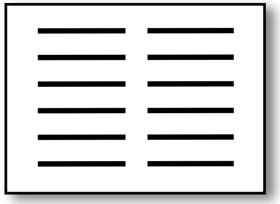
Translation by Pattern Matching

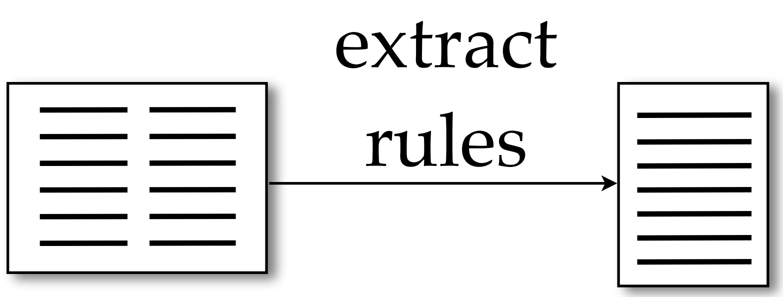
Adam Lopez
University of Edinburgh

Statistical Machine Translation



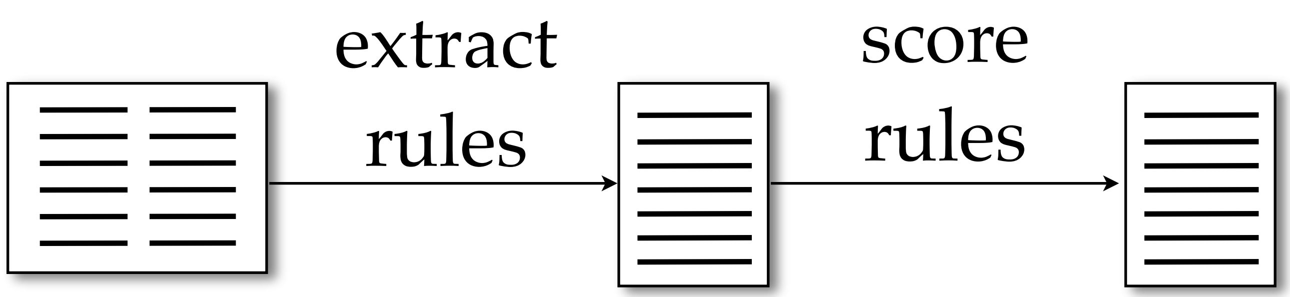
parallel text +
alignment

Statistical Machine Translation



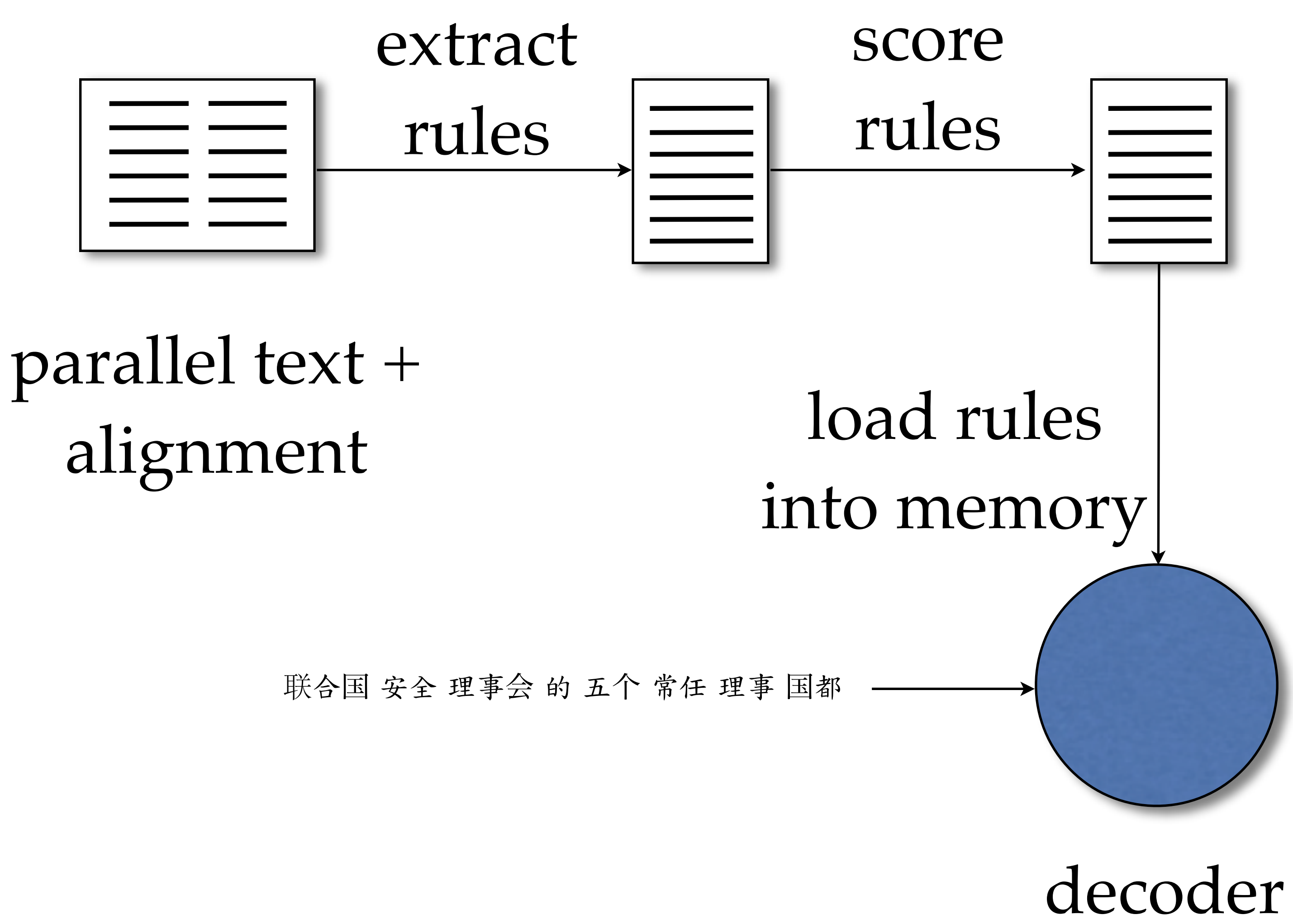
parallel text +
alignment

Statistical Machine Translation

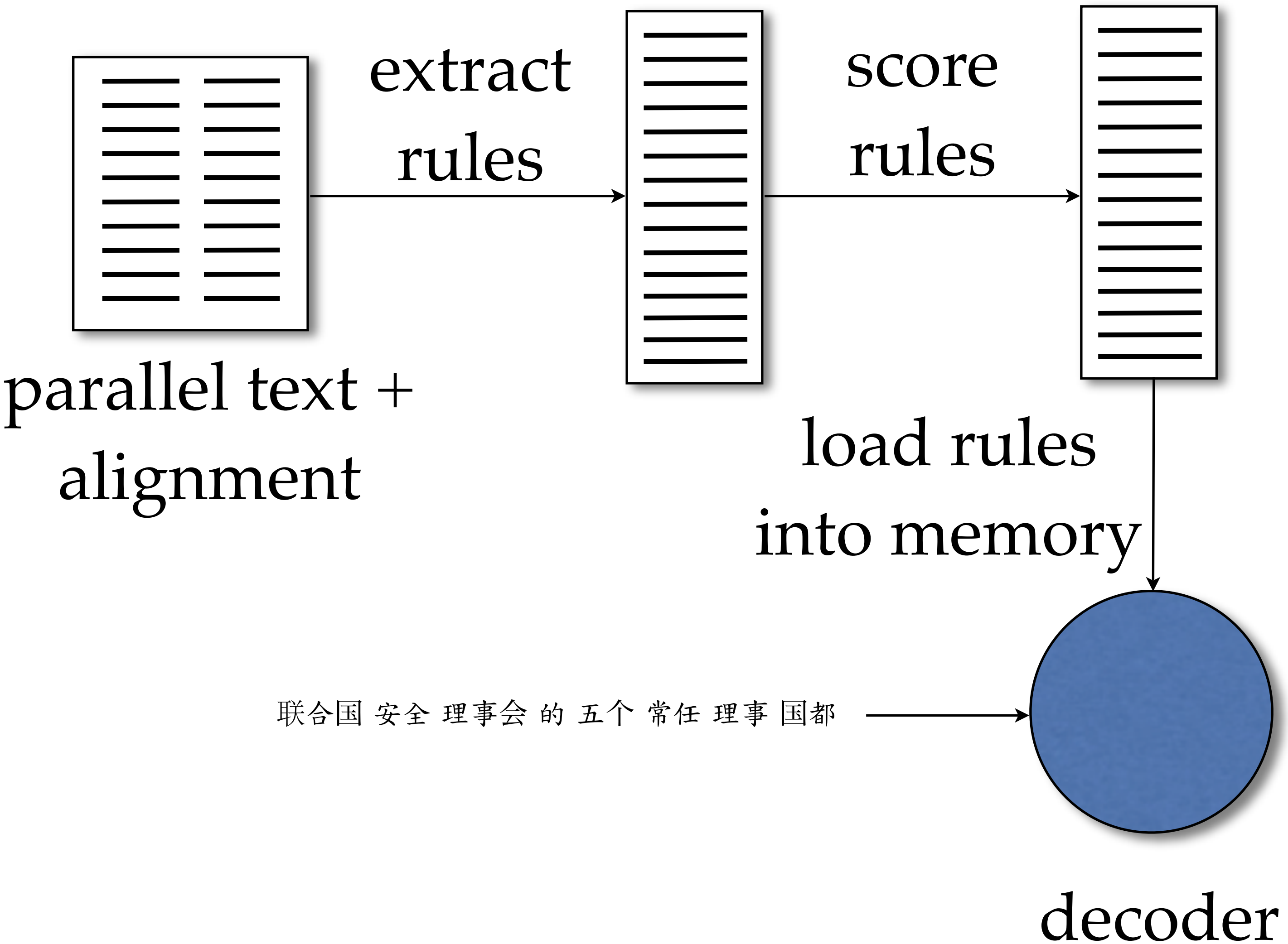


parallel text +
alignment

Statistical Machine Translation

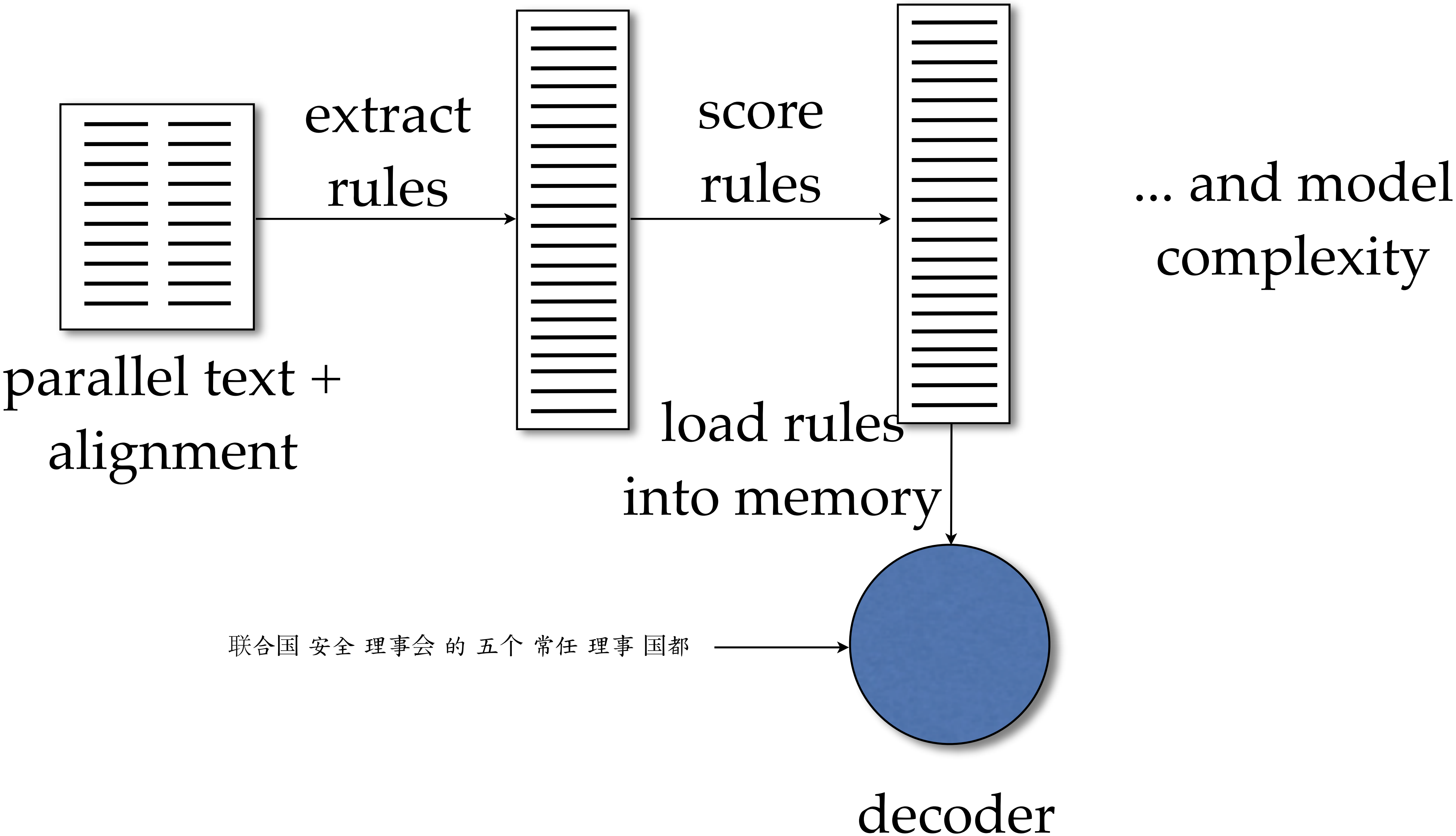


Statistical Machine Translation

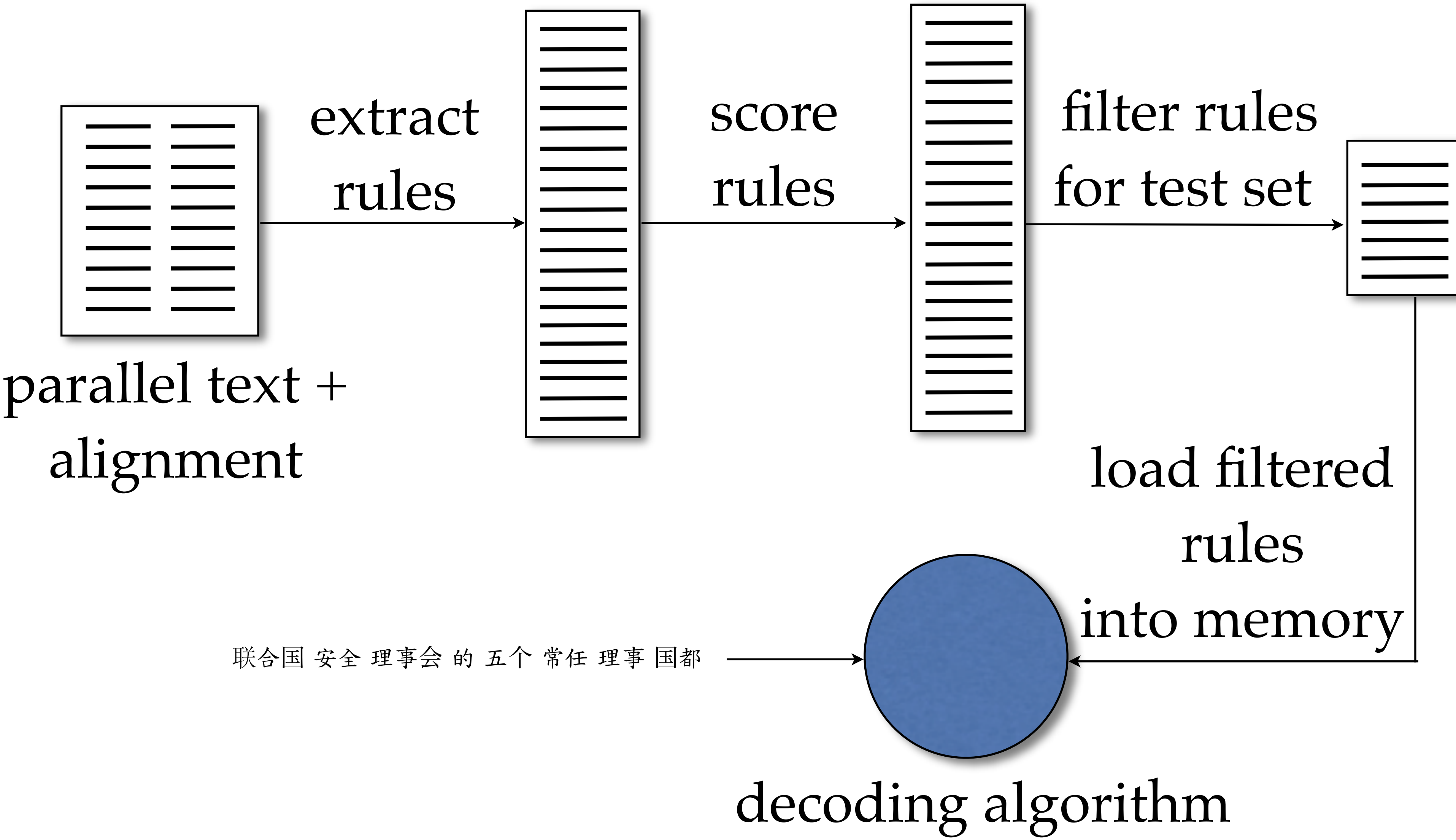


number of rules depends on corpus size...

