

Применение сжатых индексов для полнотекстового поиска

Студент:

Вадим Соколов

Научный руководитель:

Алексей Неганов

1

Цели и Задачи



Сравнение текущих подходов



Suffix array & Radix tree



Реализация Compressed suffix array



Поиск подстроки

Inverted Index

- Для текста с разделением на слова
- 5-10% от оригинального текста
- ~50% с учетом позиции
- Невозможно восстановить оригинал

"it is what it is"

"what is it"

"it is a banana"

"a": {2}

"banana": {2}

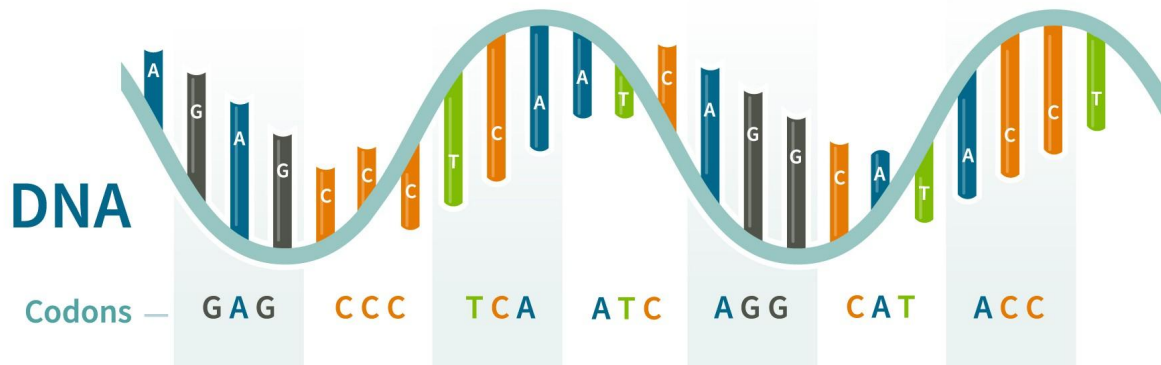
"is": {0, 1, 2}

"it": {0, 1, 2}

"what": {0, 1}

3 Проблемы Inverted Index

● ДНК и белковые структуры





Проблемы Inverted Index



Восточные языки

俄罗斯人前鋒力于发展
科学和艺术是美妙的仅

- Китайский

これが未来なので私たちは科学と芸術を緒に開発する必要があります

- Японский



Проблемы Inverted Index



Fuzzy search

Search results

This wiki is using a new search engine. ([Learn more](#))

Search

[Content pages](#) [Multimedia](#) [Translations](#) [Everything](#) [Advanced](#)

Did you mean: *andré emotions*

4

Suffix Array



Поиск подстроки



Занимает большой размер



$I = n \log n + n \log \sigma$

Idx	Suffixes	SA-Idx	Idx	Sorted Suffix
0	BANANA\$	0	6	\$
1	ANANA\$	1	5	A\$
2	NANA\$	2	3	ANA\$
3	ANA\$	3	1	ANANA\$
4	NA\$	4	0	BANANA\$
5	A\$	5	4	NA\$
6	\$	6	2	NANA\$

Suffix Array [6, 5, 3, 1, 0, 4, 2]

5

Radix Tree



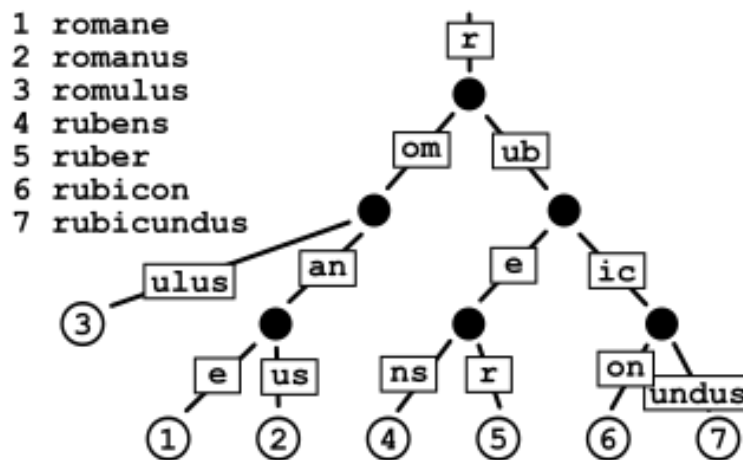
Key-value пары



Строки на ребрах



Быстрые поиск и вставка



6

Succinct Data Structures



$Z + o(Z)$ бит

Information



Self-index

Data Structure \rightarrow Data



Rank and Select

If $\text{select}(x) = y$ then
 $\text{rank}(y) = x$

7

Elias-Fano Encoding



ψ -array

Successor



Bitmap

1000100100



Таблица с отступами

Bitmap offset

8 Оптимизация Индексации

Ψ -array:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Ψ	1	3	8	12	19	26	31	2	4	9	5	7	10	11	18	23	27	29

