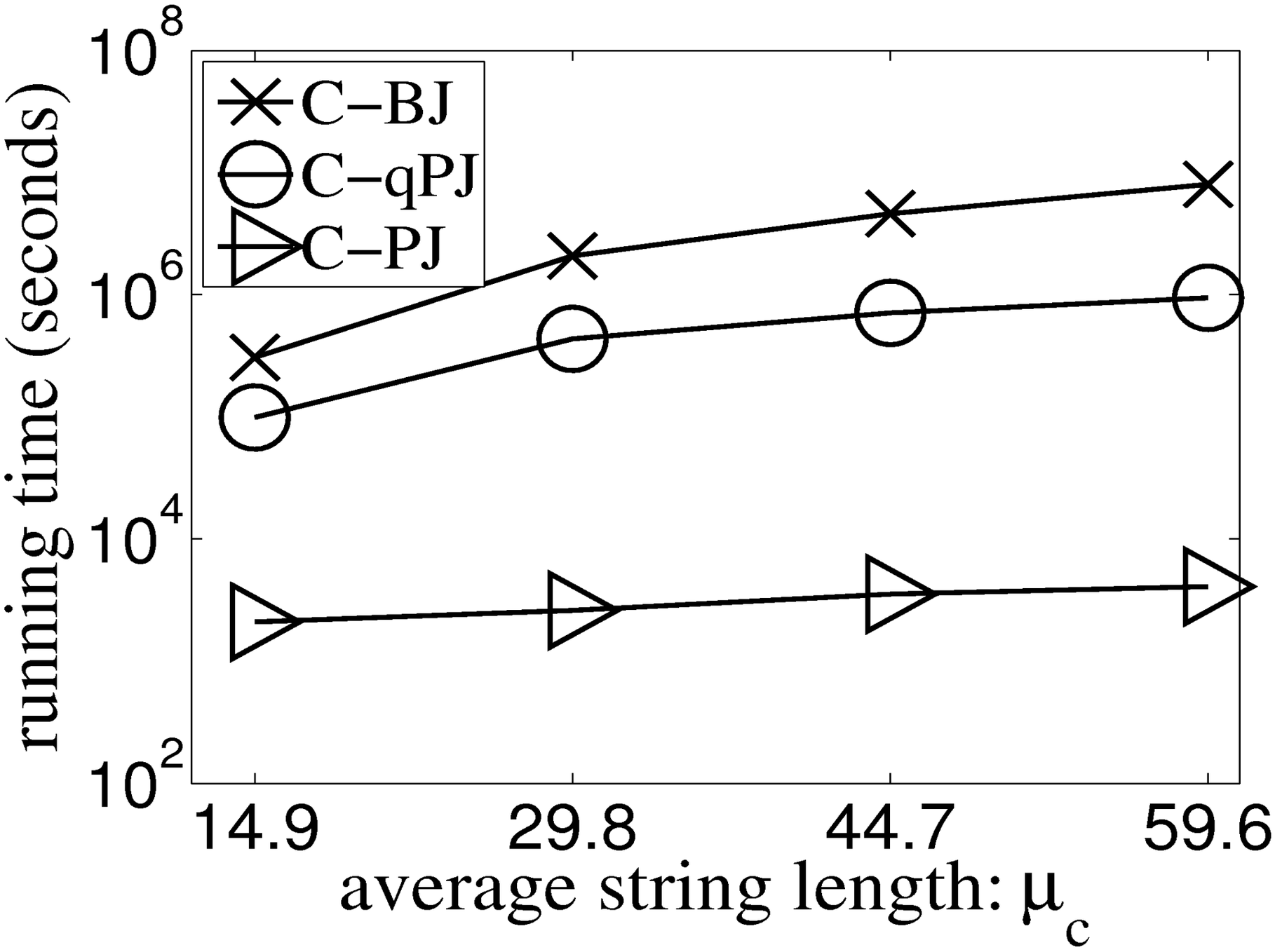
Character-level model, *Genome* dataset



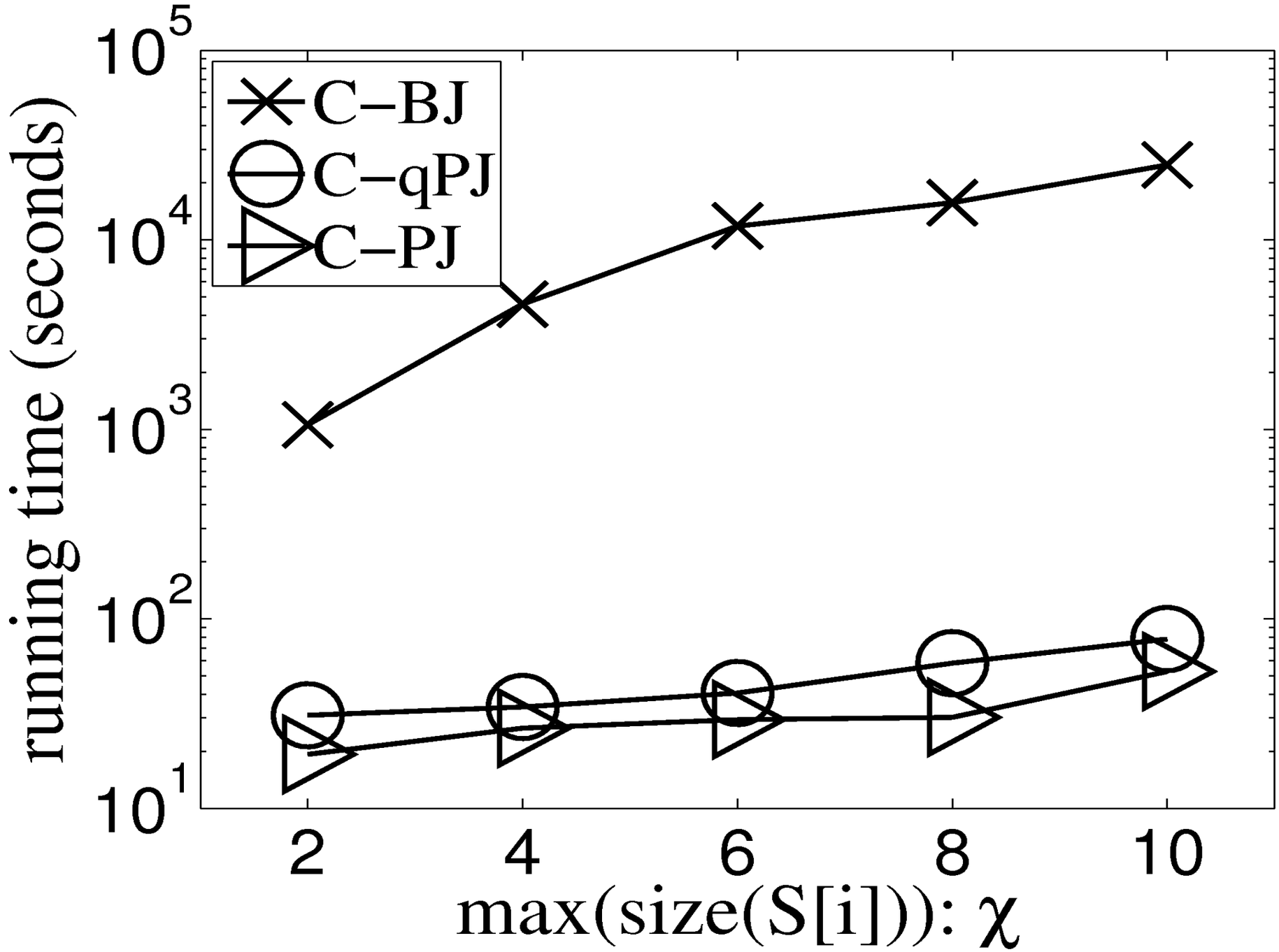
Character-level model, *Genome* dataset



Character-level model, *Genome* dataset



Character-level model, *Author2* dataset



The End

THANK YOU

Q and A

- The entire source code is available from a link at

  http:/ww2.cs.fsu.edu/~jestes

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

-
```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
3  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
4     AND R_q.cid=R.cid AND T_q.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7 ) AS L
8 WHERE L.rid=R.id AND L.tid=T.id
9 GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ
```