Statistical Machine Translation



Statistical Machine Translation

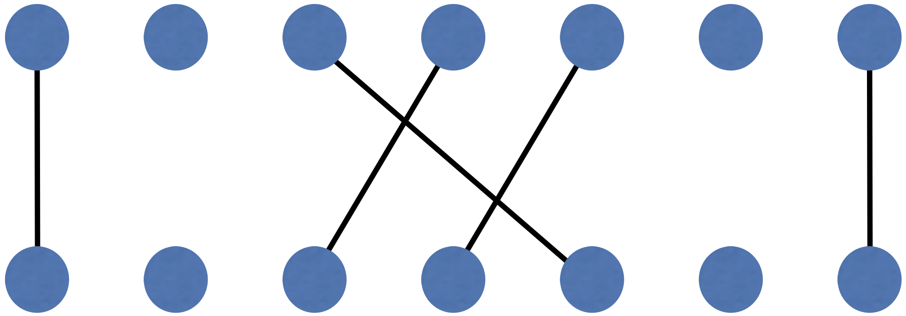


Baseline Translation Model

- Hierarchical Phrase-based translation (Chiang 2007)
- 1M parallel sentences (27M words)
- GIZA++ alignments (Och & Ney 2003, Koehn et al. 2003)
  - alignments are *dense*
- Heuristics used to restrict number of extracted rules
- 67M rules, 6.1Gb of data
  - cf. 225M (Zens & Ney 2007), 55M (DeNeefe et al. 2007)

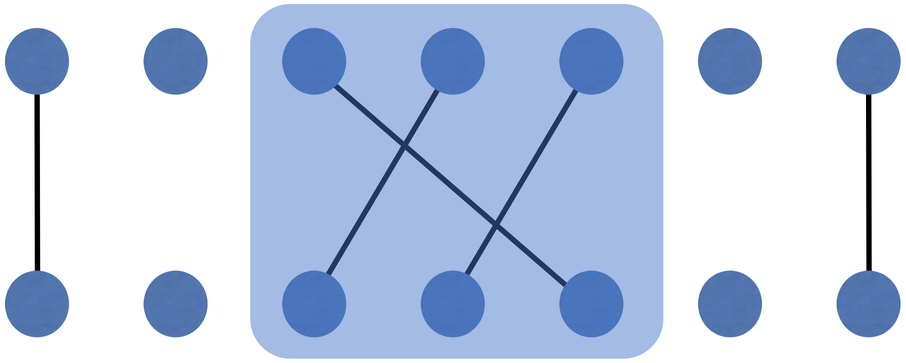
Some Possible Improvements

- 3.5M sentences (2.5M out-of-domain), 100M words
- Discriminatively trained alignments (Ayan & Dorr 2006)
  - Key difference: alignments are *sparse*
- *Loose* phrase extraction (Ayan & Dorr 2006)



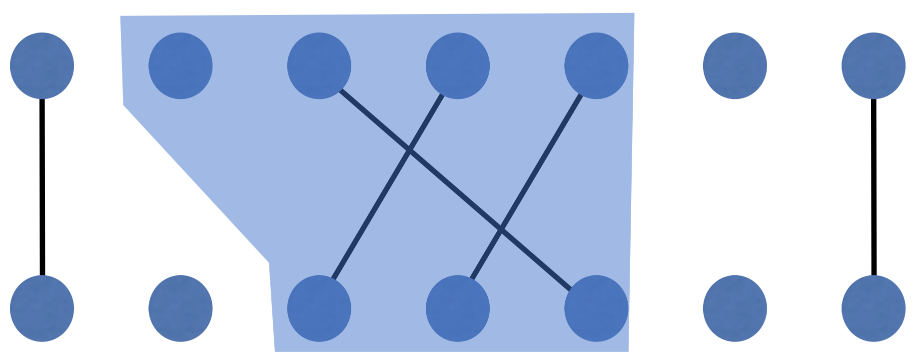
Some Possible Improvements

- 3.5M sentences (2.5M out-of-domain), 100M words
- Discriminatively trained alignments (Ayan & Dorr 2006)
    - Key difference: alignments are *sparse*
- *Loose* phrase extraction (Ayan & Dorr 2006)



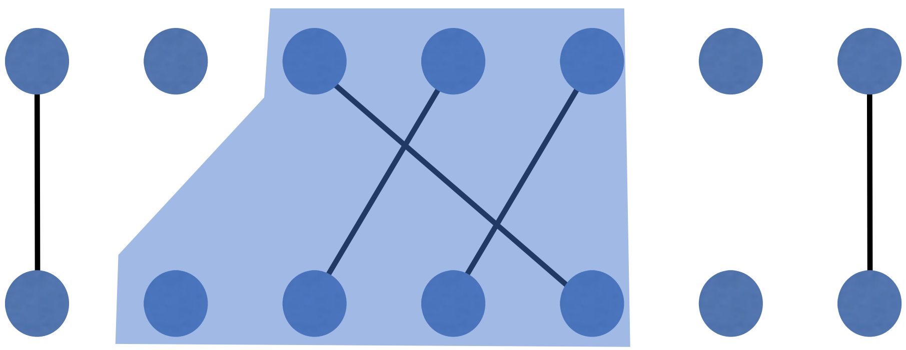
Some Possible Improvements

- 3.5M sentences (2.5M out-of-domain), 100M words
- Discriminatively trained alignments (Ayan & Dorr 2006)
  - Key difference: alignments are *sparse*
- *Loose* phrase extraction (Ayan & Dorr 2006)



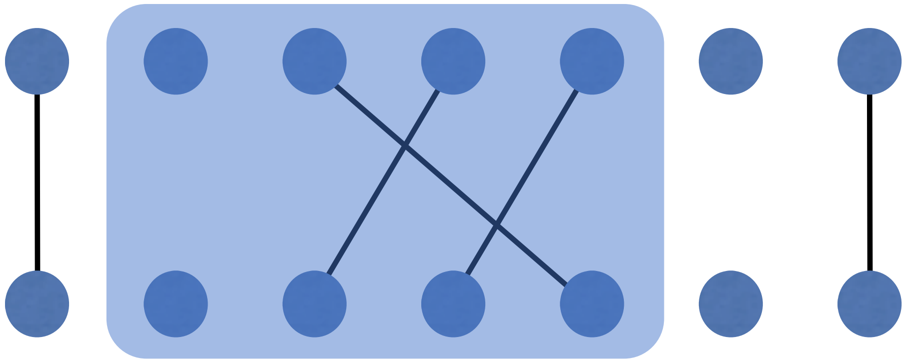
Some Possible Improvements

- 3.5M sentences (2.5M out-of-domain), 100M words
- Discriminatively trained alignments (Ayan & Dorr 2006)
  - Key difference: alignments are *sparse*
- *Loose* phrase extraction (Ayan & Dorr 2006)



Some Possible Improvements

- 3.5M sentences (2.5M out-of-domain), 100M words
- Discriminatively trained alignments (Ayan & Dorr 2006)
  - Key difference: alignments are *sparse*
- *Loose* phrase extraction (Ayan & Dorr 2006)



Some Possible Improvements

- Rule extraction time: 77 CPU days
  - does not include sorting or scoring!
- Rules counted: 20 billion
  - 2 orders of magnitude larger than state of the art
- Estimated unique rules: 6.6 billion
- Estimated extract file size: 917Gb
- Estimated phrase table size: 600Gb

The Problem

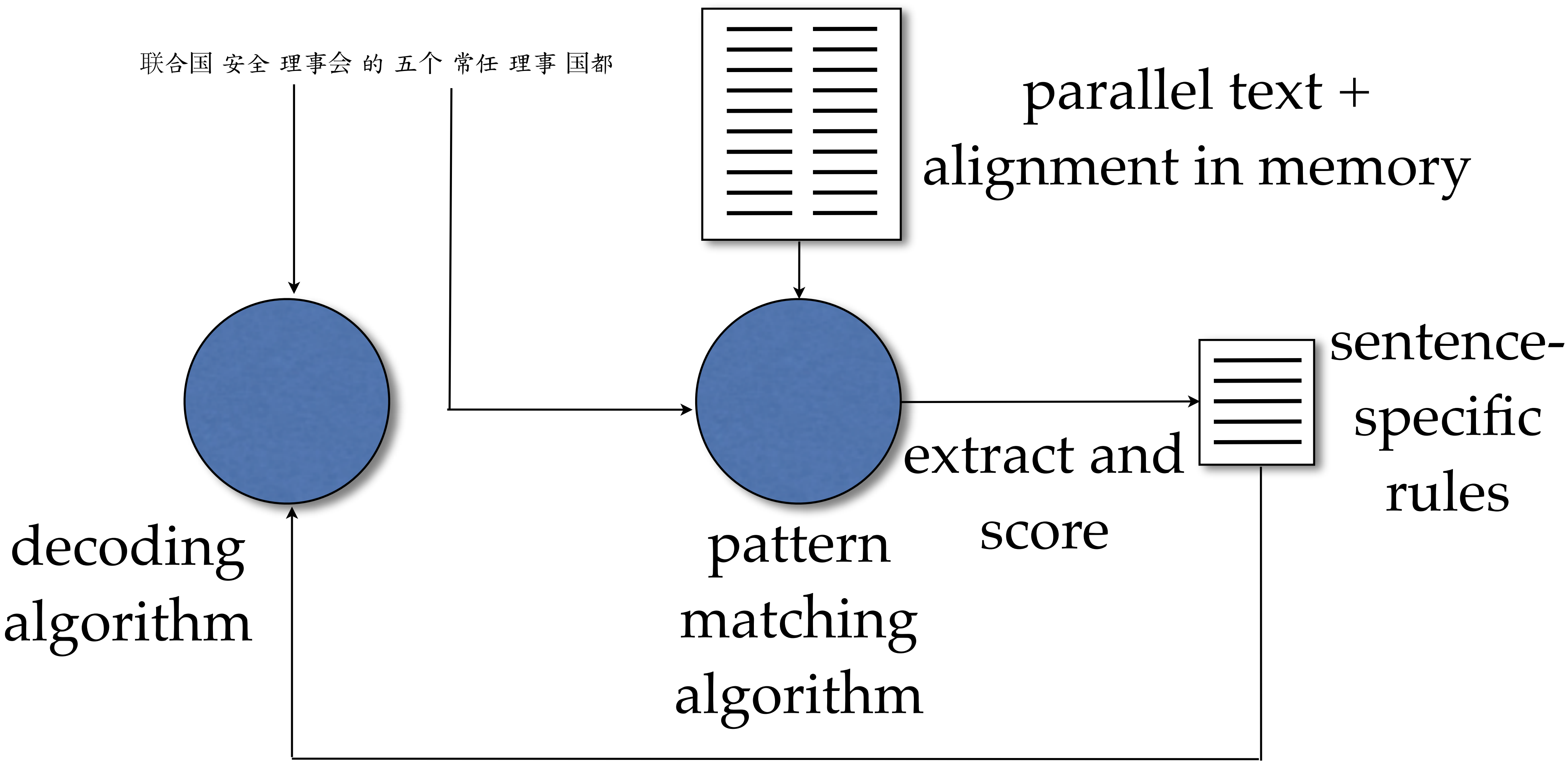
- Current models are bounded by resource limitations.
- We're already pushing the edge of what's possible.
- Parallel data aren't getting any smaller.
- Models aren't getting any less complex.

The Solution

- *Translation by pattern matching*.
- Novel pattern matching algorithms.
  - Exploit ideas developed in bioinformatics, IR
- Support for tera-scale translation models.

Idea: Translation by Pattern Matching

(Callison-Burch et al. 05, Zhang & Vogel 05)



Exact Pattern Matching

Input Pattern   it persuades him and it disheartens him

Exact Pattern Matching

Input Pattern =Query Pattern

it persuades him and it disheartens him

Pattern Matching for Phrase-Based MT

Input Pattern it persuades him and it disheartens him

Pattern Matching for Phrase-Based MT

Input Pattern it persuades him and it disheartens him

Query Patterns

it
persuades
him
and
disheartens
it persuades
persuades him
him and
and it
it disheartens
disheartens him

it persuades him
persuades him and
him and it
and it disheartens
it disheartens him
it persuades him and
persuades him and it
him and it disheartens
and it disheartens him
it persuades him and it
persuades him and it disheartens
him and it disheartens him

Suffix Arrays

it	makes	him	and	it	mars	him	,	it	sets	him	on	and	it	takes	him	off	.	#
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

Text T

Suffix Arrays

it makes him and it mars him , it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

4 it mars him , it sets him on and it takes him off . #

Suffix 4

Suffix Arrays

it makes him and it mars him , it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

0 it makes him and it mars him , it sets him on and it takes him ...
1 makes him and it mars him , it sets him on and it takes him off . #
2 him and it mars him , it sets him on and it takes him off . #
3 and it mars him , it sets him on and it takes him off . #
4 it mars him , it sets him on and it takes him off . #
5 mars him , it sets him on and it takes him off . #
6 him , it sets him on and it takes him off . #
7 , it sets him on and it takes him off . #
...

Suffix Arrays

it makes him and it mars him , it sets him on and it takes him off . #
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

3 and it mars him , it sets him on and it takes him off . #
12 and it takes him off . #
2 him and it mars him , it sets him on and it takes him off . #
15 him off . #
10 him on and it takes him off . #
6 him , it sets him on and it takes him off . #
0 it makes him and it mars him , it sets him on and it takes him ...
4 it mars him , it sets him on and it takes him off . #
...

Suffix Arrays

it makes him and it mars him , it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

3	and it mars him , it sets him on and it takes him off . #
12	and it takes him off . #
2	him and it mars him , it sets him on and it takes him off . #
15	him off . #
10	him on and it takes him off . #
6	him , it sets him on and it takes him off . #
0	it makes him and it mars him , it sets him on and it takes him ...
4	it mars him , it sets him on and it takes him off . #
⋮	...

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA

him and it

Query Pattern w

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA

him and it

Query Pattern w

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA

him and it

Query Pattern w

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA

him and it

Query Pattern w

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA $O(|w| \log |T|)$

him and it

Query Pattern w

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA

$O(|w| \log |T|)$

$O(|w| + \log |T|)$ (Manber & Myers, 93)

him and it

Query Pattern w

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array SA

$O(|w| \log |T|)$

$O(|w| + \log |T|)$ (Manber & Myers, 93)

him and it

$O(|w|)$ (Abouelhoda et al., 04)

Query Pattern w

Suffix Arrays

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text T

3	12	2	15	10	6	0	4	8	13	1	5	16	11	9	14	7	17	18

Suffix Array *SA*

him and it

Query Pattern w

$O(|w| \log |T|)$

$O(|w| + \log |T|)$ (Manber & Myers, 93)

$O(|w|)$ (Abouelhoda et al., 04)

on baseline model:

0.009 seconds / sentence

(not including extraction / scoring)

Problem: Phrases with Gaps

- Hierarchical phrase-based translation (Chiang 2005, 2007)
- Quirk et al. 2005, Simard et al. 2005, DeNeefe et al. 2007

Input	it persuades him and it disheartens him
Source Phrase	it X him

Hierarchical Phrases: Phrases with Gaps

- Hierarchical phrase-based translation (Chiang 2005, 2007)
- Quirk et al. 2005, Simard et al. 2005, DeNeefe et al. 2007

Input it persuades him and it disheartens him

Source Phrase it X him

Hierarchical Phrases: Phrases with Gaps

- Hierarchical phrase-based translation (Chiang 2005, 2007)
- Quirk et al. 2005, Simard et al. 2005, DeNeefe et al. 2007

Input	it persuades him and it disheartens him
Source Phrase	it X him

Hierarchical Phrases: Phrases with Gaps

- Hierarchical phrase-based translation (Chiang 2005, 2007)
- Quirk et al. 2005, Simard et al. 2005, DeNeefe et al. 2007

Input	it persuades him and it disheartens him
Source Phrase	it X him

Hierarchical Phrases: Phrases with Gaps

- Hierarchical phrase-based translation (Chiang 2005, 2007)
- Quirk et al. 2005, Simard et al. 2005, DeNeefe et al. 2007

Input — it persuades him and it disheartens him

Source Phrase — it X and X him

Problem Statement

Given an input sentence, efficiently find all hierarchical phrase-based translation rules for that sentence in the training corpus.

Pattern Matching for Hierachical PBMT

Input Pattern it persuades him and it disheartens him

Pattern Matching for Hierarchical PBMT

Input Pattern: it persuades him and it disheartens him

Query Patterns:

it
persuades
him
and
disheartens
it persuades
persuades him
him and
and it
it disheartens
disheartens him

it persuades him
persuades him and
him and it
and it disheartens
it disheartens him
it persuades him and
persuades him and it
him and it disheartens
and it disheartens him
it persuades him and it
persuades him and it disheartens
him and it disheartens him

Pattern Matching for Hierarchical PBMT

Input Pattern it persuades him and it disheartens him

Query Patterns

it X and
it X it
it X disheartens
it X him
persuades X it
persuades X disheartens
persuades X him
it persuades X it
it persuades X disheartens
it persuades X him
it X and it
it X it disheartens

it X disheartens him
it X and X him
persuades him X disheartens
persuades him X him
persuades X it disheartens
persuades X disheartens him
him and X him
him X disheartens him
it persuades him X disheartens
it persuades him X him
it persuades X it disheartens
it persuades X disheartens him

Pattern Matching for Hierarchical PBMT

Input Pattern it persuades him and it disheartens him

Query Patterns

it X and it disheartens
it X it disheartens him
persuades him and X him
persuades him X disheartens him
persuades X it disheartens him
it persuades him and X him
it persuades him X disheartens him
it persuades X it disheartens him
it X and it disheartens him

Pattern Matching for Hierarchical PBMT

Input Pattern it persuades him and it disheartens him

Query Patterns

it X and it disheartens
it X it disheartens him
persuades him and X him
persuades him X disheartens him
persuades X it disheartens him
it persuades him and X him
it persuades him X disheartens him
it persuades X it disheartens him
it X and it disheartens him

This is a variant of *approximate* pattern matching (Navarro '01)

Pattern Matching with Gaps

Query pattern α

him X it

3	and it mars him , it sets him ...
12	and it takes him off . #
2	him and it mars him . it sets ...
15	him off . #
10	him on and it takes him off . #
6	him , it sets him on and it ...
0	it makes him and it mars ...
4	it mars him , it sets him on ...
8	it sets him on and it takes ...
13	it takes him off . #
1	makes him and it mars him ...
⋮	

Pattern Matching with Gaps

Query pattern α

him X it

3	and it mars him , it sets him ...
12	and it takes him off . #
2	him and it mars him . it sets ...
15	him off . #
10	him on and it takes him off . #
6	him , it sets him on and it ...
0	it makes him and it mars ...
4	it mars him , it sets him on ...
8	it sets him on and it takes ...
13	it takes him off . #
1	makes him and it mars him ...
⋮	

Pattern Matching with Gaps

Query pattern α

him X it

3	and it mars him , it sets him ...
12	and it takes him off . #
2	him and it mars him . it sets ...
15	him off . #
10	him on and it takes him off . #
6	him , it sets him on and it ...
0	it makes him and it mars ...
4	it mars him , it sets him on ...
8	it sets him on and it takes ...
13	it takes him off . #
1	makes him and it mars him ...
⋮	

Pattern Matching with Gaps

Query pattern α

him X it

Subpatterns w_i

him
it

3	and it mars him , it sets him ...
12	and it takes him off . #
2	him and it mars him . it sets ...
15	him off . #
10	him on and it takes him off . #
6	him , it sets him on and it ...
0	it makes him and it mars ...
4	it mars him , it sets him on ...
8	it sets him on and it takes ...
13	it takes him off . #
1	makes him and it mars him ...
⋮	

Pattern Matching with Gaps

Query pattern α

him X it

Subpatterns w_i

him

it

3	and it mars him , it sets him ...
12	and it takes him off . #
2	him and it mars him . it sets ...
15	him off . #
10	him on and it takes him off . #
6	him , it sets him on and it ...
0	it makes him and it mars ...
4	it mars him , it sets him on ...
8	it sets him on and it takes ...
13	it takes him off . #
1	makes him and it mars him ...
⋮	