Bitmap offset table:

offset	0	7	10
letter	a	b	c

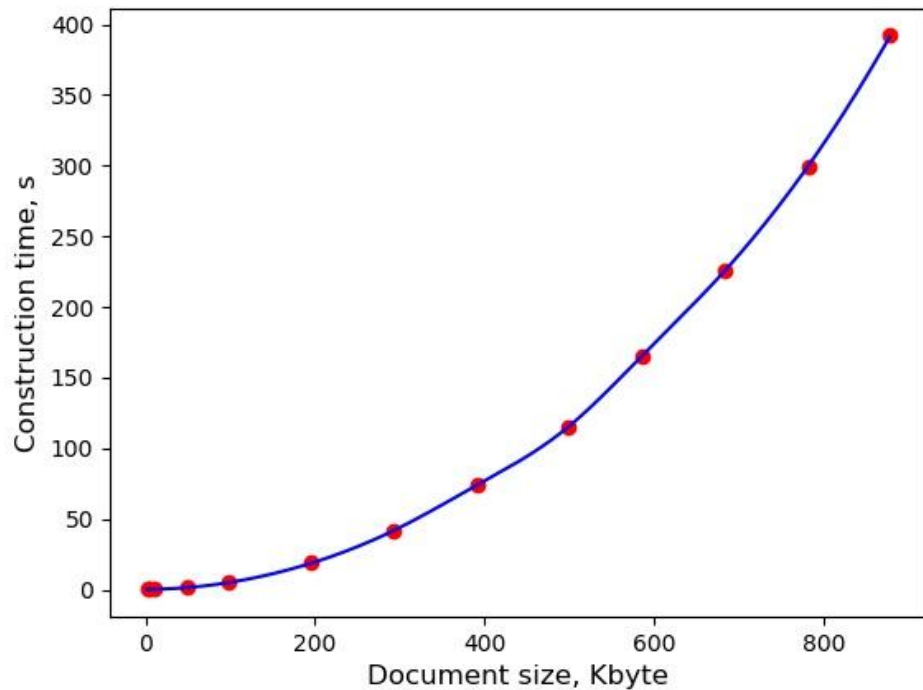
9

Результаты

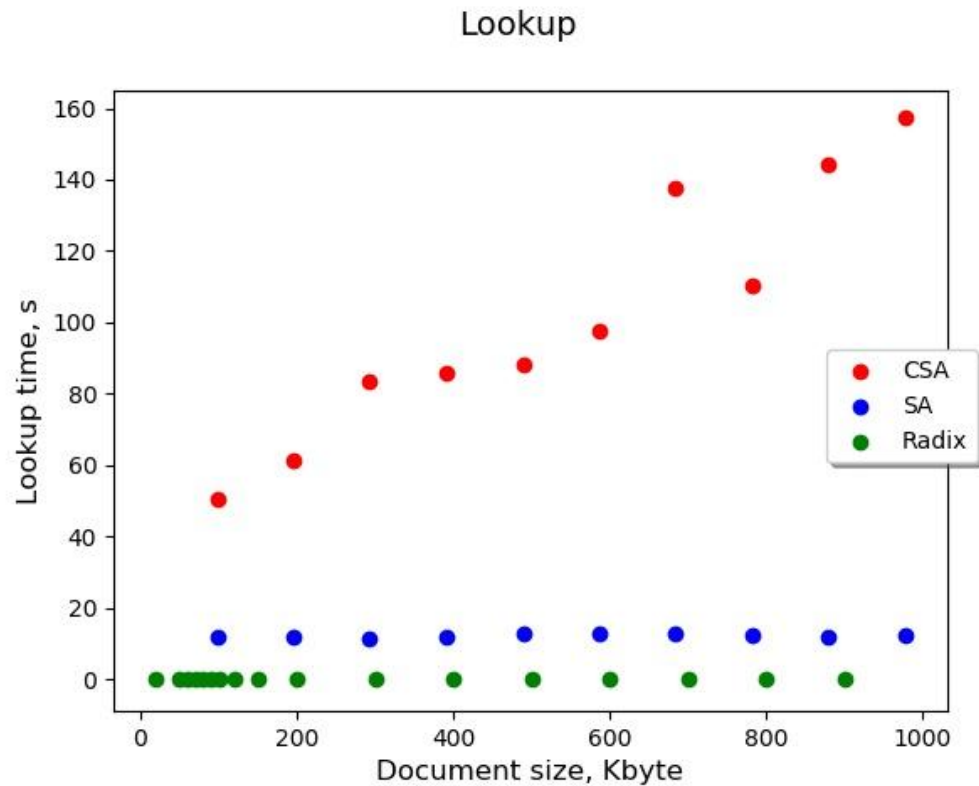
- Разработан CSA
- Поиск подстроки
- Сравнение SA, CSA, Radix tree
- Время, память, сжатие
- Различные тексты