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
3  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
4     AND R_q.cid=R.cid AND T_q.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7 ) AS L
8 WHERE L.rid=R.id AND L.tid=T.id
9 GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ
```

String-Level Probabilistic Strings: q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}[|S_1|], \mathbf{E}[|S_2|])}{q} - \frac{\sum_{(\rho_1,\rho_2)\in G_{S_1}\cap G_{S_2}} p(\rho_1)p(\rho_2) + 1}{q}.$$

-
```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,Rq,Tq
3  WHERE Rq.g=Tq.g AND Rq.id=R.id AND Tq.id=T.id
4     AND Rq.cid=R.cid AND Tq.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7 ) AS L
8 WHERE L.rid=R.id AND L.tid=T.id
9 GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤  τ
```

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

- 1 SELECT R.id, T.id FROM R, T,
  2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
  3 WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
  4 AND R_q.cid=R.cid AND T_q.cid=T.cid
  5 GROUP BY R.id, T.id, R.len, T.len
  6 HAVING 1 + $\frac{1}{q}$*(max(R.len,T.len) - 1 -
  7 ub(R_q.cid,R.p,R_q.ℓ,R_q.g,T_q.cid,T.p,T_q.ℓ,T_q.g,τ))$\leq \tau$
  8) AS L
  9 WHERE L.rid=R.id AND L.tid=T.id
  10 GROUP BY R.id, T.id
  11 HAVING SUM(R.p*T.p*d(R.A,T.A)) $\leq$ τ

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

- ```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,Rq,Tq
3 WHERE Rq.g=Tq.g AND Rq.id=R.id AND Tq.id=T.id
4   AND Rq.cid=R.cid AND Tq.cid=T.cid
5 GROUP BY R.id, T.id, R.len, T.len
6   HAVING 1 + 1/q*(max(R.len,T.len) - 1 -
7 ub(Rq.cid,R.p,Rq.ℓ,Rq.g,Tq.cid,T.p,Tq.ℓ,Tq.g,τ))≤ τ
8) AS L
9 WHERE L.rid=R.id AND L.tid=T.id
10 GROUP BY R.id, T.id
11 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ
```
```

String-Level Probabilistic Strings: Improving q-Gram Lower Bounds

- For any string-level probabilistic strings S_1 and S_2,

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(\mathbf{E}(|S_1|), \mathbf{E}(|S_2|)) - \mathrm{UB}_\tau - 1}{q}.$$

- 1 SELECT R.id, T.id FROM R, T,
  2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
  3 WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
  4 AND R_q.cid=R.cid AND T_q.cid=T.cid
  5 GROUP BY R.id, T.id, R.len, T.len
  6 HAVING 1 + $\frac{1}{q}$*(max(R.len,T.len) - 1 -
  7 ub(R_q.cid,R.p,R_q.ℓ,R_q.g,T_q.cid,T.p,T_q.ℓ,T_q.g,τ))$\leq \tau$
  8) AS L
  9 WHERE L.rid=R.id AND L.tid=T.id
  10 GROUP BY R.id, T.id
  11 HAVING SUM(R.p*T.p*d(R.A,T.A)) $\leq$ τ

String-Level Probabilistic String Similiarity Join

```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
3  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
4     AND R_q.cid=R.cid AND T_q.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7  EXCEPT
8  SELECT R.id AS rid, T.id AS tid FROM R, T
9  GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*ABS(|R.A|-|T.A|)) > τ
11 ) AS L
12 WHERE L.rid=R.id AND L.tid=T.id
13 GROUP BY R.id, T.id
14 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ          (Q3)
```

String-Level Probabilistic String Similiarity Join

```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
3  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
4     AND R_q.cid=R.cid AND T_q.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7  EXCEPT
8  SELECT R.id AS rid, T.id AS tid FROM R, T
9  GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*ABS(|R.A|-|T.A|)) > τ
11 ) AS L
12 WHERE L.rid=R.id AND L.tid=T.id
13 GROUP BY R.id, T.id
14 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ          (Q3)
```

String-Level Probabilistic String Similiarity Join

```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
3  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
4     AND R_q.cid=R.cid AND T_q.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7  EXCEPT
8  SELECT R.id AS rid, T.id AS tid FROM R, T
9  GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*ABS(|R.A|-|T.A|)) > τ
11 ) AS L
12 WHERE L.rid=R.id AND L.tid=T.id
13 GROUP BY R.id, T.id
14 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ          (Q3)
```