Pattern Matching with Gaps

Query pattern α

him X it

Subpatterns w_i

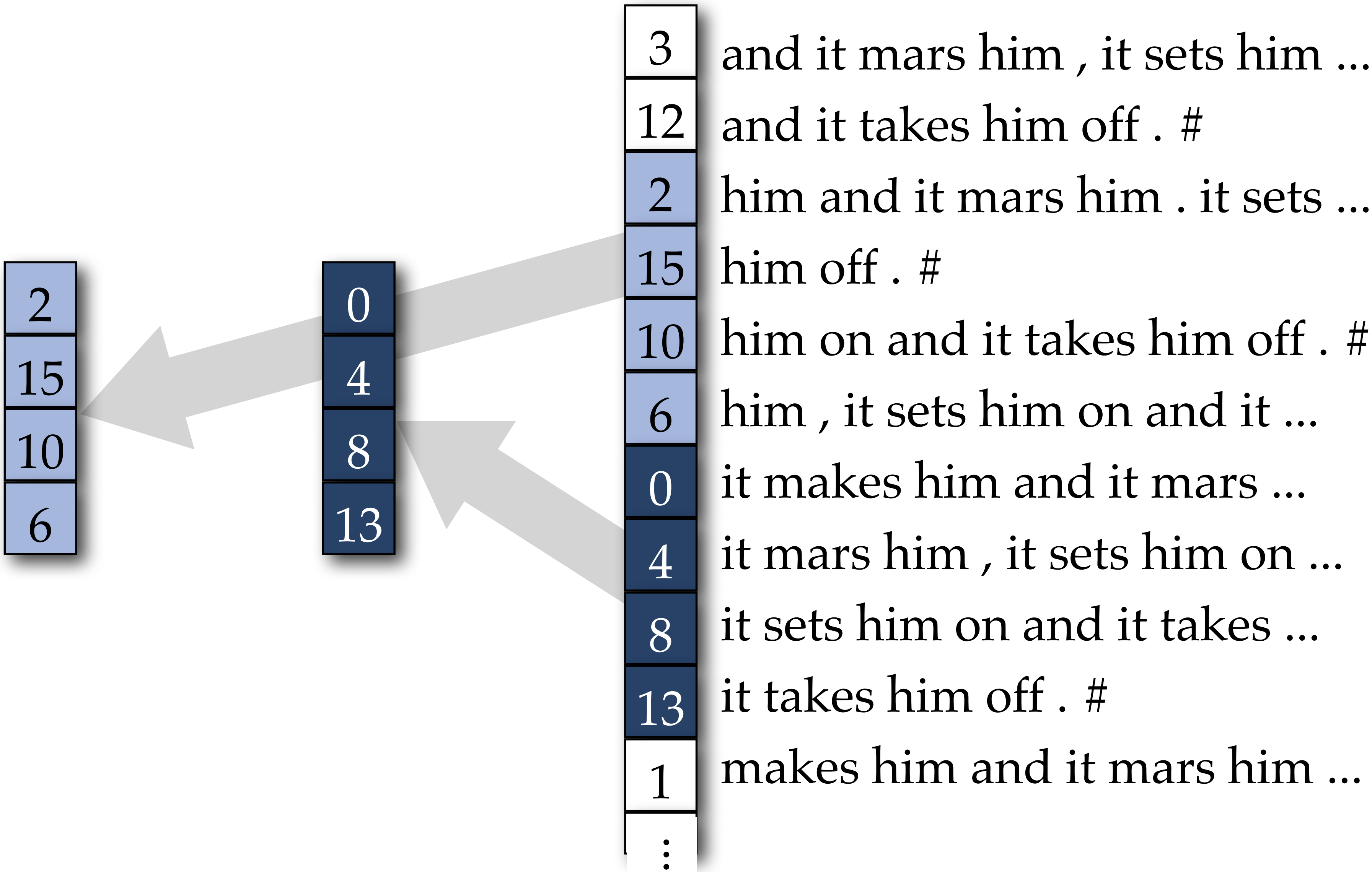
him

it

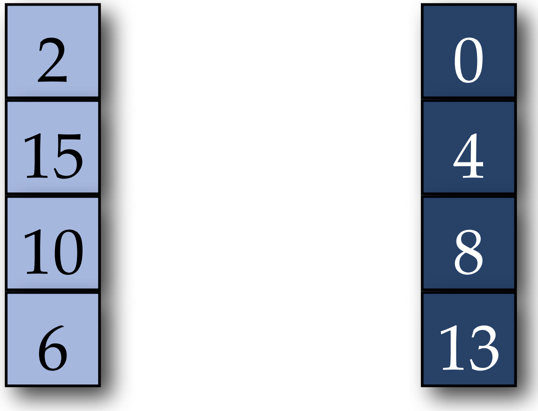
n_i Occurrences

3	and it mars him , it sets him ...
12	and it takes him off . #
2	him and it mars him . it sets ...
15	him off . #
10	him on and it takes him off . #
6	him , it sets him on and it ...
0	it makes him and it mars ...
4	it mars him , it sets him on ...
8	it sets him on and it takes ...
13	it takes him off . #
1	makes him and it mars him ...
⋮	

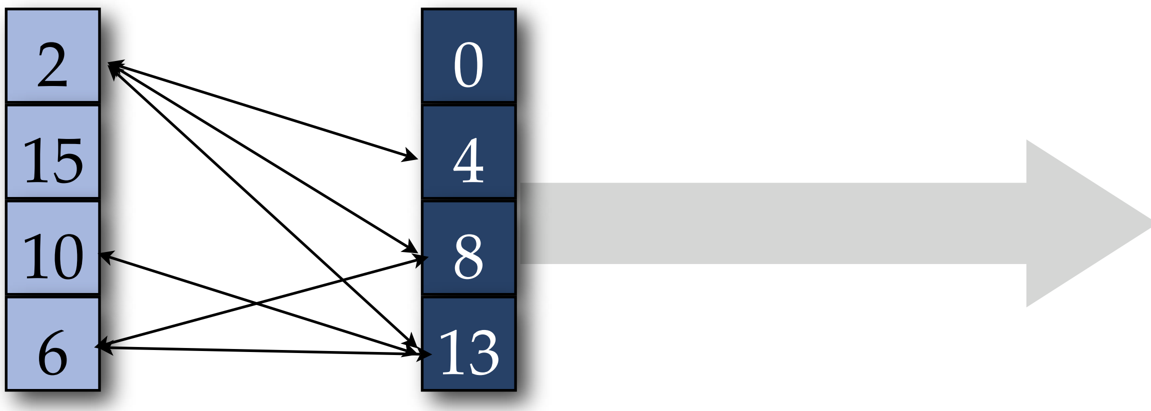
Pattern Matching with Gaps



Pattern Matching with Gaps

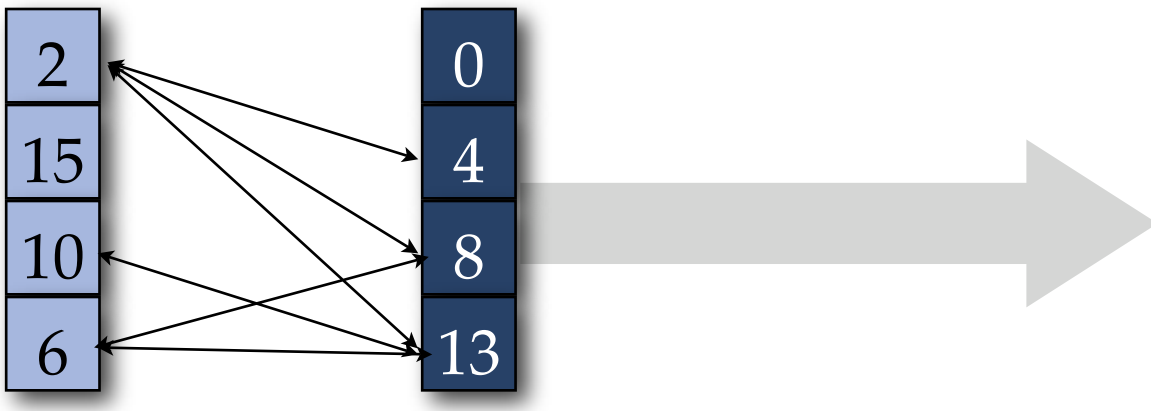


Pattern Matching with Gaps



(2, 4)
(2, 8)
(2, 13)
(6, 8)
(6, 13)
(10, 13)

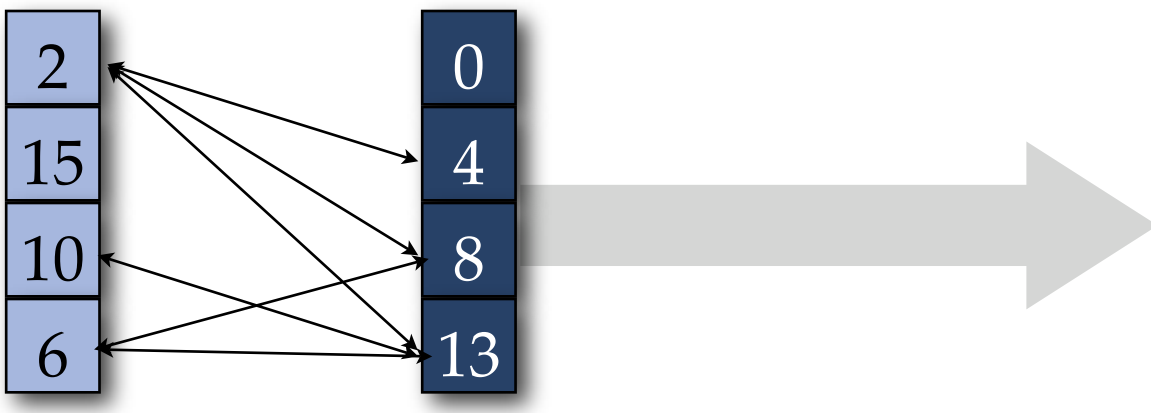
Pattern Matching with Gaps



(2, 4)
(2, 8)
(2, 13)
(6, 8)
(6, 13)
(10, 13)

RILMS (Rahman et al., 06)

Pattern Matching with Gaps



(2, 4)
(2, 8)
(2, 13)
(6, 8)
(6, 13)
(10, 13)

RILMS (Rahman et al., 06)

linear in number of occurrences of subpatterns: $O(\sum_i n_i)$

Baseline Timing Result

221

seconds
per sentence

compare: 0.009 seconds per sentence
for *contiguous* phrases

Complexity Analysis

contiguous

$$\sum_{w} (|w| + \log |T|)$$

137 5 27

discontiguous

$$\sum_{\alpha=w_1 X \ldots X w_I} \sum_{i=1}^{I} (|w_i| + \log |T| + n_i)$$

2825 3 5 27 82069

Complexity Analysis

contiguous $\sum_{w} (|w| + \log|T|)$

137 5 27

discontiguous $\boxed{\sum_{\alpha = w_1 X \ldots X w_I}} \sum_{i=1}^{I} (|w_i| + \log|T| + n_i)$

2825 3 5 27 82069

Exploiting Redundancy

Input Pattern it persuades him and it disheartens him

Query Patterns

it X and	it X disheartens him
it X it	it X and X him
it X disheartens	persuades him X disheartens
it X him	persuades him X him
persuades X it	persuades X it disheartens
persuades X disheartens	persuades X disheartens him
persuades X him	him and X him
it persuades X it	him X disheartens him
it persuades X disheartens	it persuades him X disheartens
it persuades X him	it persuades him X him
it X and it	it persuades X it disheartens
it X it disheartens	it persuades X disheartens him

Exploiting Redundancy

Input Pattern it persuades him and it disheartens him

Query Patterns

it X and	it X disheartens him
it X it	it X and X him
it X disheartens	persuades him X disheartens
it X him	persuades him X him
persuades X it	persuades X it disheartens
persuades X disheartens	persuades X disheartens him
persuades X him	him and X him
it persuades X it	him X disheartens him
it persuades X disheartens	it persuades him X disheartens
it persuades X him	it persuades him X him
it X and it	it persuades X it disheartens
it X it disheartens	it persuades X disheartens him

Exploiting Redundancy

Query Pattern it persuades X disheartens him

Exploiting Redundancy

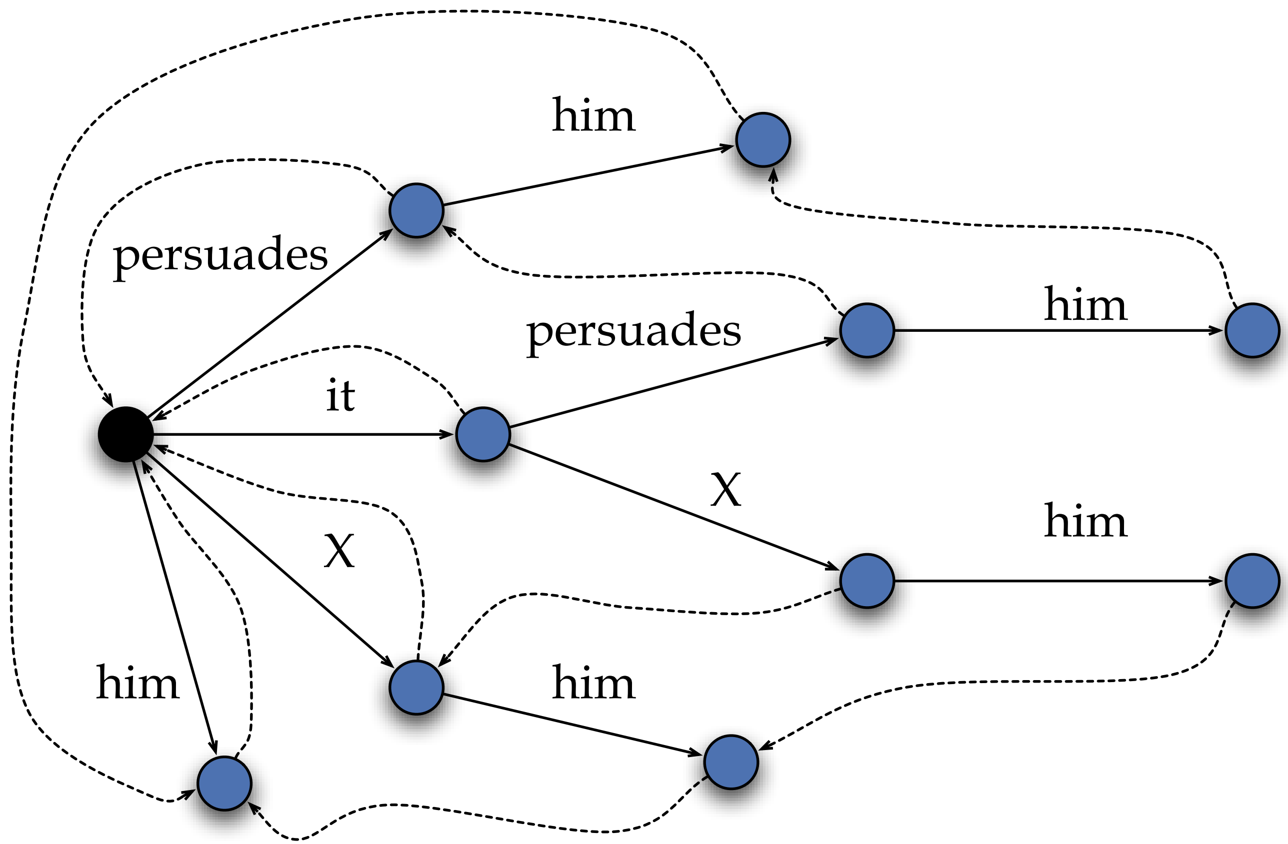
Query Pattern	it persuades X disheartens him
Maximal Prefix	it persuades X disheartens

(Zhang & Vogel 2005)

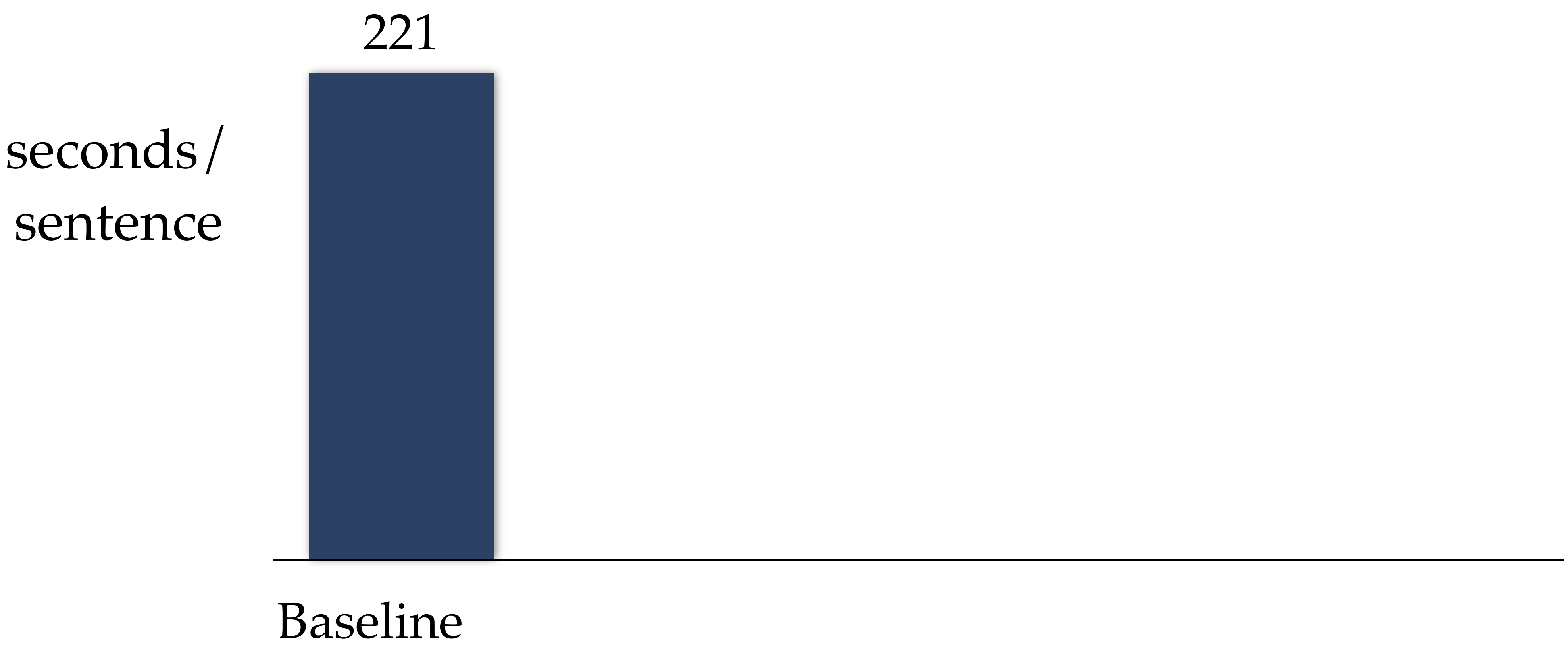
Exploiting Redundancy

Query Pattern	it persuades X disheartens him
Maximal Prefix	it persuades X disheartens
Maximal Suffix	persuades X disheartens him