ВРЕМЯ ПОСТРОЕНИЯ ИНДЕКСА

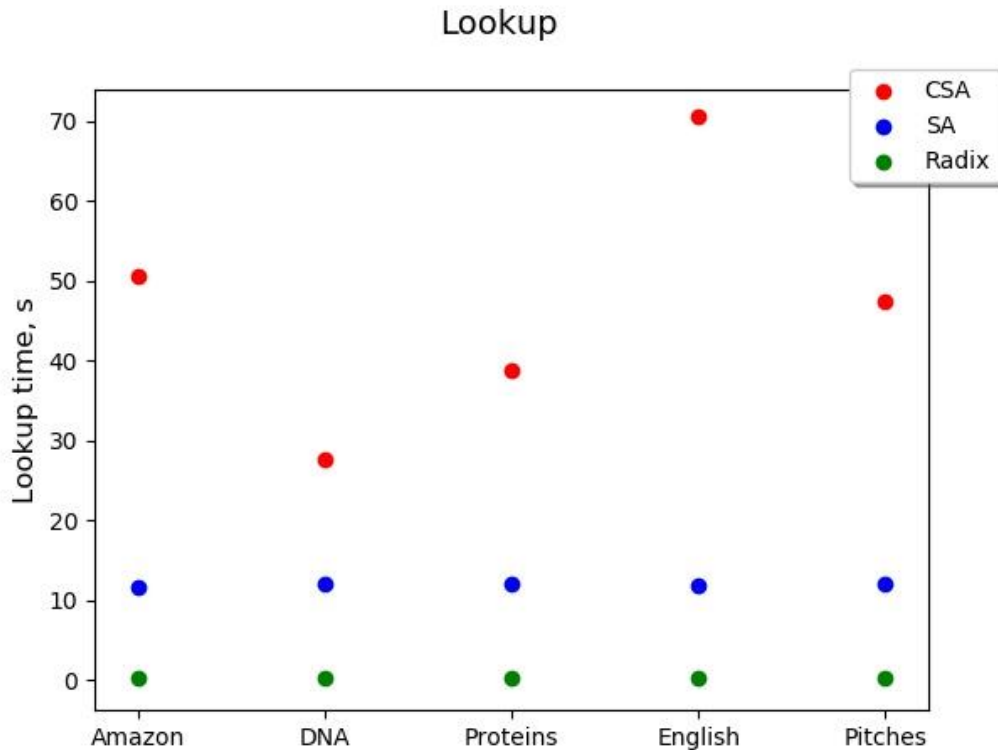
Construction time Amazon



ПОИСК ПОДСТРОКИ

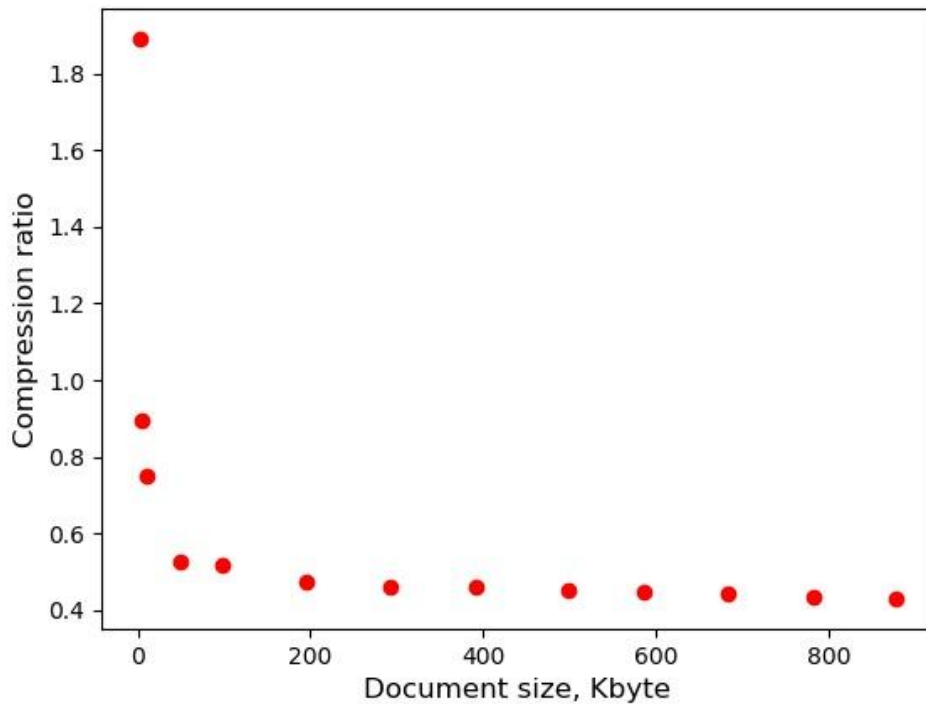


ПОИСК ПОДСТРОКИ ДЛЯ РАЗНЫХ ТЕКСТОВ

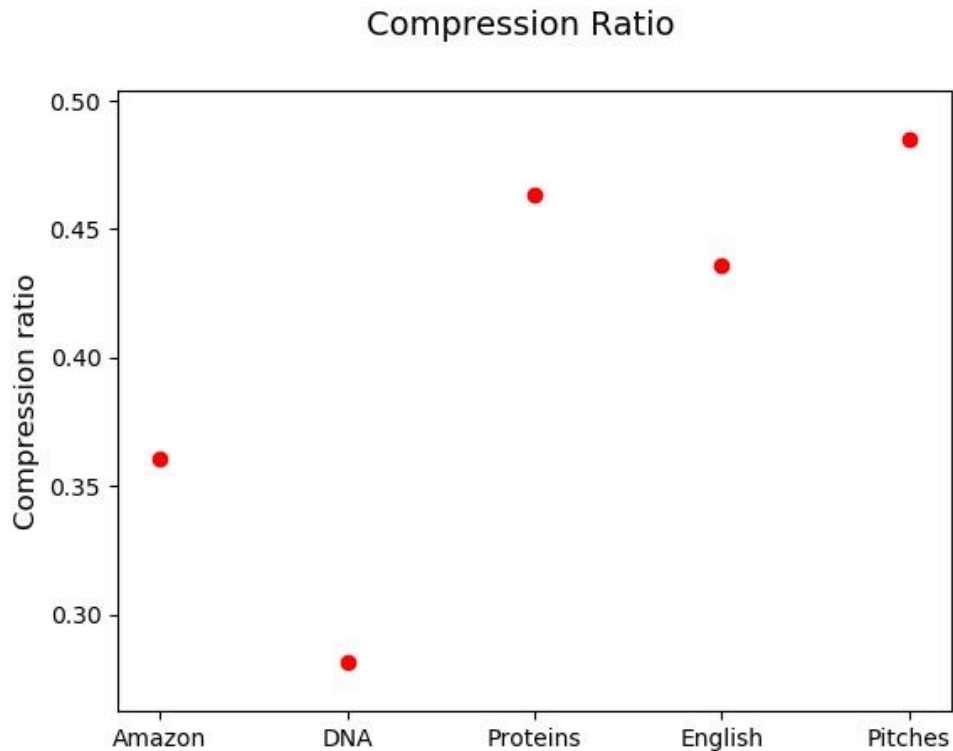


СЖАТИЕ CSA

Compression Ratio Amazon



СЖАТИЕ CSA ДЛЯ РАЗНЫХ ТЕКСТОВ

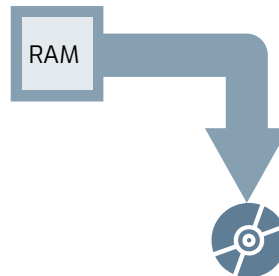


- CSA эффективен по памяти
- Для небольших документов SA эффективнее
- CSA медленнее SA
- Можно ускорить поиск
- Поиск в Radix tree за $O(1)$

11

Заключение

- Разработка алгоритмов сжатия
- Pros & cons
- Индекс во внешней памяти



СПАСИБО ЗА
ВНИМАНИЕ