String-Level Probabilistic String Similiarity Join

```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
3  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
4     AND R_q.cid=R.cid AND T_q.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7  EXCEPT
8  SELECT R.id AS rid, T.id AS tid FROM R, T
9  GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*ABS(|R.A|-|T.A|)) > τ
11 ) AS L
12 WHERE L.rid=R.id AND L.tid=T.id
13 GROUP BY R.id, T.id
14 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ          (Q3)
```

String-Level Probabilistic String Similiarity Join

```
1 SELECT R.id, T.id FROM R, T,
2 (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q
3  WHERE R_q.g=T_q.g AND R_q.id=R.id AND T_q.id=T.id
4     AND R_q.cid=R.cid AND T_q.cid=T.cid
5  GROUP BY R.id, T.id, R.len, T.len
6  HAVING 1+(max(R.len,T.len)-SUM(R.p*T.p)-1)/q≤ τ
7  EXCEPT
8  SELECT R.id AS rid, T.id AS tid FROM R, T
9  GROUP BY R.id, T.id
10 HAVING SUM(R.p*T.p*ABS(|R.A|-|T.A|)) > τ
11 ) AS L
12 WHERE L.rid=R.id AND L.tid=T.id
13 GROUP BY R.id, T.id
14 HAVING SUM(R.p*T.p*d(R.A,T.A)) ≤ τ          (Q3)
```

String-Level Probabilistic String Similiarity Join

- SELECT R.id, T.id FROM R, T,
  (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q,
    (···) AS L *(same as lines 2-11 in Q3)*
    WHERE L.rid=R.id AND L.tid=T.id AND R_q.g=T_q.g
      AND R_q.cid=R.cid AND T_q.cid=T.cid
      AND R_q.id=R.id AND T_q.id=T.id
  GROUP BY R.id, T.id, R.len, T.len
  HAVING 1 + $\frac{1}{q}$*($\max$(R.len,T.len) - 1 -
 ub(R_q.cid,R.p,R_q.ℓ,R_q.g,T_q.cid,T.p,T_q.ℓ,T_q.g,τ))$\leq \tau$
) AS L2
WHERE L2.rid=R.id AND L2.tid=T.id
GROUP BY R.id, T.id
HAVING SUM(R.p*T.p*d(R.A,T.A)) $\leq$ τ (Q4)

String-Level Probabilistic String Similiarity Join

- SELECT R.id, T.id FROM R, T,
  (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q,
    (···) AS L *(same as lines 2-11 in Q3)*
    WHERE L.rid=R.id AND L.tid=T.id AND R_q.g=T_q.g
      AND R_q.cid=R.cid AND T_q.cid=T.cid
      AND R_q.id=R.id AND T_q.id=T.id
    GROUP BY R.id, T.id, R.len, T.len
    HAVING 1 + $\frac{1}{q}$*($\max$(R.len,T.len) - 1 -
  ub(R_q.cid,R.p,R_q.ℓ,R_q.g,T_q.cid,T.p,T_q.ℓ,T_q.g,τ))$\leq \tau$
) AS L2
  WHERE L2.rid=R.id AND L2.tid=T.id
  GROUP BY R.id, T.id
  HAVING SUM(R.p*T.p*d(R.A,T.A)) $\leq$ τ (Q4)

String-Level Probabilistic String Similiarity Join

- SELECT R.id, T.id FROM R, T,
  (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q,
    (···) AS L *(same as lines 2-11 in Q3)*
    WHERE L.rid=R.id AND L.tid=T.id AND R_q.g=T_q.g
      AND R_q.cid=R.cid AND T_q.cid=T.cid
      AND R_q.id=R.id AND T_q.id=T.id
    GROUP BY R.id, T.id, R.len, T.len
    HAVING 1 + $\frac{1}{q}$*($\max$(R.len,T.len) - 1 -
  ub(R_q.cid,R.p,R_q.ℓ,R_q.g,T_q.cid,T.p,T_q.ℓ,T_q.g,τ))$\leq \tau$
) AS L2
  WHERE L2.rid=R.id AND L2.tid=T.id
  GROUP BY R.id, T.id
  HAVING SUM(R.p*T.p*d(R.A,T.A)) $\leq$ τ (Q4)