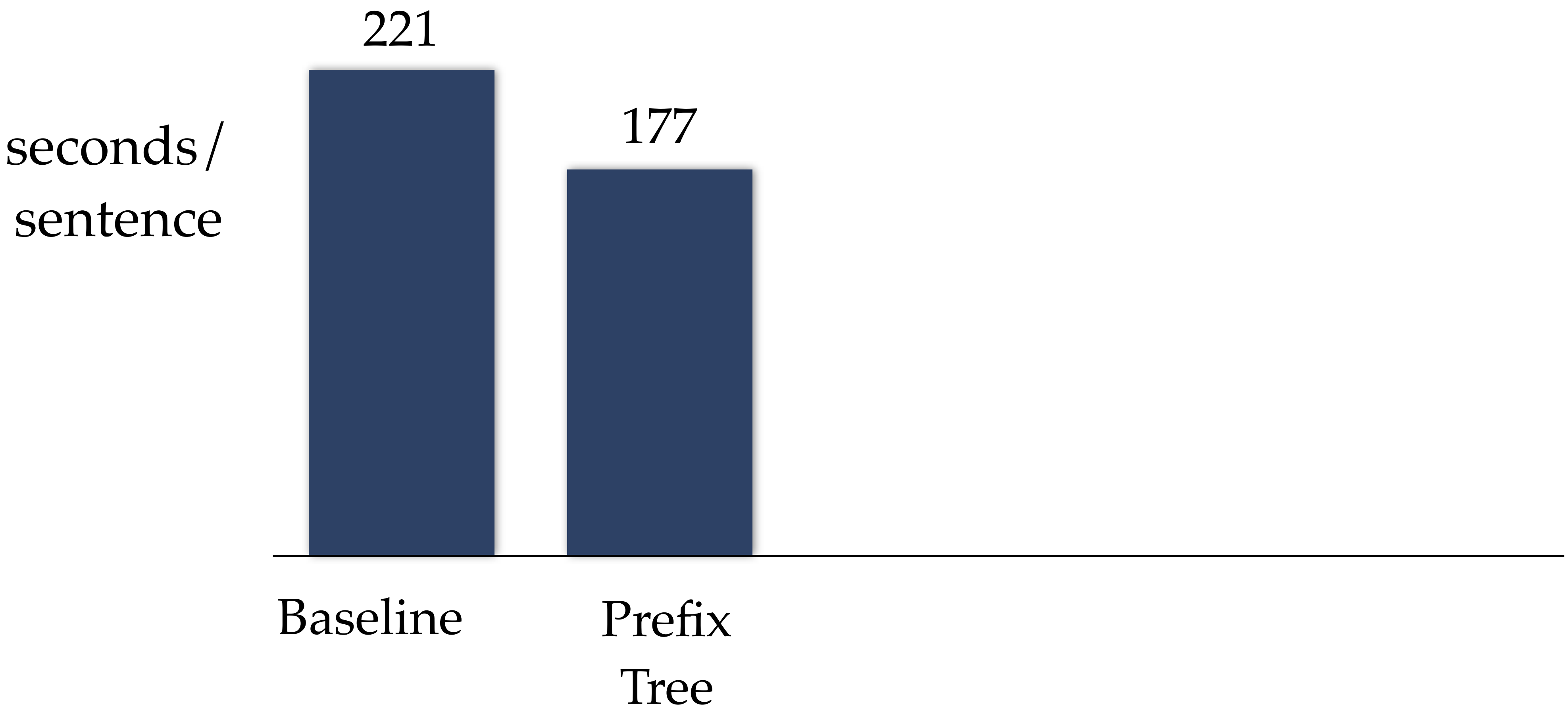
Prefix Tree with Suffix Links



Timing Results



Timing Results



Complexity Analysis

contiguous

$$\sum_{w} (|w| + \log |T|)$$

137 5 27

discontiguous

$$\sum_{\alpha = w_1 X \ldots X w_I} \sum_{i=1}^{I} (|w_i| + \log |T| + n_i)$$

2825 3 5 27 82069

Complexity Analysis

contiguous

$$\sum_{w} (|w| + \log |T|)$$

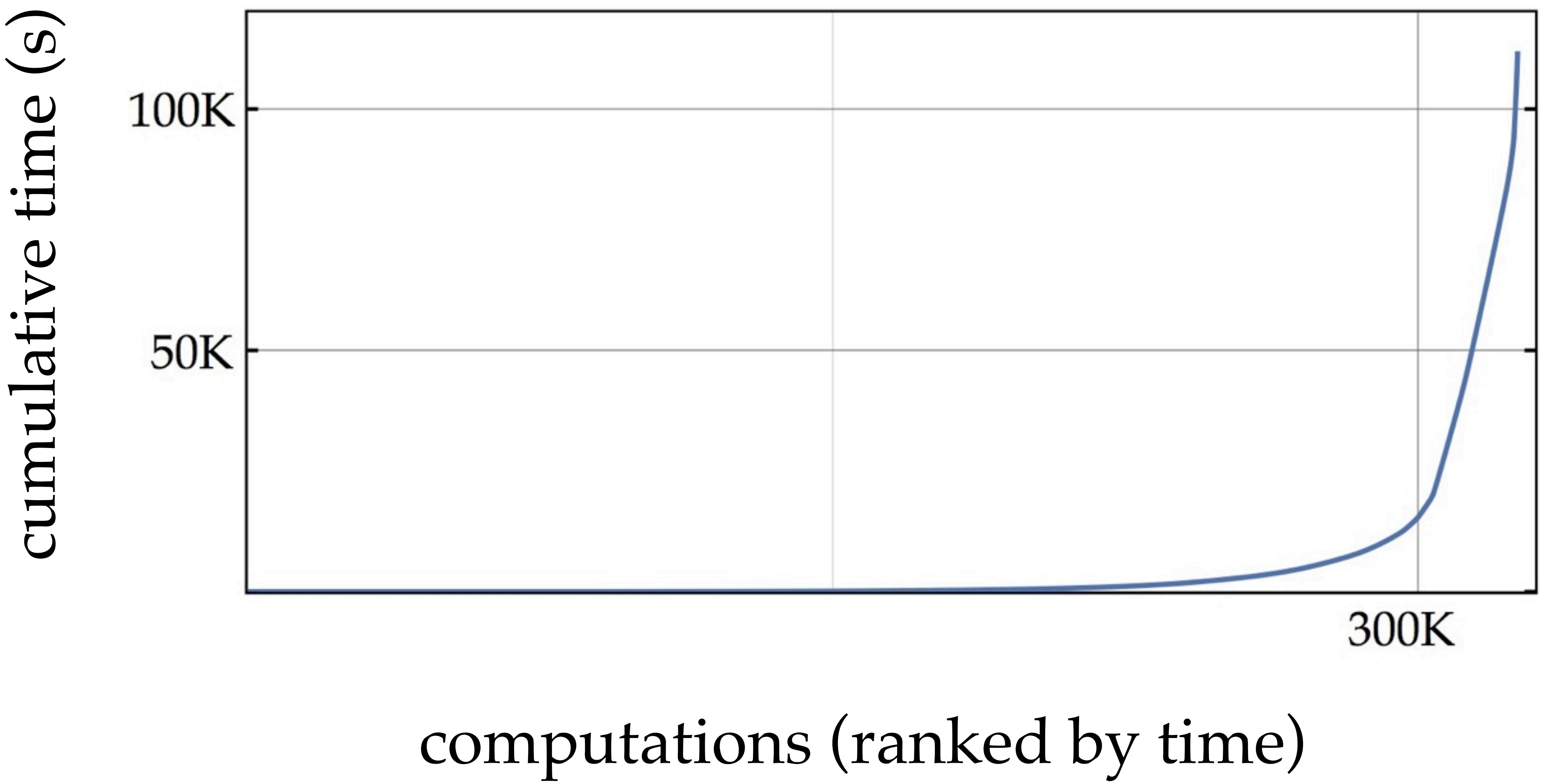
137 5 27

discontiguous

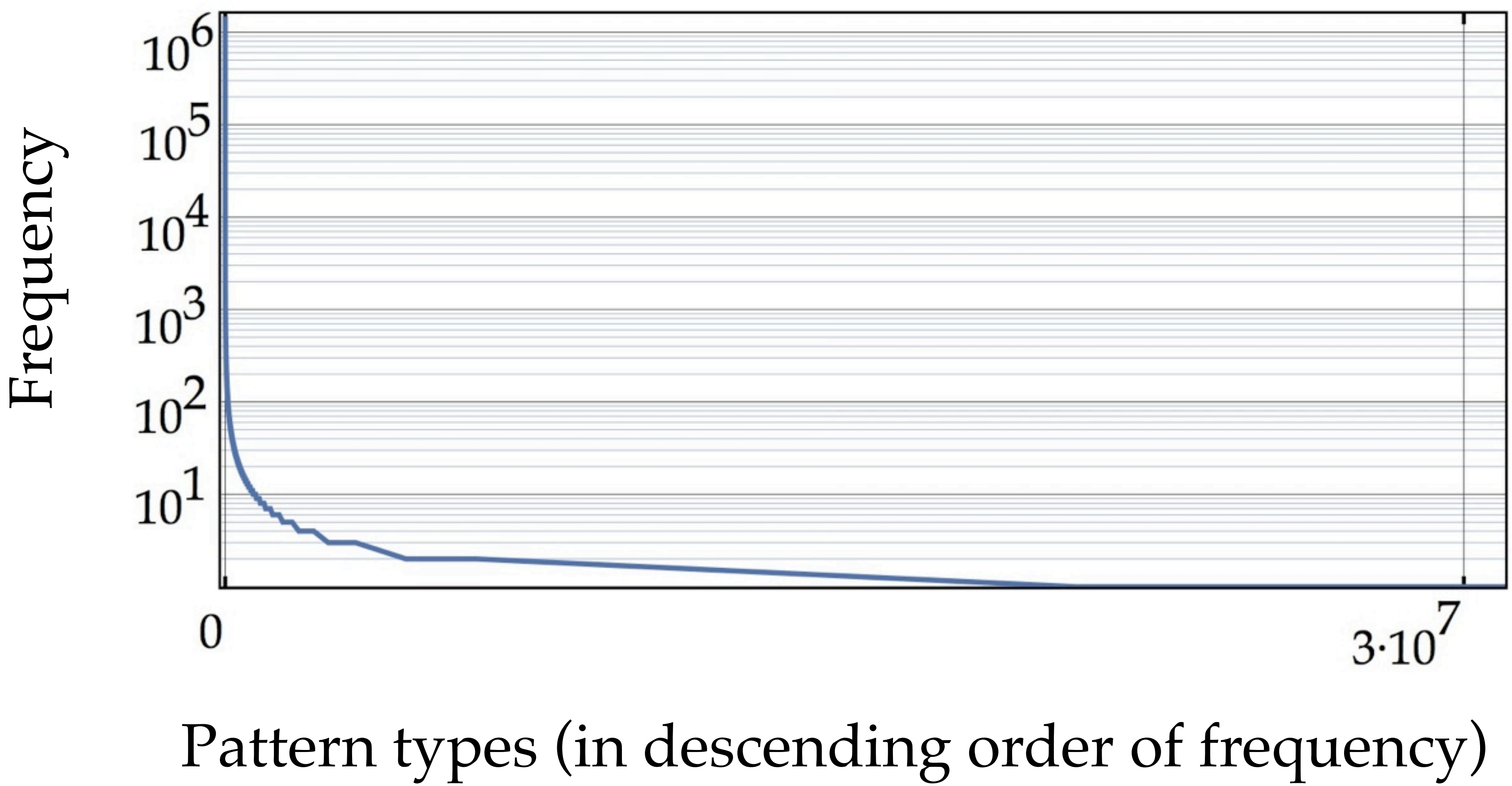
$$\sum_{\alpha = w_1 X \ldots X w_I} \sum_{i=1}^{I} (|w_i| + \log |T| + n_i)$$

2825 3 5 27 82069

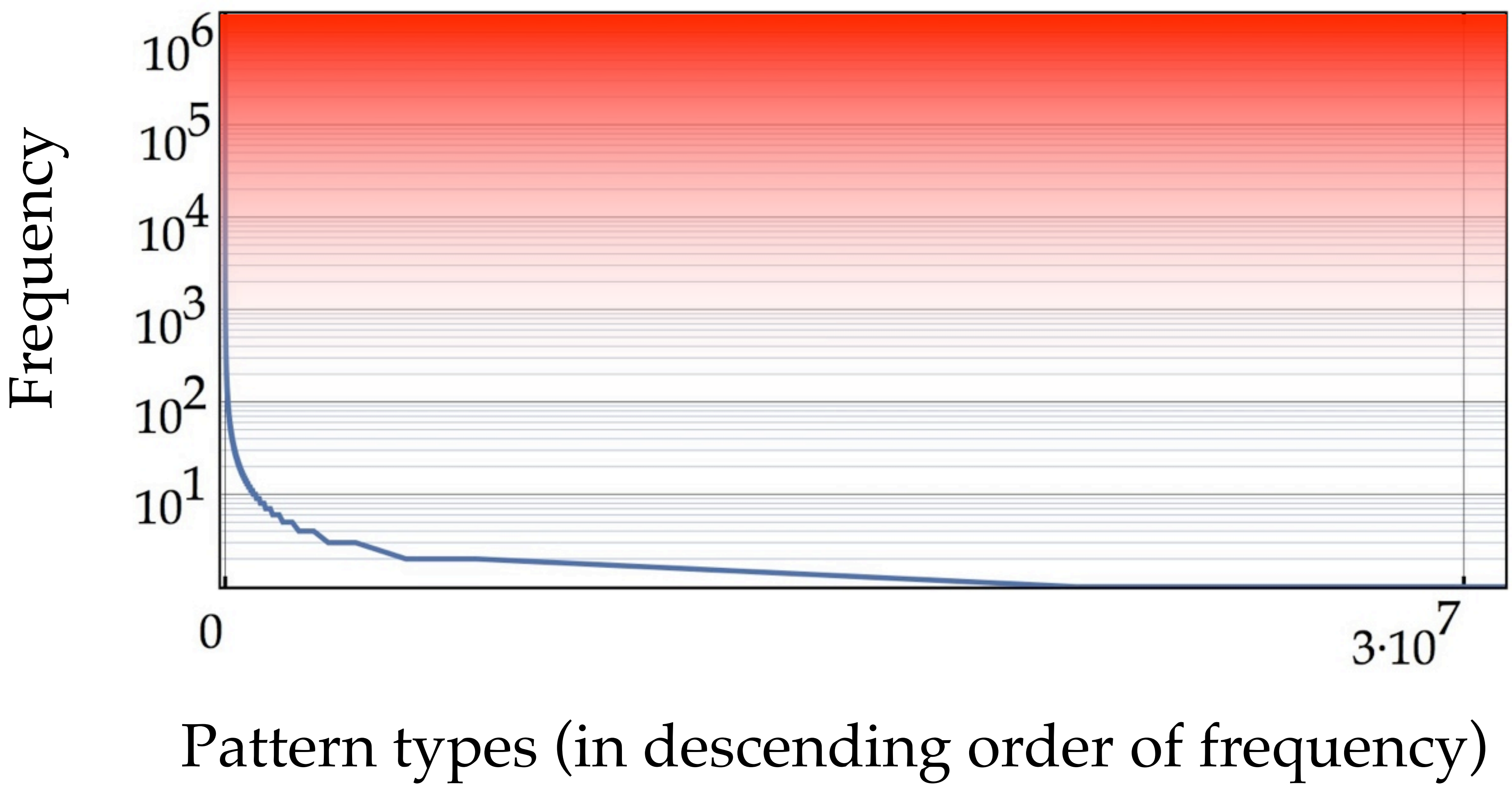
Empirical Analysis



Distribution of Patterns in Training Data



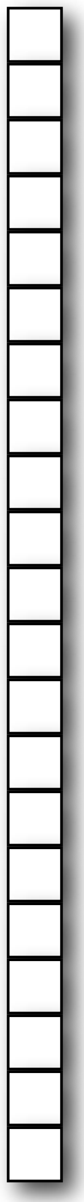
Distribution of Patterns in Training Data



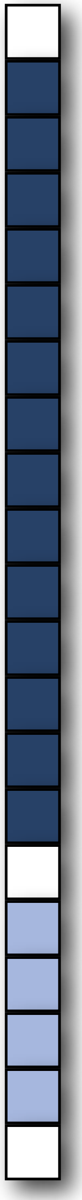
Analysis of Problem

- The expensive computations involve at least one frequent subpattern. There are two cases.
  - A frequent pattern paired with an infrequent pattern
  - Two frequent patterns paired with each other

