The Query Complexity of Edit Distance

Krzysztof Onak
MIT

Joint work with:

Alexandr Andoni (Princeton/CCI)
Robert Krauthgamer (Weizmann Institute)

Edit Distance (or Levenshtein Distance)

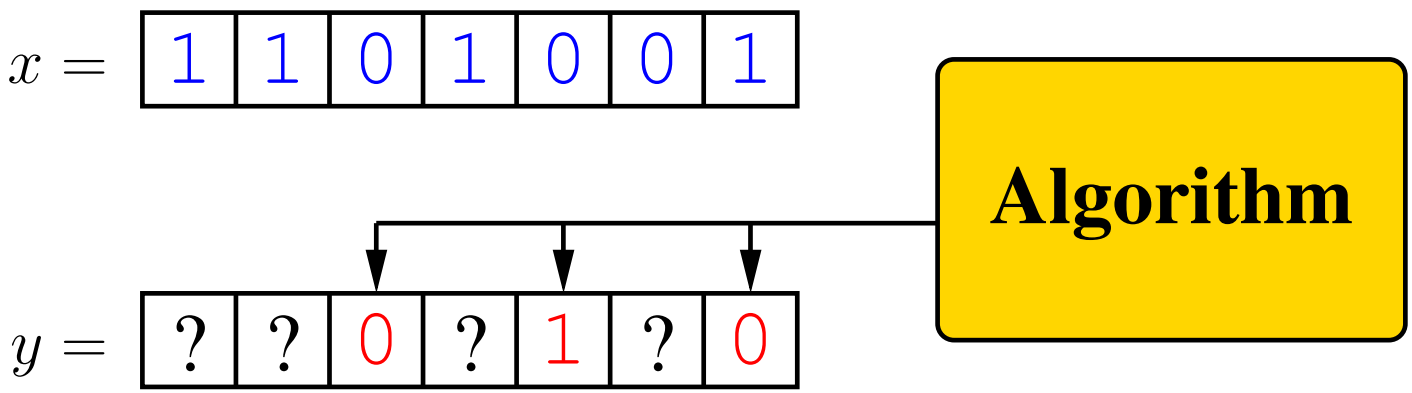
$\mathrm{ed}(x, y) =$ number of deletions, insertions, and substitutions to transform x into y

algorithm
algerithm
algebrithm
algebrathm
algebrath
algebrah
algebra

$\mathrm{ed}(\text{algorithm}, \text{algebra}) = 6$

The Model

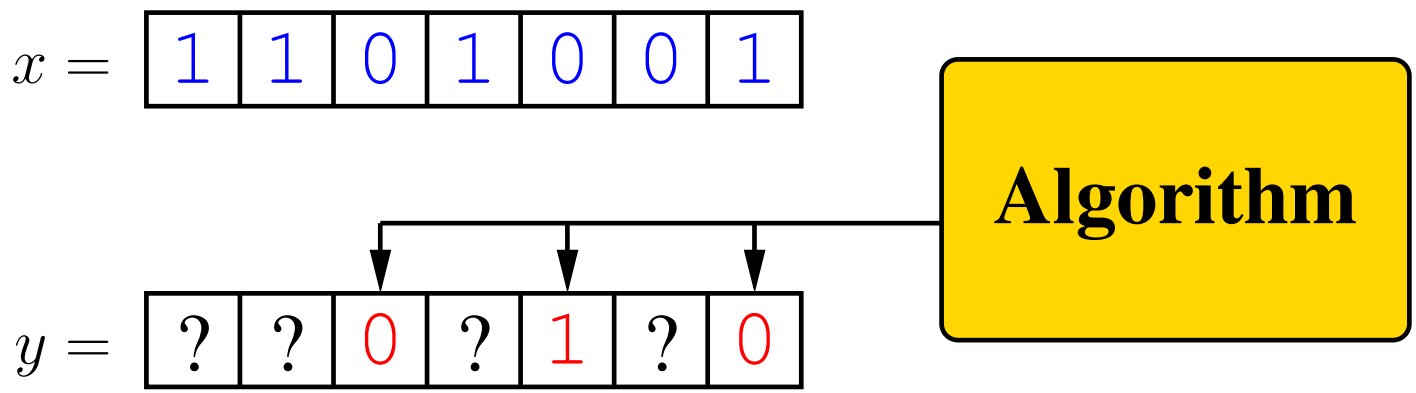
- Input:
  - two strings x and y of length n
  - x is known to the algorithm
  - y is not known, the algorithm can query it



The Model

- Input:
  - two strings x and y of length n
  - x is known to the algorithm
  - y is not known, the algorithm can query it

- Sample question: How many queries are necessary to tell $\mathrm{ed}(x,y) \leq .2n$ from $\mathrm{ed}(x,y) \geq .6n$?