String-Level Probabilistic String Similiarity Join

- SELECT R.id, T.id FROM R, T,
  (SELECT R.id AS rid, T.id AS tid FROM R,T,R_q,T_q,
    (···) AS L *(same as lines 2-11 in Q3)*
    WHERE L.rid=R.id AND L.tid=T.id AND R_q.g=T_q.g
      AND R_q.cid=R.cid AND T_q.cid=T.cid
      AND R_q.id=R.id AND T_q.id=T.id
  GROUP BY R.id, T.id, R.len, T.len
  HAVING 1 + $\frac{1}{q}$*($\max$(R.len,T.len) - 1 -
 ub(R_q.cid,R.p,R_q.ℓ,R_q.g,T_q.cid,T.p,T_q.ℓ,T_q.g,τ))$\leq \tau$
) AS L2
WHERE L2.rid=R.id AND L2.tid=T.id
GROUP BY R.id, T.id
HAVING SUM(R.p*T.p*d(R.A,T.A)) $\leq$ τ (Q4)

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

- **Proof** Let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  By the deterministic **q-gram filtering** lemma we have,

$$\begin{aligned} & \sum_{s \in \Omega} w(s) |G_{\sigma_1} \cap G_{\sigma_2}| \\ \geq & \sum_{s \in \Omega} w(s)(\max(|\sigma_1|, |\sigma_2|) - 1 - q(d(s) - 1)) \end{aligned}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

- **Proof** Let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  By the deterministic **q-gram filtering** lemma we have,

$$\begin{aligned} & \sum_{s \in \Omega} w(s) |G_{\sigma_1} \cap G_{\sigma_2}| \\ \geq & \sum_{s \in \Omega} w(s) (\max(|\sigma_1|, |\sigma_2|) - 1 - q(d(s) - 1)) \\ = & \max(|S_1|, |S_2|) - 1 - q(\widehat{d}(S_1, S_2) - 1). \end{aligned}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1 and S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

- **Proof** Let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  By the deterministic **q-gram filtering** lemma we have,

$$\begin{aligned} & \boxed{\sum_{s \in \Omega} w(s) |G_{\sigma_1} \cap G_{\sigma_2}|} \\ \geq & \sum_{s \in \Omega} w(s)(\max(|\sigma_1|, |\sigma_2|) - 1 - q(d(s) - 1)) \\ = & \max(|S_1|, |S_2|) - 1 - q(\widehat{d}(S_1, S_2) - 1). \end{aligned}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  $\gamma_1 = (i, S_1[i..i+q-1]) \in G_{S_1}$
  $\gamma_2 = (j, S_2[j..j+q-1]) \in G_{S_2}$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Let $s \in \Omega$ and $s = ((\sigma_1, p_1), (\sigma_2, p_2))$
  $\gamma_1 = (i, S_1[i..i+q-1]) \in G_{S_1}$
  $\gamma_2 = (j, S_2[j..j+q-1]) \in G_{S_2}$

- Define $\gamma_1 =_s \gamma_2$ as the event that $S_1[i..i+q-1] = S_2[j..j+q-1]$ in s. We have

$$\Pr(\gamma_1 =_s \gamma_2) = \begin{cases} w(s), & \text{if } \sigma_1[i..i+q-1] = \sigma_2[j..j+q-1]; \\ 0, & \text{otherwise.} \end{cases}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Define $\gamma_1 =_s \gamma_2$ as the event that $S_1[i..i+q-1] = S_2[j..j+q-1]$ in s. We have

$$\Pr(\gamma_1 =_s \gamma_2) = \begin{cases} w(s), & \text{if } \sigma_1[i..i+q-1] = \sigma_2[j..j+q-1]; \\ 0, & \text{otherwise.} \end{cases}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Define $\gamma_1 =_s \gamma_2$ as the event that $S_1[i..i+q-1] = S_2[j..j+q-1]$ in s. We have

$$\Pr(\gamma_1 =_s \gamma_2) = \begin{cases} w(s), & \text{if } \sigma_1[i..i+q-1] = \sigma_2[j..j+q-1]; \\ 0, & \text{otherwise.} \end{cases}$$