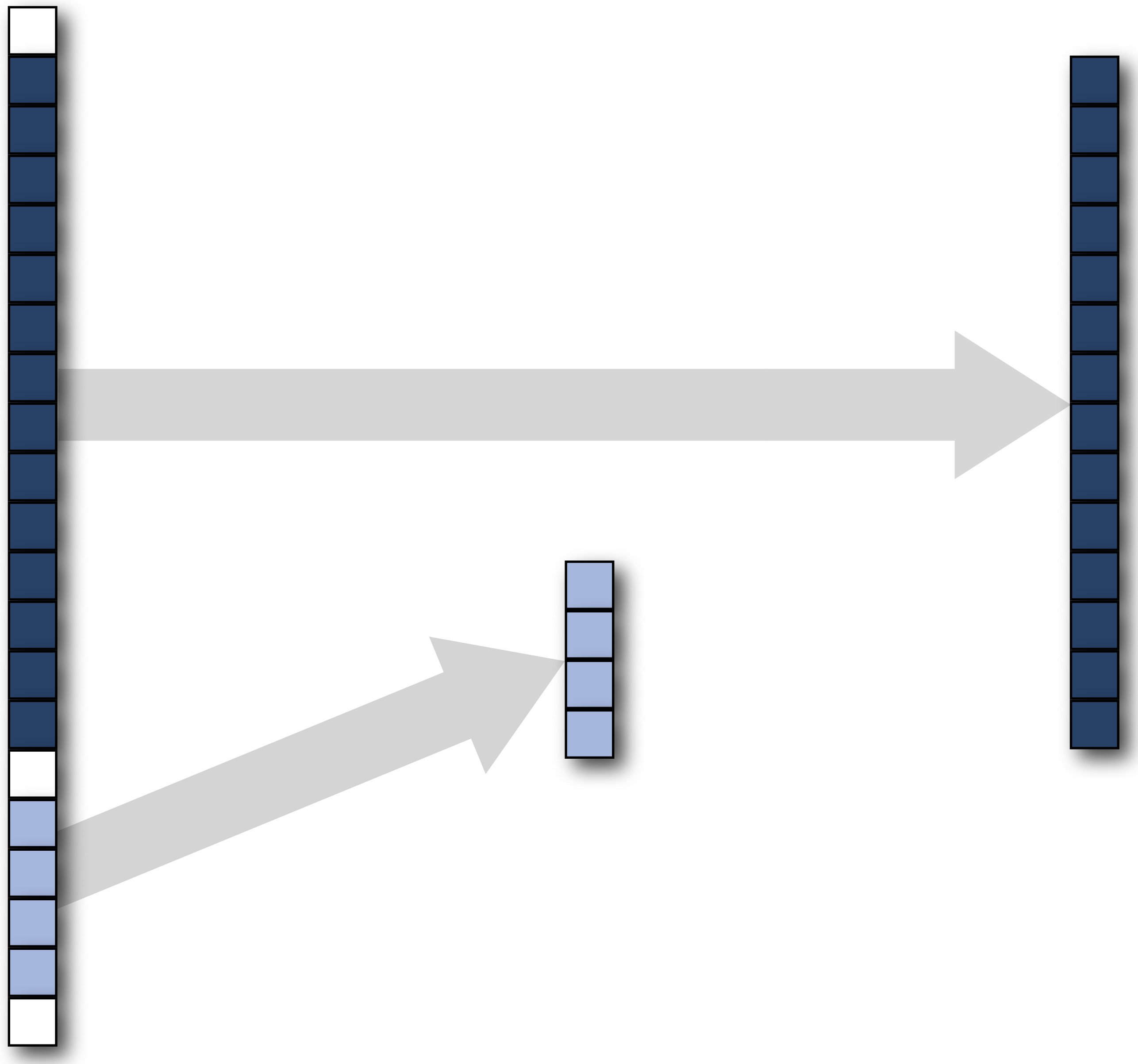
Frequent × Infrequent Subpatterns



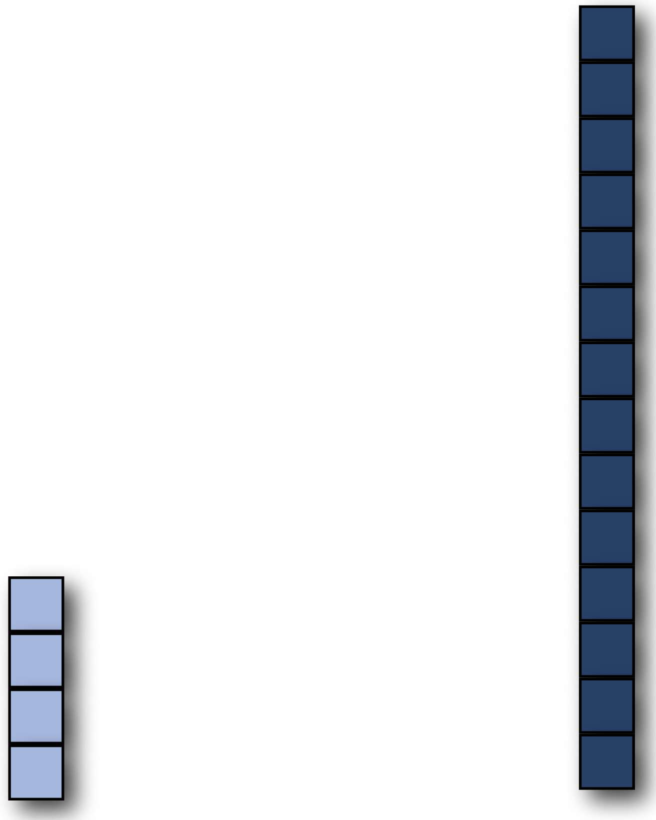
Frequent × Infrequent Subpatterns



Frequent × Infrequent Subpatterns

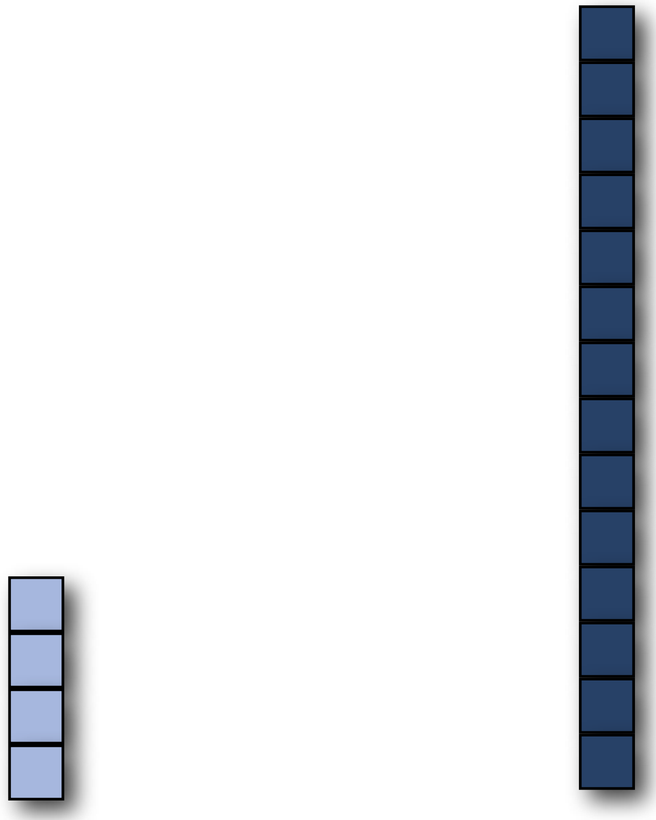


Frequent × Infrequent Subpatterns



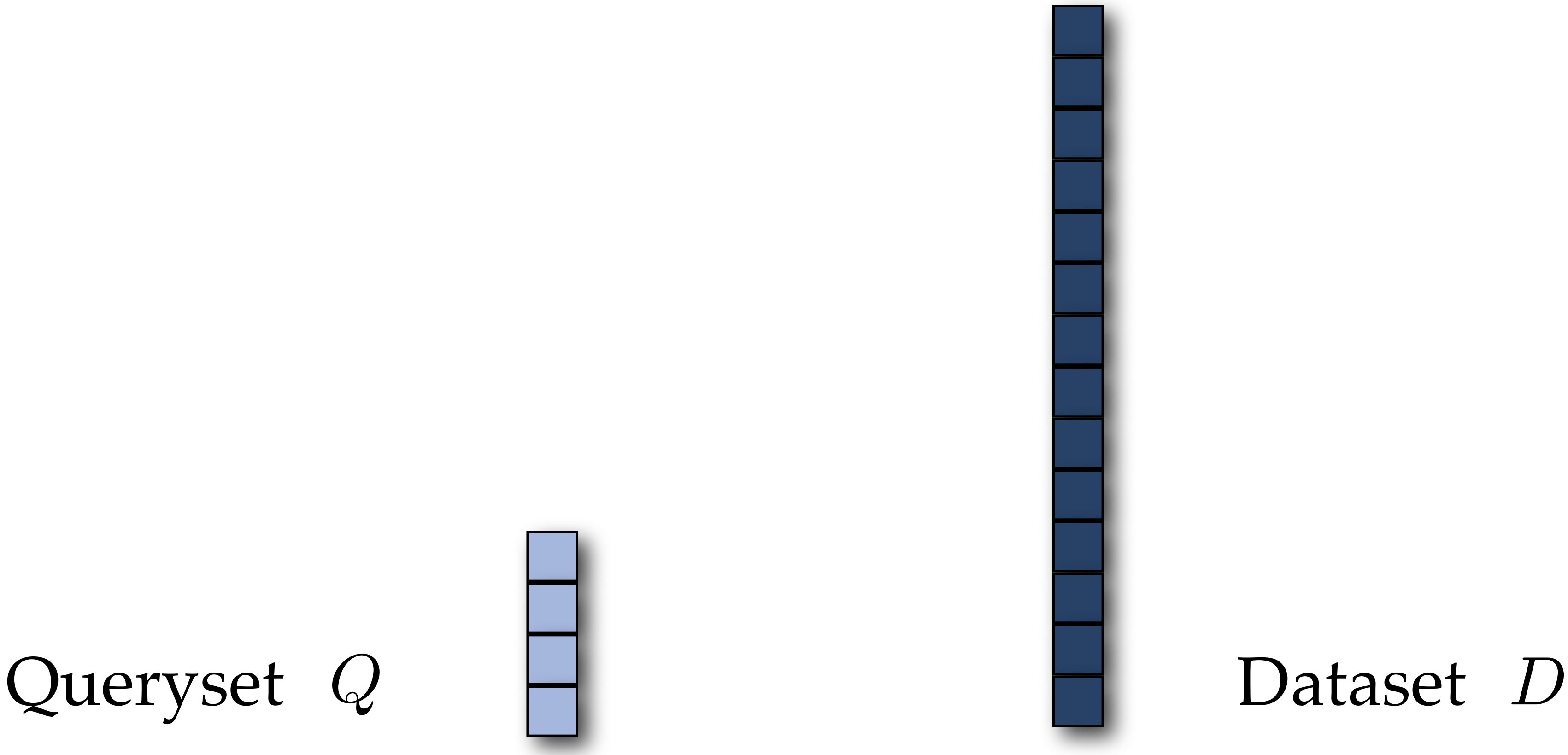
Double Binary Search

Baeza-Yates, 04



Double Binary Search

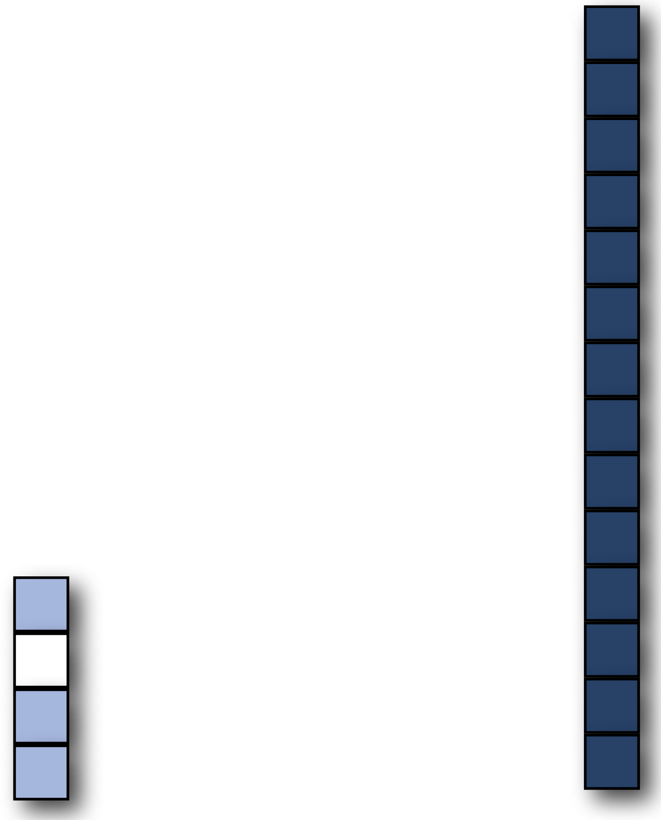
Baeza-Yates, 04



Double Binary Search

Baeza-Yates, 04

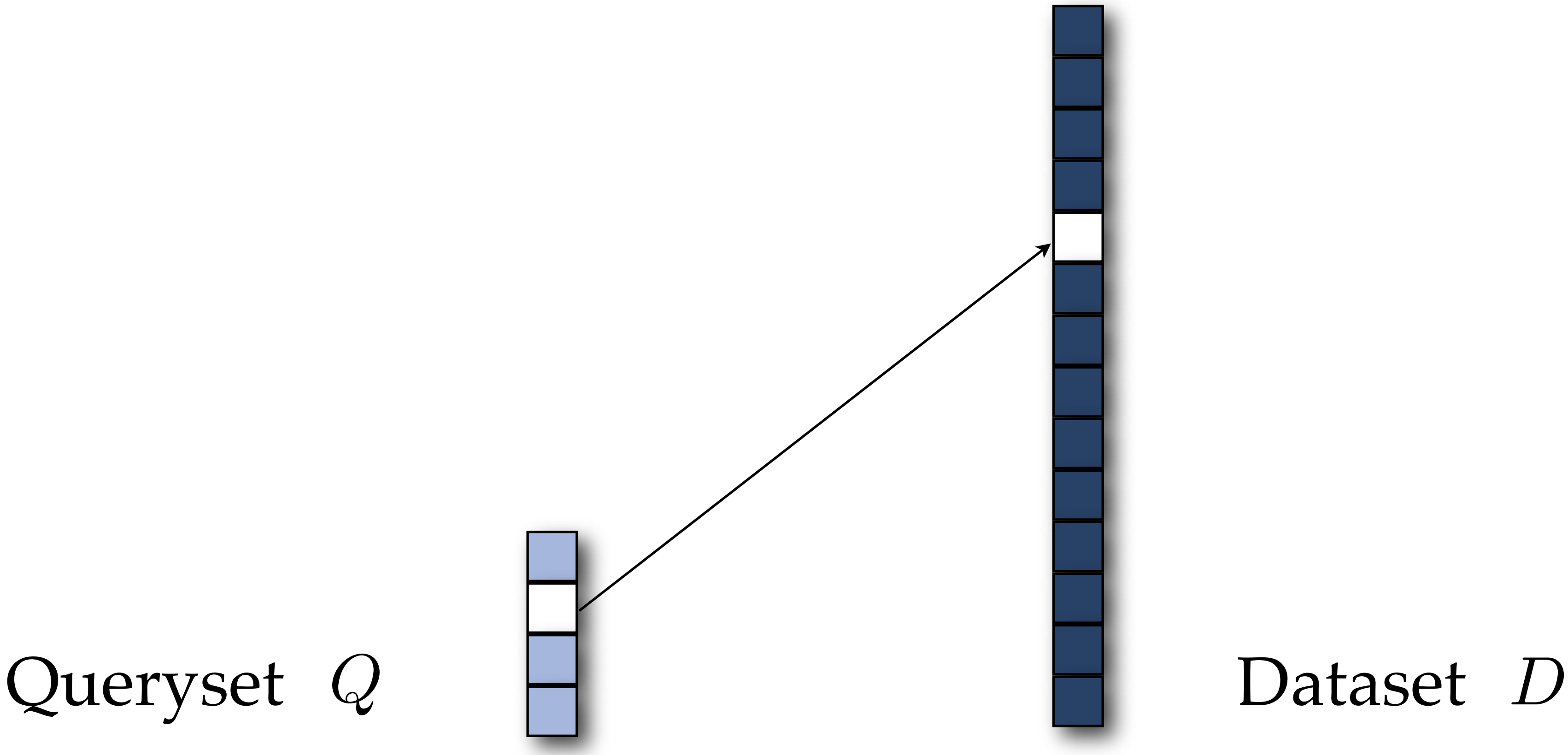
Queryset Q



Dataset D

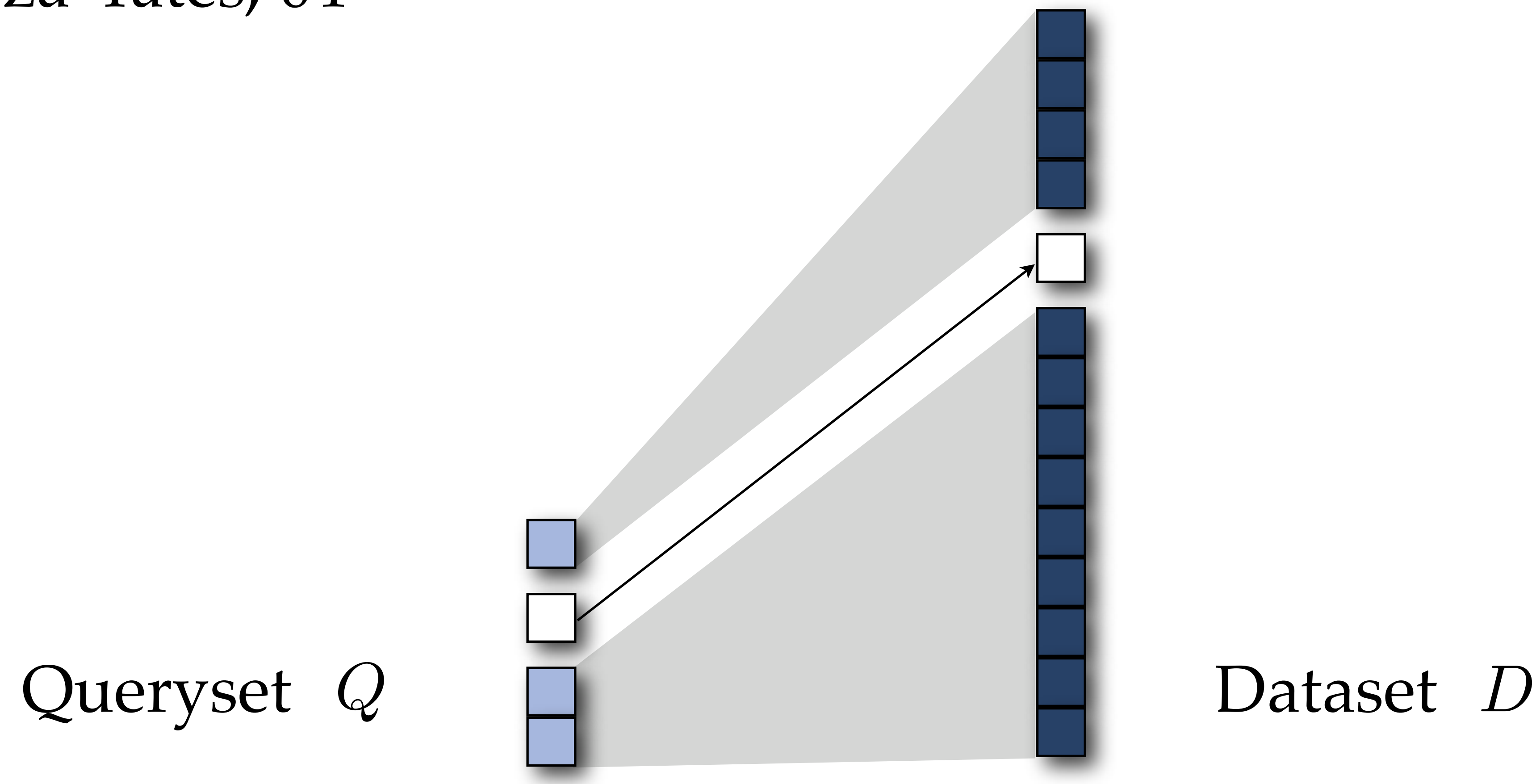