Lower Bound

- Telling edit distance $\leq .2n$ from $\geq .6n$:

$$2^{\Omega\left(\frac{\log n}{\log \log n}\right)} \text{ queries}$$

Lower Bound

- Telling edit distance $\leq .2n$ from $\geq .6n$:

$$2^{\Omega\left(\frac{\log n}{\log\log n}\right)} \text{ queries}$$

- Telling edit distance $O(n/\alpha)$ from $\Omega(n)$:

$$2^{\Omega\left(\frac{\log n}{\log\alpha+\log\log n}\right)} \text{ queries}$$

Edit Distance vs. Ulam Distance

- Ulam distance:
  - $\mathrm{lcs}(x, y) =$ longest common subsequence length
  - Ulam distance defined for strings with no element appearing twice
  - $\mathrm{Ulam}(x, y) = |x| + |y| - 2 \cdot \mathrm{lcs}(x, y)$

Edit Distance vs. Ulam Distance

- Ulam distance:
  - $\mathrm{lcs}(x, y) =$ longest common subsequence length
  - Ulam distance defined for strings with no element appearing twice
  - $\mathrm{Ulam}(x, y) = |x| + |y| - 2 \cdot \mathrm{lcs}(x, y)$
- Exact algorithms:
  - Ulam distance: $O(n \log n)$ via patience sorting
  - Edit distance: $O((n/\log n)^2)$ for binary alphabet [Masek, Paterson 1980]

Edit Distance vs. Ulam Distance

- Ulam distance:
  - $\mathrm{lcs}(x, y) =$ longest common subsequence length
  - Ulam distance defined for strings with no element appearing twice
  - $\mathrm{Ulam}(x, y) = |x| + |y| - 2 \cdot \mathrm{lcs}(x, y)$
- Exact algorithms:
  - Ulam distance: $O(n \log n)$ via patience sorting
  - Edit distance: $O((n/\log n)^2)$ for binary alphabet [Masek, Paterson 1980]
- Number of queries to tell distance $\leq .2n$ from $\geq .6n$
  - Ulam distance: $O(\log n)$ [Ailon, Chazelle, Comandur, Liu]
  - Edit distance: $2^{\Omega(\frac{\log n}{\log \log n})}$

Edit Distance vs. Ulam Distance

- Ulam distance:
  - $\mathrm{lcs}(x, y) =$ longest common subsequence length
  - Ulam distance defined for strings with no element appearing twice
  - $\mathrm{Ulam}(x, y) = |x| + |y| - 2 \cdot \mathrm{lcs}(x, y)$
- Exact algorithms:
  - Ulam distance: $O(n \log n)$ via patience sorting
  - Edit distance: $O((n/\log n)^2)$ for binary alphabet [Masek, Paterson 1980]
- Number of queries to tell distance $\leq .2n$ from $\geq .6n$
  - Ulam distance: $O(\log n)$ [Ailon, Chazelle, Comandur, Liu]
  - Edit distance: $2^{\Omega(\frac{\log n}{\log \log n})}$
- First separation between the two

Construction

- Need $\Omega(\log n)$ queries to tell apart:
  1. Close:
     $x =$ random string
     $y = x$ shifted by random offset in [0,n/100]

Construction

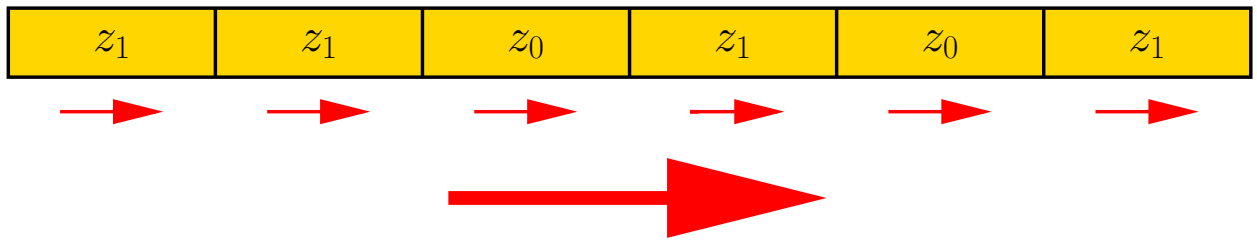
- Need $\Omega(\log n)$ queries to tell apart:
  1. Close:
     $x =$ random string
     $y = x$ shifted by random offset in [0,n/100]
  2. Far:
     $x =$ random string
     $y =$ random string