- Recall that $|G_{\sigma_1} \cap G_{\sigma_2}|$ is the number of matching pairs of q-grams in G_{σ_1} and G_{σ_2}, so

$$\sum_{s \in \Omega} w(s)|G_{\sigma_1} \cap G_{\sigma_2}| = \sum_{s \in \Omega} \sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 =_s \gamma_2)$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Define $\gamma_1 =_s \gamma_2$ as the event that $S_1[i..i+q-1] = S_2[j..j+q-1]$ in s. We have

$$\Pr(\gamma_1 =_s \gamma_2) = \begin{cases} w(s), & \text{if } \sigma_1[i..i+q-1] = \sigma_2[j..j+q-1]; \\ 0, & \text{otherwise.} \end{cases}$$

- Recall that $|G_{\sigma_1} \cap G_{\sigma_2}|$ is the number of matching pairs of q-grams in G_{σ_1} and G_{σ_2}, so

$$\sum_{s\in\Omega} w(s)|G_{\sigma_1} \cap G_{\sigma_2}| = \sum_{s\in\Omega} \sum_{\substack{\gamma_1\in G_{S_1} \\ \gamma_2\in G_{S_2}}} \Pr(\gamma_1 =_s \gamma_2)$$

$$= \sum_{\substack{\gamma_1\in G_{S_1} \\ \gamma_2\in G_{S_2}}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2)$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- Define $\gamma_1 =_s \gamma_2$ as the event that $S_1[i..i+q-1] = S_2[j..j+q-1]$ in s. We have

$$\Pr(\gamma_1 =_s \gamma_2) = \begin{cases} w(s), & \text{if } \sigma_1[i..i+q-1] = \sigma_2[j..j+q-1]; \\ 0, & \text{otherwise.} \end{cases}$$

- Recall that $|G_{\sigma_1} \cap G_{\sigma_2}|$ is the number of matching pairs of q-grams in G_{σ_1} and G_{σ_2}, so

$$\sum_{s\in\Omega} w(s)|G_{\sigma_1} \cap G_{\sigma_2}| = \sum_{s\in\Omega} \sum_{\substack{\gamma_1\in G_{S_1} \\ \gamma_2\in G_{S_2}}} \Pr(\gamma_1 =_s \gamma_2)$$

$$= \sum_{\substack{\gamma_1\in G_{S_1} \\ \gamma_2\in G_{S_2}}} \sum_{s\in\Omega} \Pr(\gamma_1 =_s \gamma_2) = \sum_{\substack{\gamma_1\in G_{S_1} \\ \gamma_2\in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) \quad \square$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

```
1  SELECT R.id,T.id FROM R,T,
2  (SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
3   WHERE R_q.g=T_q.g AND R.id=R_q.id AND T.id=T_q.id
4   GROUP BY R.id, T.id, R.len, T.len
5   HAVING 1 + max(R.len,T.len)/q -
6    SUM( T_q.p*R_q.p)/q - 1/q ≤ τ
7   ) AS L
8   WHERE L.rid=R.id AND L.tid=T.id
9   AND ed(R.A,T.A)≤ τ
```

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

```
1  SELECT R.id,T.id FROM R,T,
2  (SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
3   WHERE R_q.g=T_q.g AND R.id=R_q.id AND T.id=T_q.id
4   GROUP BY R.id, T.id, R.len, T.len
5   HAVING 1 + max(R.len,T.len)/q -
6    SUM( T_q.p*R_q.p)/q - 1/q ≤ τ
7   ) AS L
8   WHERE L.rid=R.id AND L.tid=T.id
9   AND ed(R.A,T.A)≤ τ
```

Character-Level Probabilistic Strings: q-Gram Lower Bounds

- For any character-level probabilistic strings S_1, S_2:

$$\widehat{d}(S_1, S_2) \geq 1 + \frac{\max(|S_1|, |S_2|)}{q} - \frac{\sum_{\substack{\gamma_1 \in G_{S_1} \\ \gamma_2 \in G_{S_2}}} \Pr(\gamma_1 = \gamma_2) + 1}{q}$$

```
1  SELECT R.id,T.id FROM R,T,
2  (SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
3   WHERE R_q.g=T_q.g AND R.id=R_q.id AND T.id=T_q.id
4   GROUP BY R.id, T.id, R.len, T.len
5   HAVING 1 + max(R.len,T.len)/q -
6    SUM( T_q.p*R_q.p)/q - 1/q ≤ τ
7   ) AS L
8   WHERE L.rid=R.id AND L.tid=T.id
9   AND ed(R.A,T.A)≤ τ
```