Double Binary Search

Baeza-Yates, 04



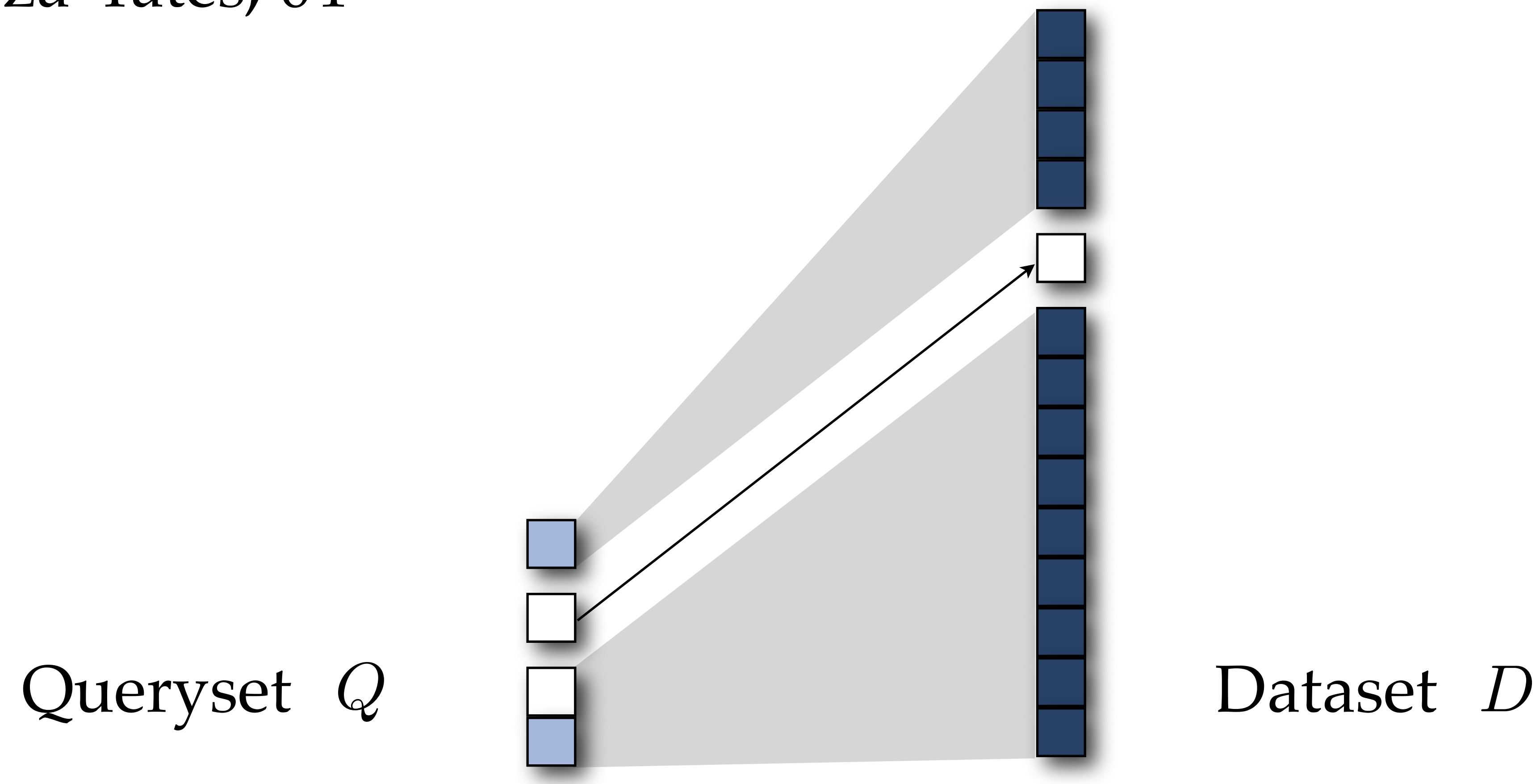
Double Binary Search

Baeza-Yates, 04



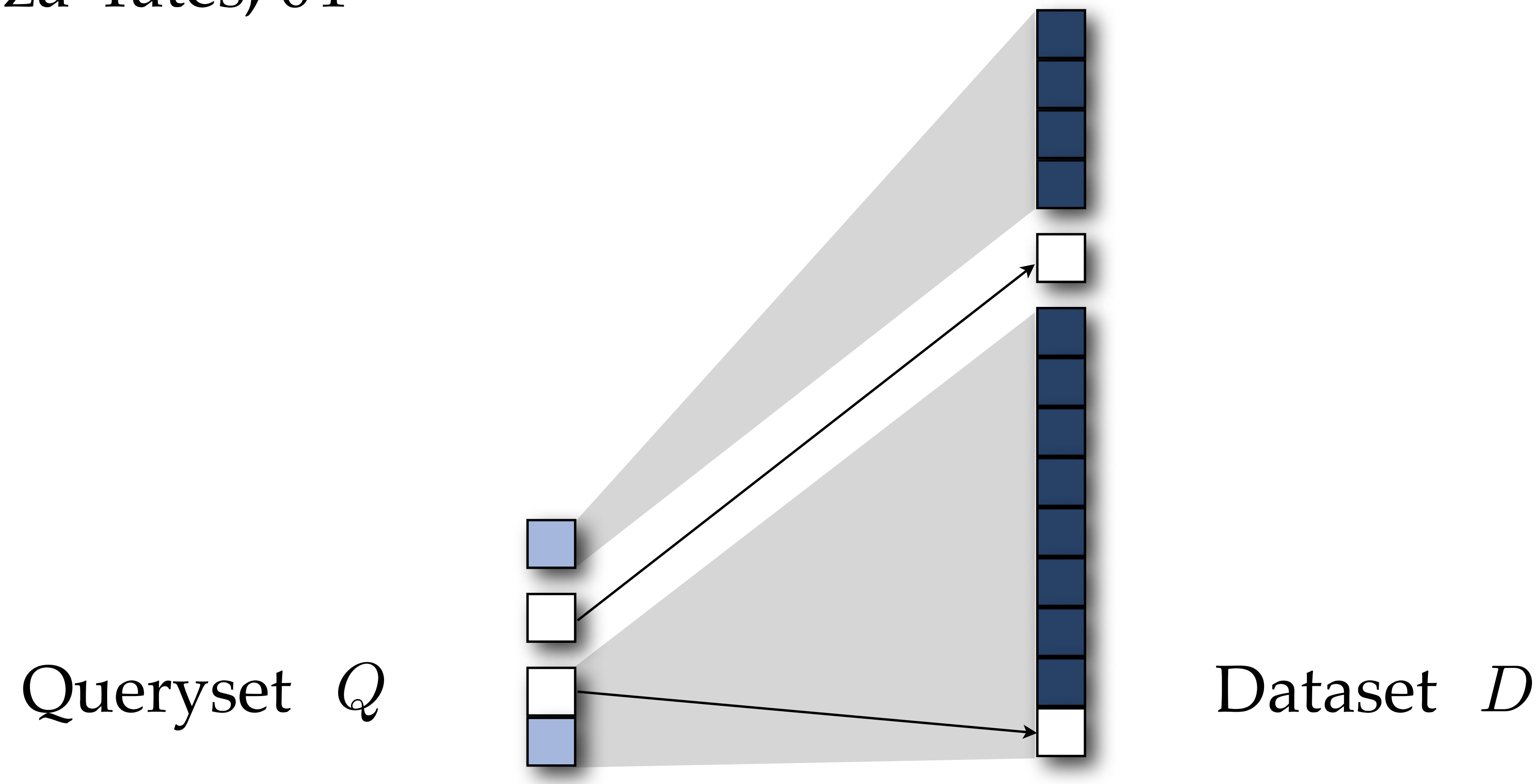
Double Binary Search

Baeza-Yates, 04



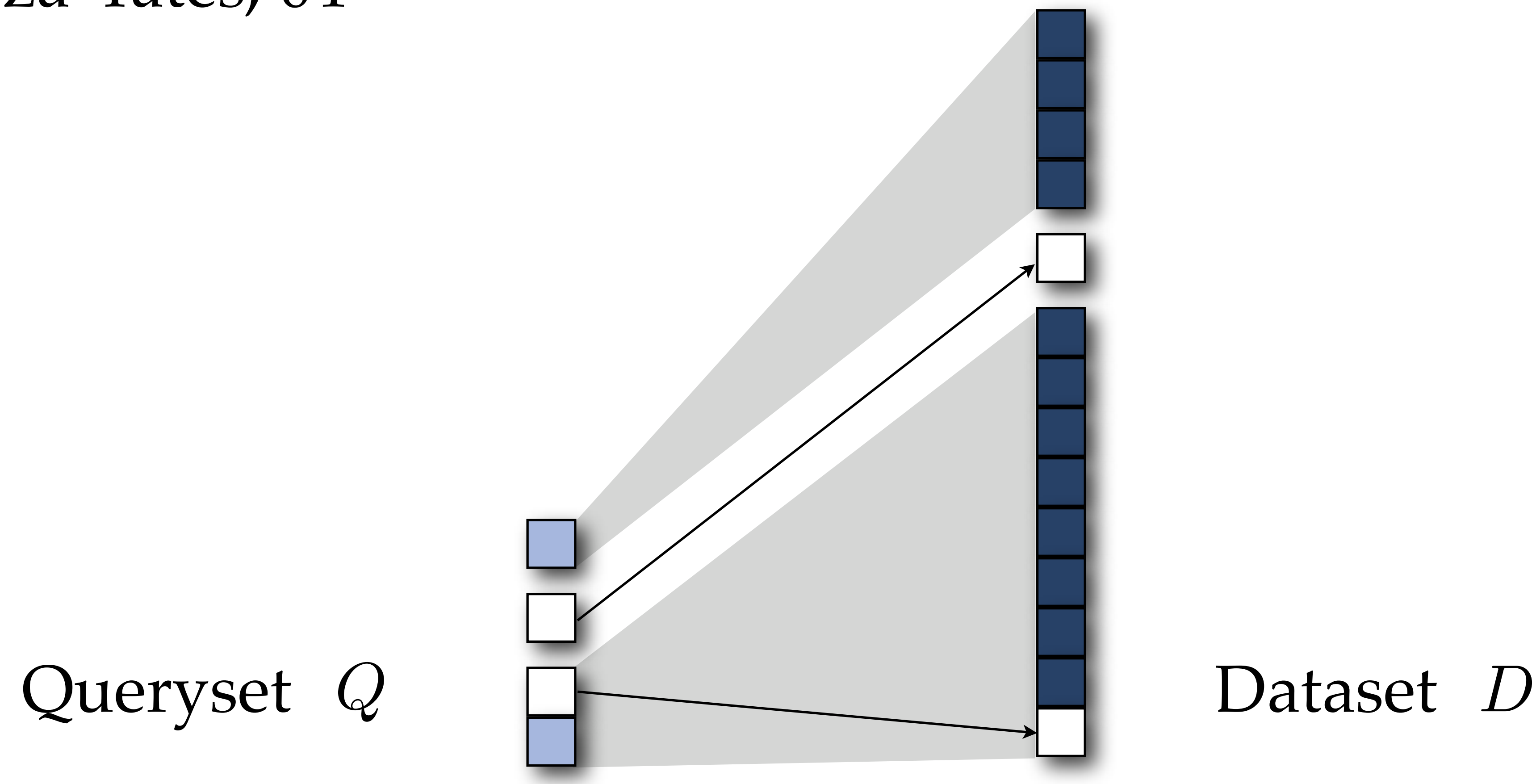
Double Binary Search

Baeza-Yates, 04



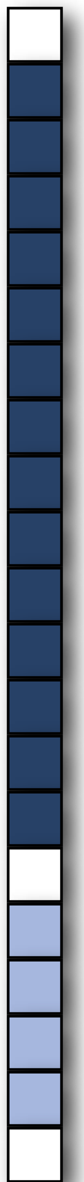
Double Binary Search

Baeza-Yates, 04



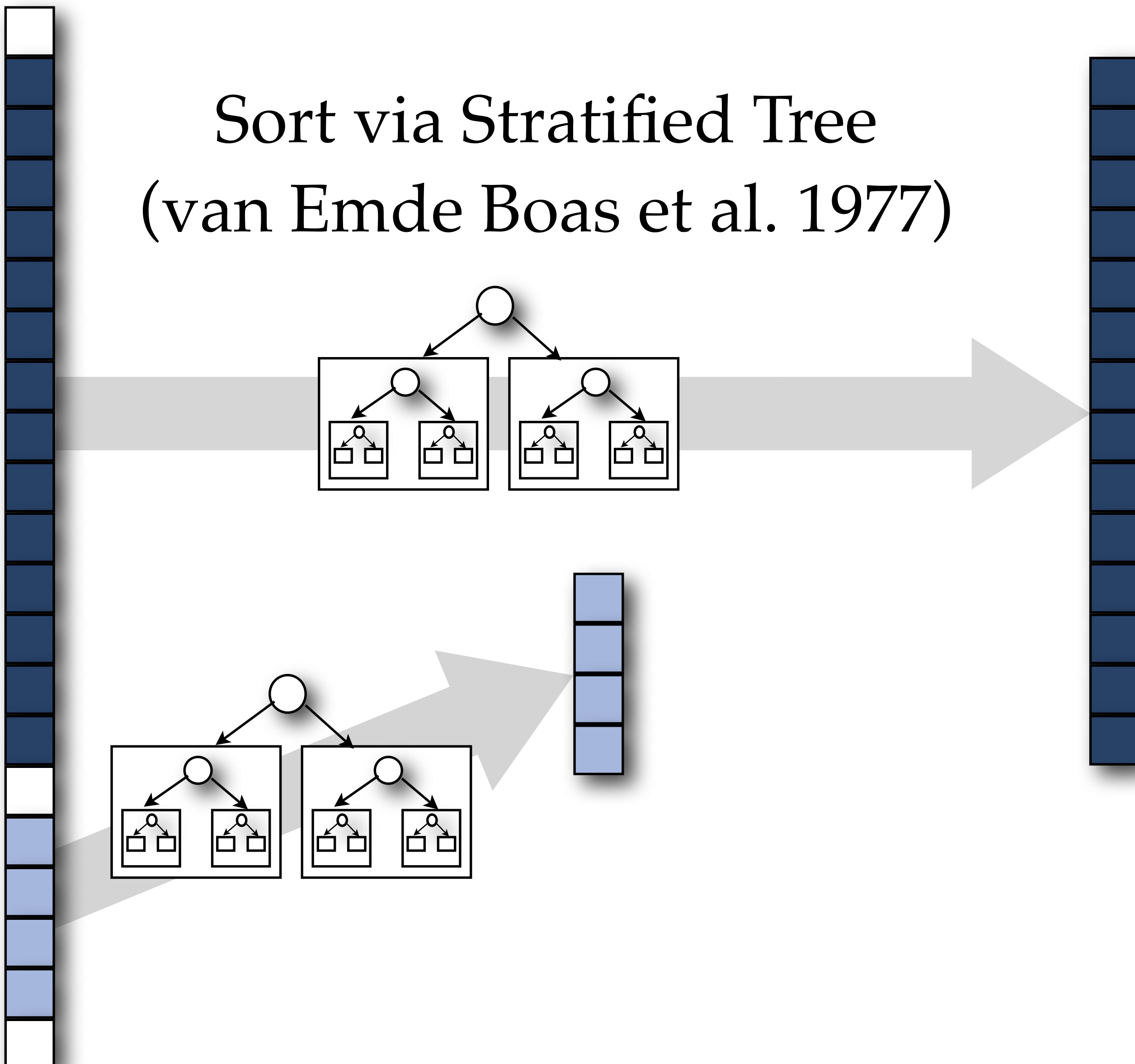
Upper bound complexity: $|Q| \log |D|$

Obtaining Sorted Sets



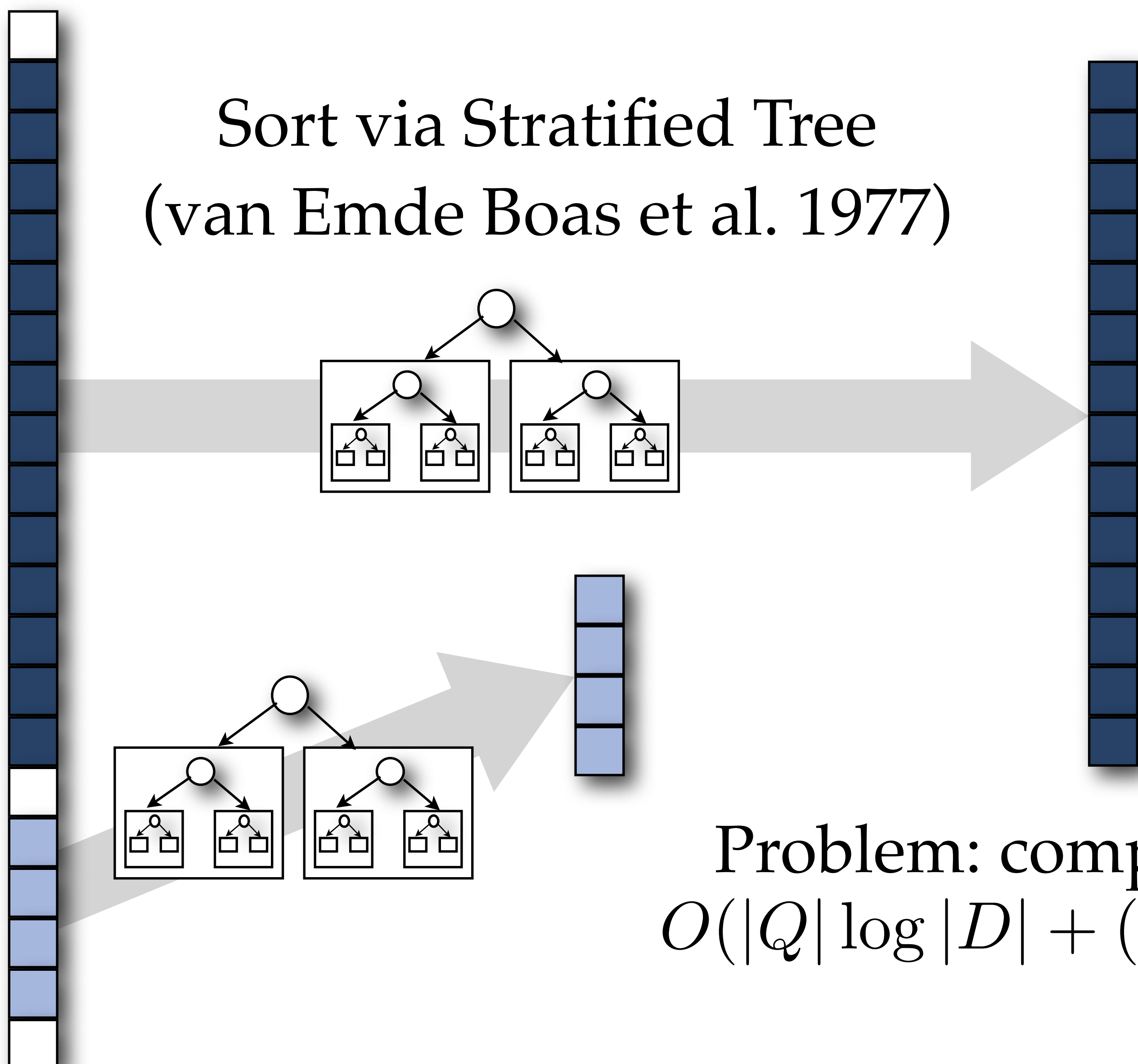
Obtaining Sorted Sets

Sort via Stratified Tree
(van Emde Boas et al. 1977)