Construction

- Need $\Omega(\log n)$ queries to tell apart:
  1. Close:
     $x =$ random string
     $y = x$ shifted by random offset in [0,n/100]
  2. Far:
     $x =$ random string
     $y =$ random string
- $O(\log n)$ queries sufficient, need better construction

Construction

- Need $\Omega(\log n)$ queries to tell apart:
  1. Close:
     $x =$ random string
     $y = x$ shifted by random offset in [0,n/100]
  2. Far:
     $x =$ random string
     $y =$ random string
- $O(\log n)$ queries sufficient, need better construction
- Solution: Recursion
  - Fix random z_0 and z_1
  - Replace every 0 with z_0 and every 1 with z_1, always shift at random by a little bit



Construction

- Need $\Omega(\log n)$ queries to tell apart:
  1. Close:
     $x =$ random string
     $y = x$ shifted by random offset in [0,n/100]
  2. Far:
     $x =$ random string
     $y =$ random string
- $O(\log n)$ queries sufficient, need better construction
- Solution: Recursion
  - Fix random z_0 and z_1
  - Replace every 0 with z_0 and every 1 with z_1, always shift at random by a little bit
  - Will need $\Omega(\log^2 n)$ queries

Upper Bounds

- We also have some upper bounds

Upper Bounds

- We also have some upper bounds
- Example: telling edit distance $O(n/2^{\sqrt{\log n}})$ from $\Omega(n)$
  - our algorithm makes $2^{O(\sqrt{\log n})}$ queries
  - lower bound $2^{\Omega(\sqrt{\log n})}$

Upper Bounds

- We also have some upper bounds
- Example: telling edit distance $O(n/2^{\sqrt{\log n}})$ from $\Omega(n)$
  - our algorithm makes $2^{O(\sqrt{\log n})}$ queries
  - lower bound $2^{\Omega(\sqrt{\log n})}$

Stay tuned!!!

The Other Model

What if neither of the strings known?

The Other Model

What if neither of the strings known?

- Batu, Ergün, Kilian, Magen, Raskhodnikova, Rubinfeld, Sami 2003:

can tell edit distance $O(n^\alpha)$ from $\Omega(n)$
with $\tilde{O}(n^{\alpha/2} + n^{2\alpha-1})$ queries

The Other Model

What if neither of the strings known?

- Batu, Ergün, Kilian, Magen, Raskhodnikova, Rubinfeld, Sami 2003:

  can tell edit distance $O(n^\alpha)$ from $\Omega(n)$
  with $\tilde{O}(n^{\alpha/2} + n^{2\alpha-1})$ queries

- Andoni, O. 2009:

  can tell edit distance $O(n^\alpha)$ from $\Omega(n^\beta)$
  with $O(n^{\alpha+2(1-\beta)+o(1)})$ queries

The Other Model

What if neither of the strings known?

- Batu, Ergün, Kilian, Magen, Raskhodnikova, Rubinfeld, Sami 2003:

  can tell edit distance $O(n^\alpha)$ from $\Omega(n)$
  with $\tilde{O}(n^{\alpha/2} + n^{2\alpha-1})$ queries

- Andoni, O. 2009:

  can tell edit distance $O(n^\alpha)$ from $\Omega(n^\beta)$
  with $O(n^{\alpha+2(1-\beta)+o(1)})$ queries

- Andoni, Nguyen 2010:

  near optimal sublinear-time algorithm for Ulam distance

The Other Model

What if neither of the strings known?

- Batu, Ergün, Kilian, Magen, Raskhodnikova, Rubinfeld, Sami 2003:

  can tell edit distance $O(n^\alpha)$ from $\Omega(n)$
  with $\tilde{O}(n^{\alpha/2} + n^{2\alpha-1})$ queries

- Andoni, O. 2009:

  can tell edit distance $O(n^\alpha)$ from $\Omega(n^\beta)$
  with $O(n^{\alpha+2(1-\beta)+o(1)})$ queries

- Andoni, Nguyen 2010:

  near optimal sublinear-time algorithm for Ulam distance

- Exact query complexity of edit distance still open

Thank you!