Character-Level Probabilistic String Similarity Join

```
1  SELECT R.id,T.id FROM R,T,
2  (SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
3   WHERE R_q.g=T_q.g AND R.id=R_q.id AND T.id=T_q.id
4     AND ABS(R.len - T.len) ≤ τ
5   GROUP BY R.id, T.id, R.len, T.len
6   HAVING 1 + (max(R.len,T.len)-1)/q -
7    SUM( FLOOR(T_q.p*R_q.p)*
8      min(1, τ/max(1,ABS(R_q.ℓ - T_q.ℓ))) )/q -
9    SUM( CEILING(1-R_q.p*T_q.p)*T_q.p*R_q.p )/q ≤ τ
10  EXCEPT
11  SELECT R.id AS rid,T.id AS tid FROM R,T,R_q,T_q
12  WHERE R_q.g=T_q.g AND R_q.ℓ=T_q.ℓ AND R.id=R_q.id
13    AND T.id=T_q.id AND ABS(R.len - T.len) ≤ τ
14  GROUP BY R.id, T.id, R.len, T.len
15  HAVING max(R.len,T.len)+q-1-SUM(T_q.p*R_q.p)≤ τ
16 ) AS L
17 WHERE L.rid=R.id AND L.tid=T.id AND ld(R.A,T.A)
18  ≤ τ AND ud(R.A,T.A)> τ AND ed(R.A,T.A)≤ τ   (Q6)
```

Special Case: No Matching q-Grams

- Consider the string-level probabilistic strings:

$$\{(cat, 1)\} \qquad \{(dog, 1)\}$$

Special Case: No Matching q-Grams

- Consider the string-level probabilistic strings:

$$\{(cat, 1)\} \qquad \{(dog, 1)\}$$

- The string-level probabilistic 2-grams are:

$$\{(1, 1, 1, \#c), (1, 1, 2, ca), (1, 1, 3, at), (1, 1, 4, t\$)\}$$

$$\{(1, 1, 1, \#d), (1, 1, 2, do), (1, 1, 3, og), (1, 1, 4, g\$)\}$$

Special Case: No Matching q-Grams

- Consider the string-level probabilistic strings:

$$\{(cat, 1)\} \qquad \{(dog, 1)\}$$

- The string-level probabilistic 2-grams are:

$$\{(1, 1, 1, \#c), (1, 1, 2, ca), (1, 1, 3, at), (1, 1, 4, t\$)\}$$

$$\{(1, 1, 1, \#d), (1, 1, 2, do), (1, 1, 3, og), (1, 1, 4, g\$)\}$$

- Our queries would miss the pair (cat,dog) when $\tau = 3$

Special Case: No Matching q-Grams

- Consider the string-level probabilistic strings:

$$\{(cat, 1)\} \qquad \{(dog, 1)\}$$

- The string-level probabilistic 2-grams are:

$$\{(1, 1, 1, \#c), (1, 1, 2, ca), (1, 1, 3, at), (1, 1, 4, t\$)\}$$

$$\{(1, 1, 1, \#d), (1, 1, 2, do), (1, 1, 3, og), (1, 1, 4, g\$)\}$$

- Our queries would miss the pair (cat,dog) when $\tau = 3$

  Solution: Add special system choices and q-grams

$$\{(\#\$, 0), (cat, 1)\} \quad \{(\#\$, 0), (dog, 1)\}$$

$$\{(0, 0, 0, \#\$), (1, 1, 1, \#c), (1, 1, 2, ca), (1, 1, 3, at), (1, 1, 4, t\$)\}$$

$$\{(0, 0, 0, \#\$), (1, 1, 1, \#d), (1, 1, 2, do), (1, 1, 3, og), (1, 1, 4, g\$)\}$$

String-level model, *Author1* dataset