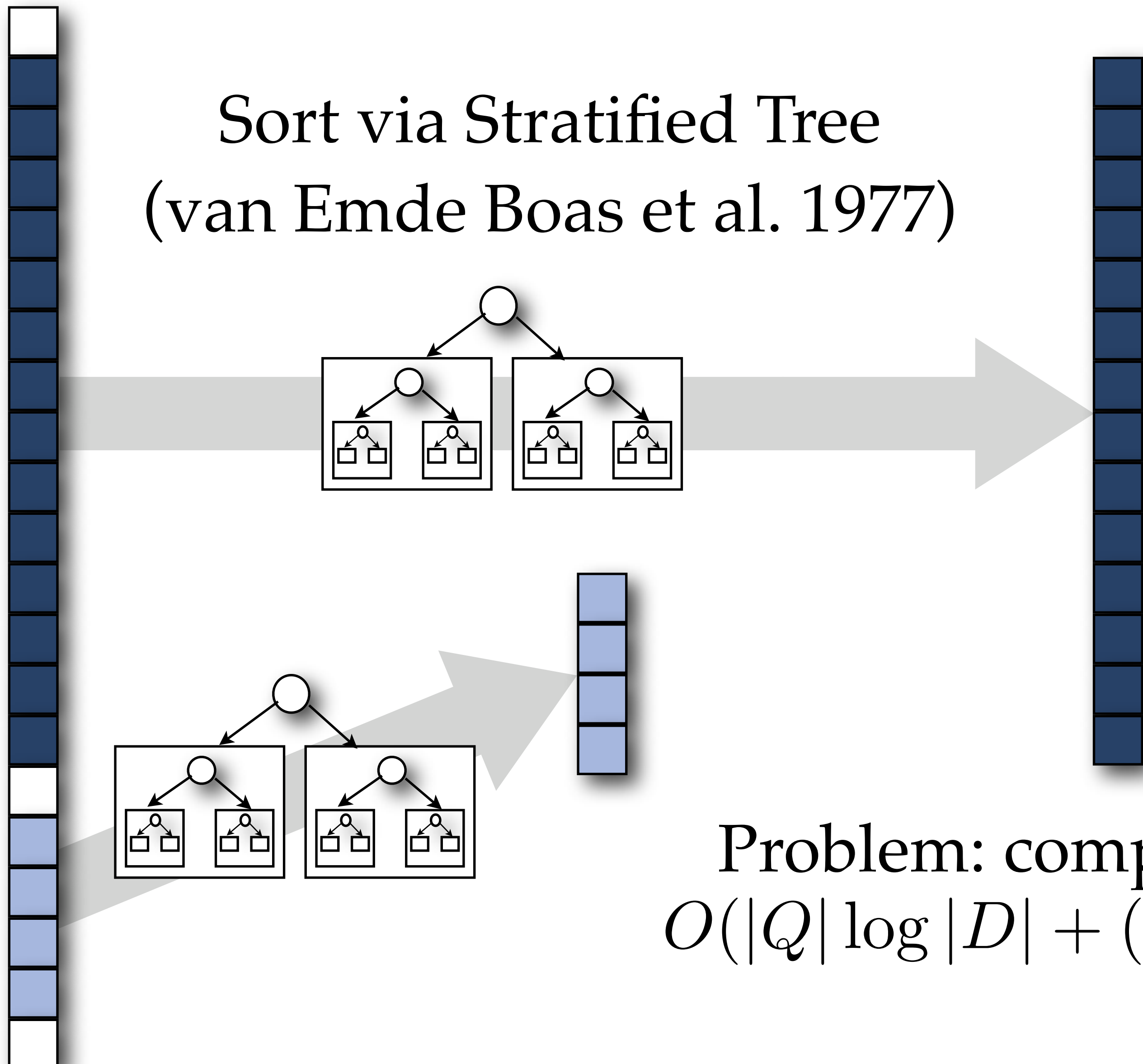
Obtaining Sorted Sets

Sort via Stratified Tree
(van Emde Boas et al. 1977)



Problem: complexity increases to
$O(|Q| \log |D| + (|Q| + |D|) \log \log |T|)$

Obtaining Sorted Sets

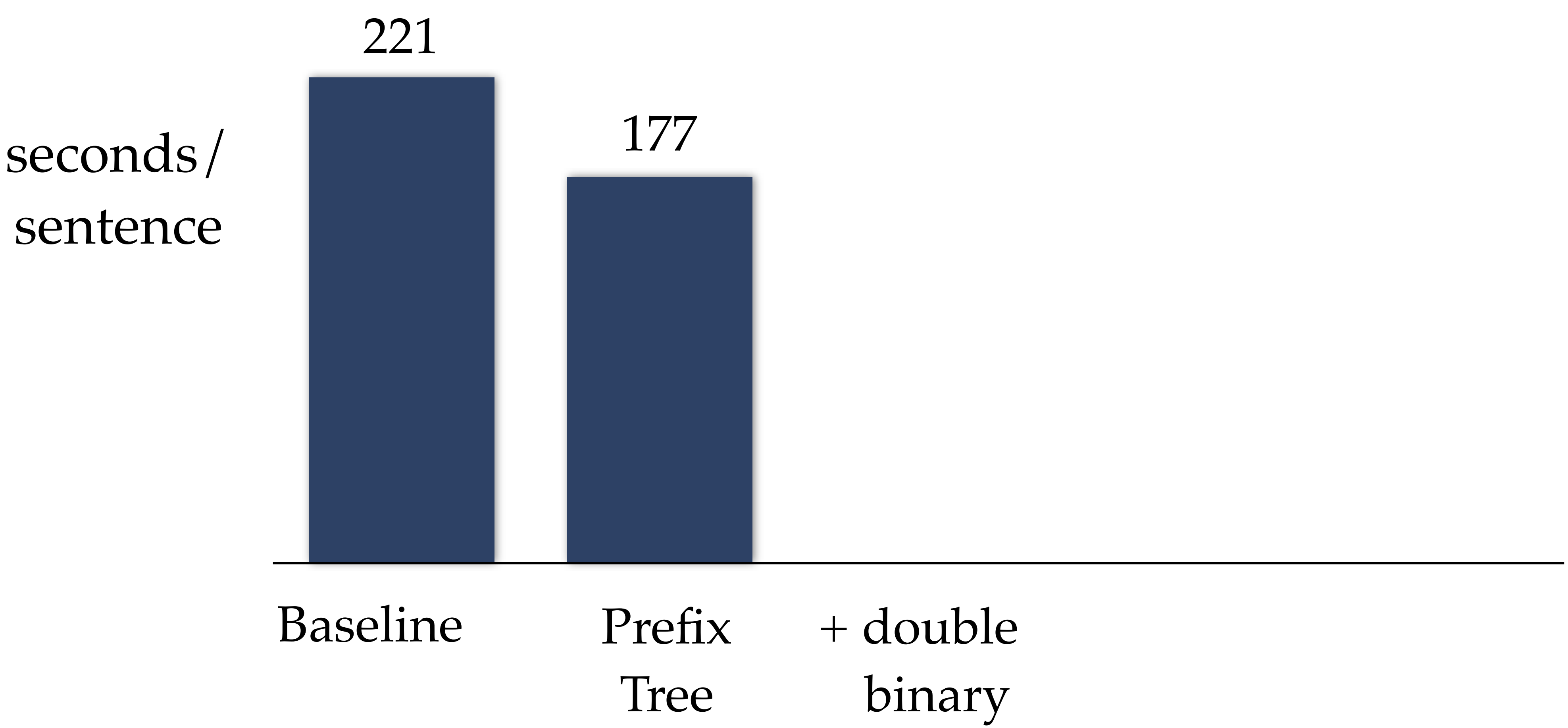


Sort via Stratified Tree
(van Emde Boas et al. 1977)

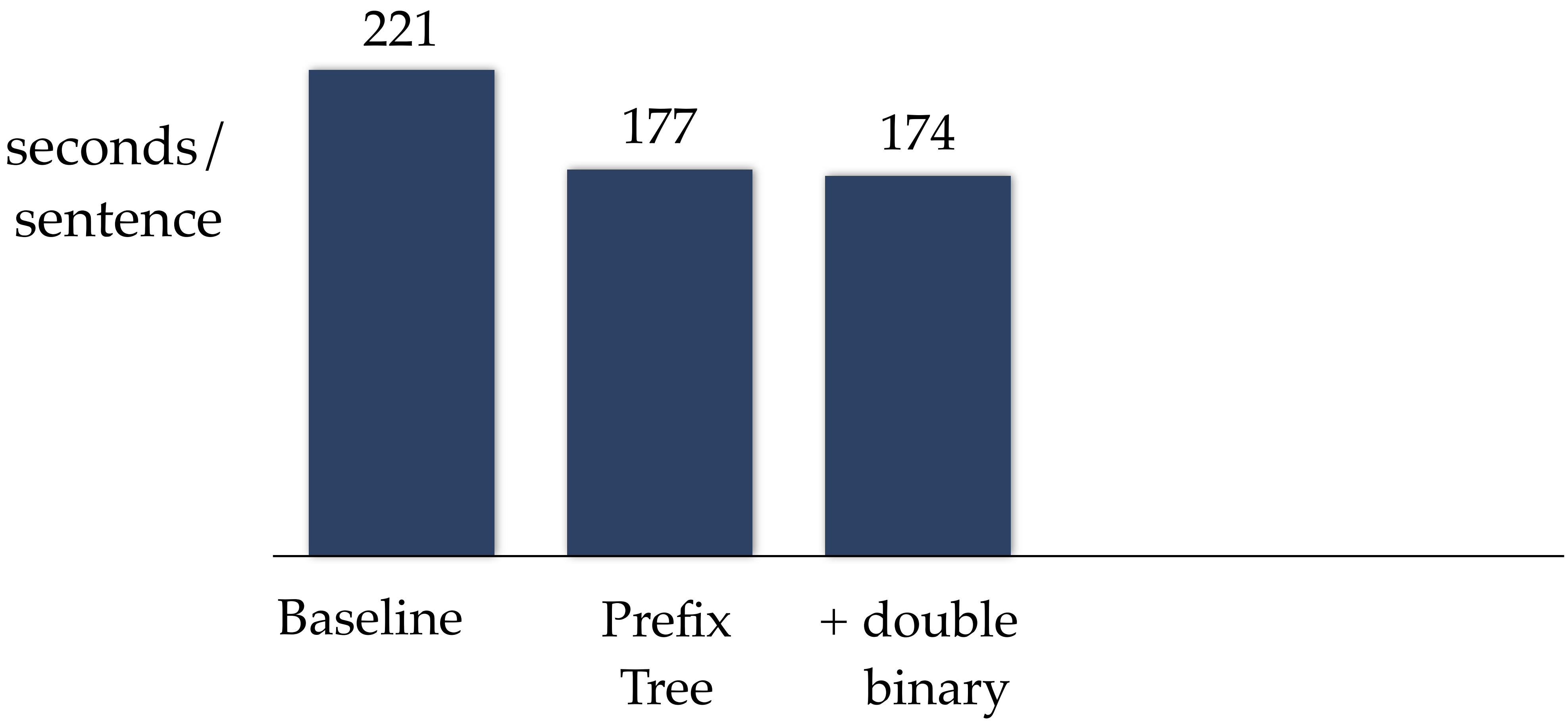
Solution: cache sorted set in prefix tree

Problem: complexity increases to
$O(|Q| \log |D| + (|Q| + |D|) \log \log |T|)$

Timing Results

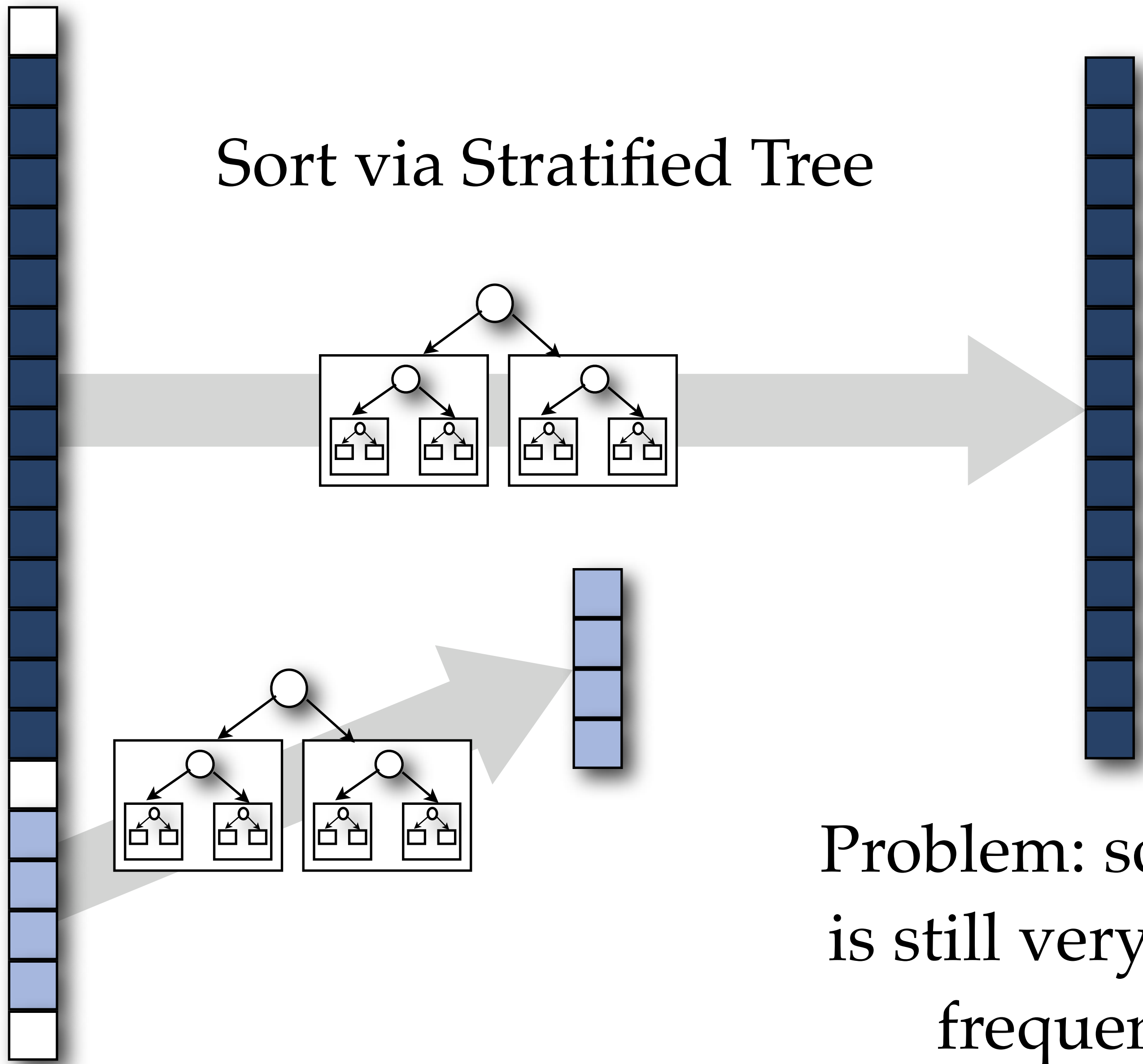


Timing Results



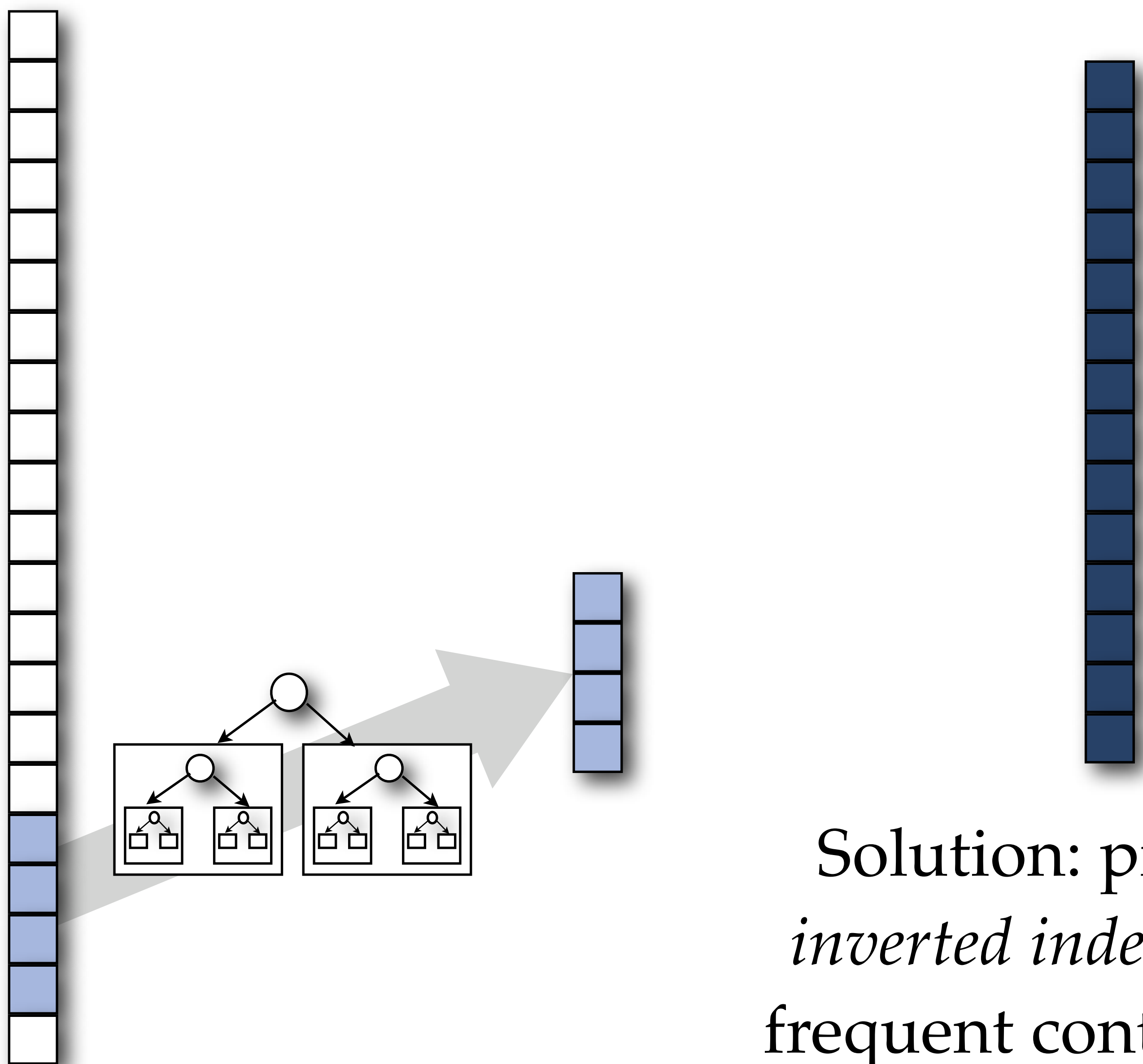
Obtaining Sorted Sets

Sort via Stratified Tree



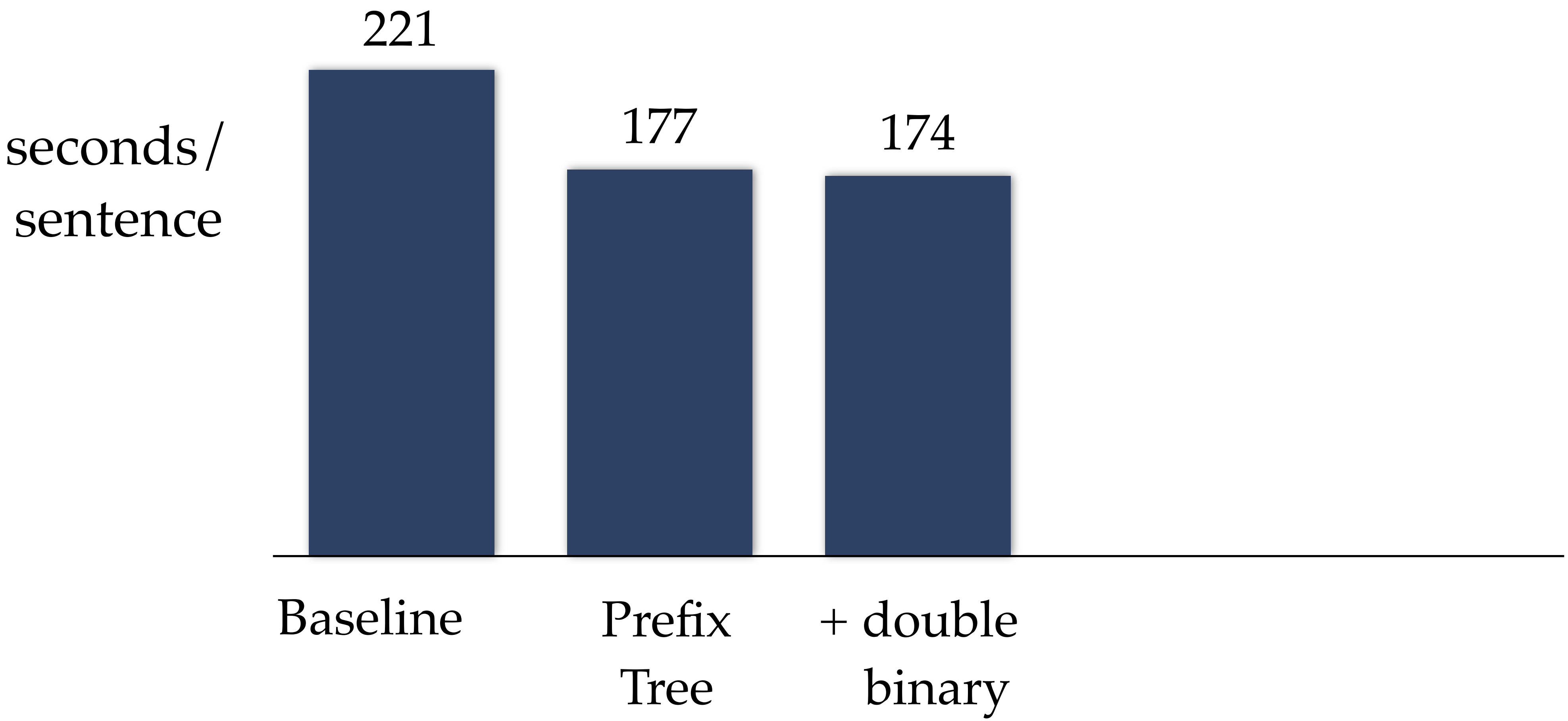
Problem: sort complexity is still very high for very frequent patterns

Obtaining Sorted Sets

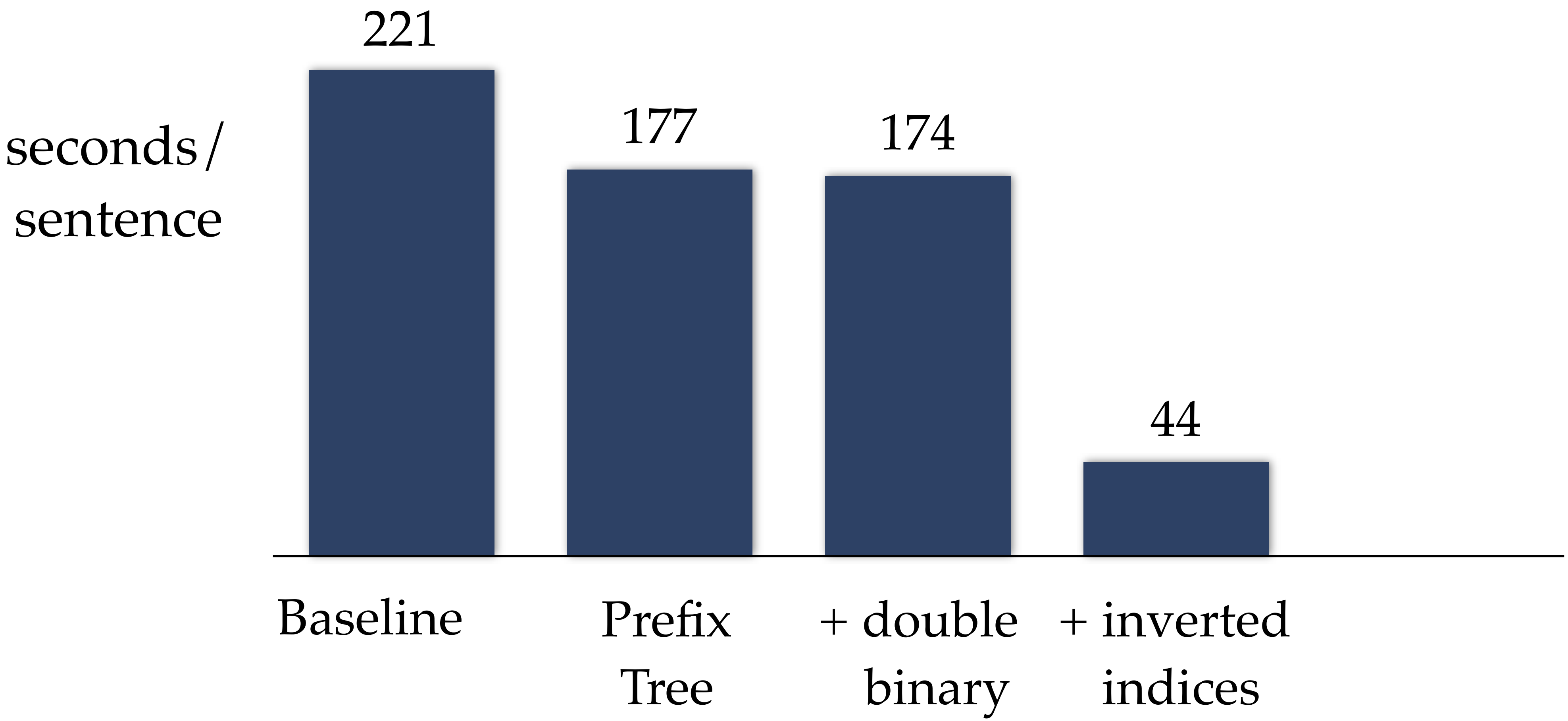


Solution: precompute the *inverted index* for 1000 most frequent contiguous patterns

Timing Results



Timing Results



Frequent × Frequent Subpatterns

Frequent × Frequent Subpatterns

Problem:
There is no clever algorithm to
solve this problem

Solution: Precomputation

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text

Solution: Precomputation

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text

Most Frequent Patterns

it (4)
him (4)

Precomputed Pattern Matches

it X him

him X it

it X it

him X him

Solution: Precomputation

it makes him and it mars him . it sets him on and it takes him off . #

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

Text

Most Frequent Patterns

it (4)

him (4)

Precomputed Pattern Matches

it X him

(0, 2)(0, 6)(13, 15)
(4, 6)(4, 10)(8, 10)(8, 15)

him X it

(2, 4)(2, 8)(10, 13)
(6, 8)(6, 13)

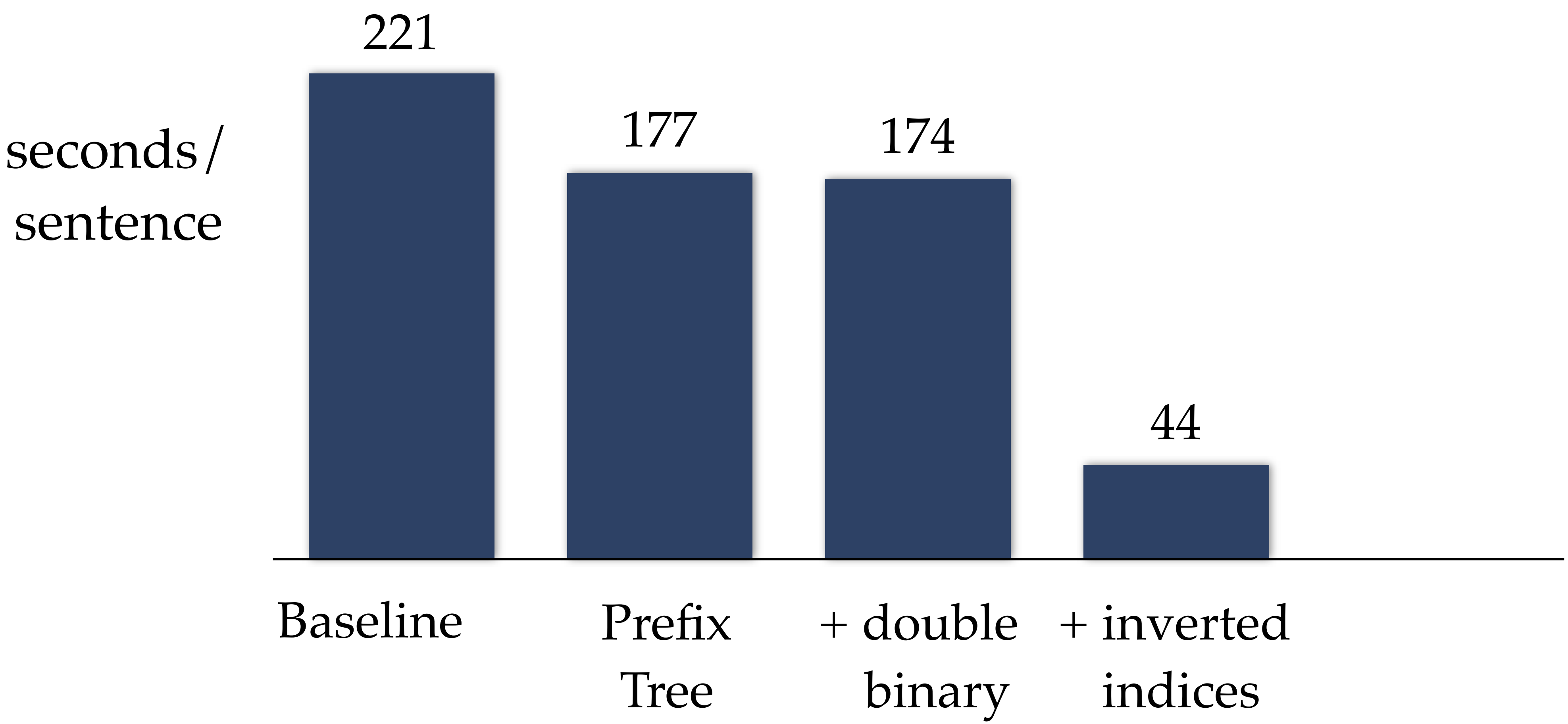
it X it

(0, 4)(0, 8)
(4, 8)(4, 13)(8, 13)

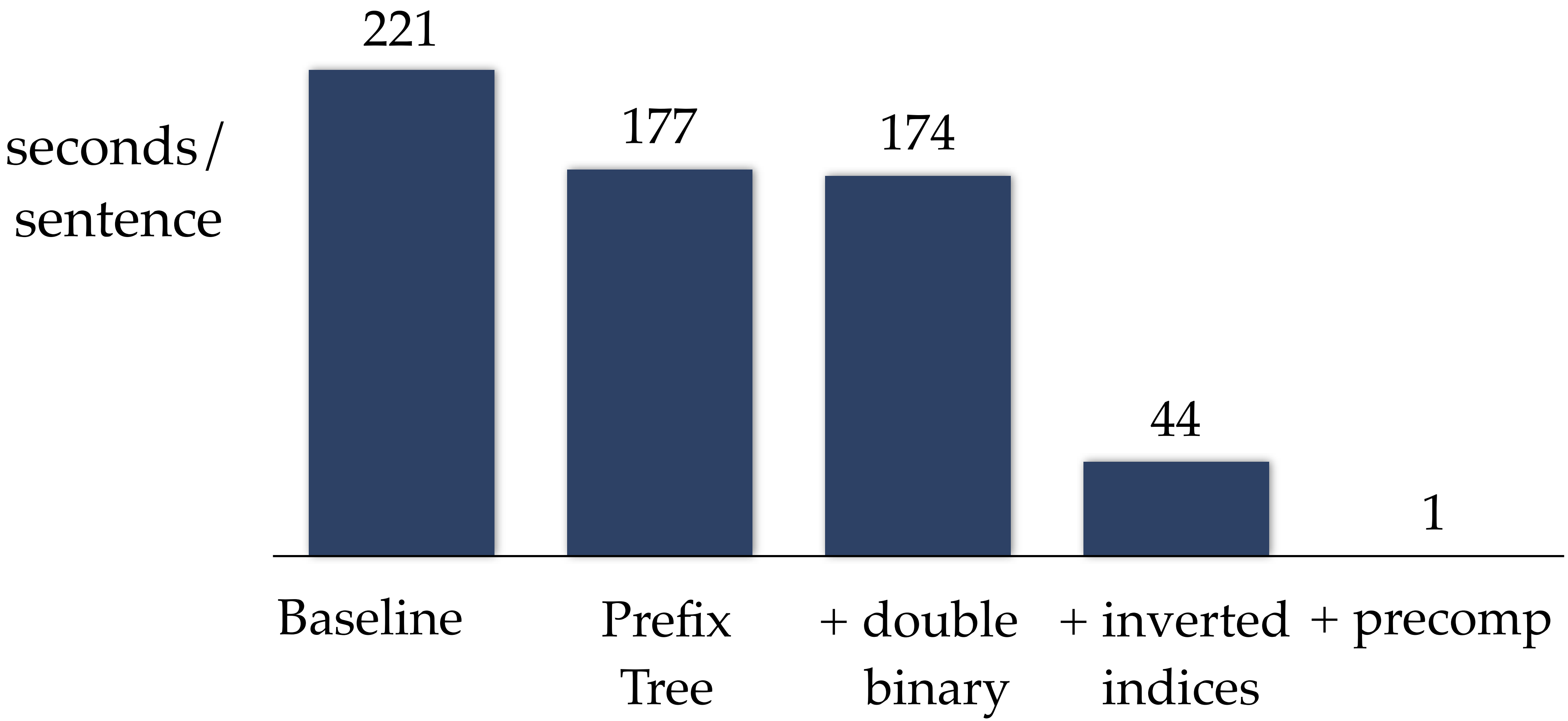
him X him

(2, 6)(2, 10)(10, 15)
(6, 10)(6, 15)

Timing Results



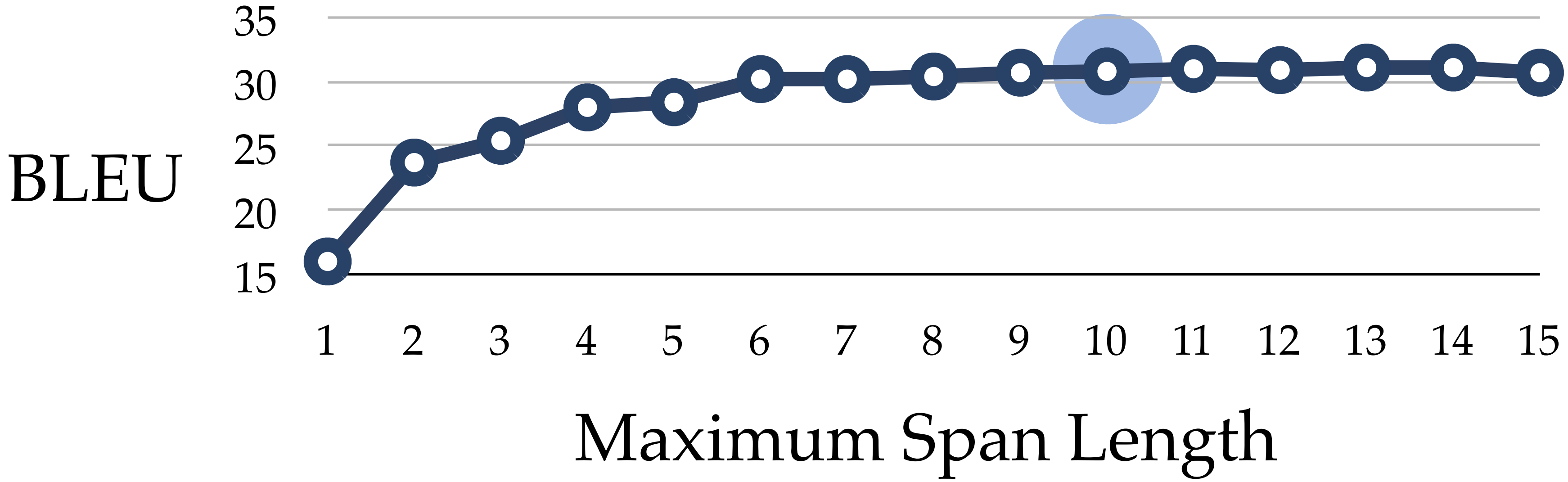
Timing Results

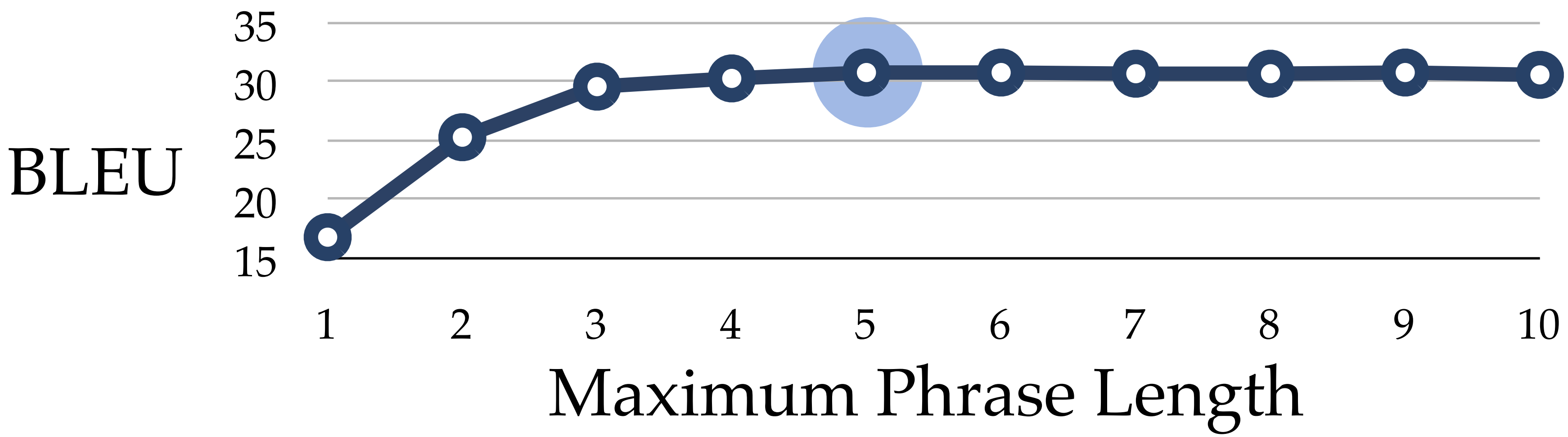


Analysis of Fixed Memory Usage

- Source Text: $|T|$
- Suffix Array: $|T|$
- Alignments: $|T|$
- Target Text: $|T|$
- Total Cost: $4|T|$
- For 27M words: about 700M
- including indices for 1000 words: about 2.1 Gb
    - for 100 words: 1.1Gb, increases time to 1.6 secs/sent

Longer Spans, Longer Phrases





The Tera-Scale Translation Model

- Task: NIST Chinese-English 2005
- Baseline Model: 30.7
- Tera-Scale Model: 32.6
- All modifications contribute to overall score
- With better language model and number translation:
  - Baseline Model: 31.9
  - Tera-Scale Model: 34.5

Open Questions

- Can we improve speed?
- Can we improve memory use? *Compressed self-indexes?*
- Uses for arbitrarily large translation models?
  - Context-sensitive models (Chan et al. 2007, Carpuat & Wu 2007)
  - Factored models (Koehn et al. 2007)
  - Syntax-based model (DeNeefe et al. 2007)
- What other algorithms can we use from bioinformatics?

Thanks

Acknowledgements:
David Chiang, Chris Dyer, Philip Resnik