



Реализация и применение строковых алгоритмов к задаче поиска повторов в документации программного обеспечения

Автор: Мишин Никита Матвеевич, группа 16.Б11-мм
Научный руководитель: к.ф-м.н., доцент С. В. Григорьев
Научный консультант: к.ф-м.н. Д. А. Березун
Рецензент: д. ф. н., Тискин А. В.

Санкт-Петербургский государственный университет
Кафедра системного программирования

13 июня 2020г.

Поиск повторяющихся фрагментов текста с целью:

- Избавление от нежелательной избыточности
- Выявление ошибок и несогласованности
- Переиспользование повторяющихся фрагментов
- ...

Задачи поиска повторов

- Поиск шаблона в заданном тексте:

Луцив Д. В. *Поиск неточных повторов в документации программного обеспечения* (диссертация, 2018)

- Поиск групп повторов:

- ▶ Неясно, насколько часто встречаются древовидные группы повторы
- ▶ Часто ищут в *JavaDoc* документации

- Позволяет решать более широкий класс задач:
Tiskin A. *Semi-local string comparison: algorithmic techniques and applications (draft-book)*
- Алгоритмы с хорошими теоретическими показателями
- Не применялись к поиску к поиску повторов в документации
- Не были реализованы на практике

Постановка задачи

Цель:

- *Адаптация алгоритмов решения задач полулокальных поиска наибольшей общей подпоследовательности и выравнивая строк к задачам поиска повторов в документации ПО*

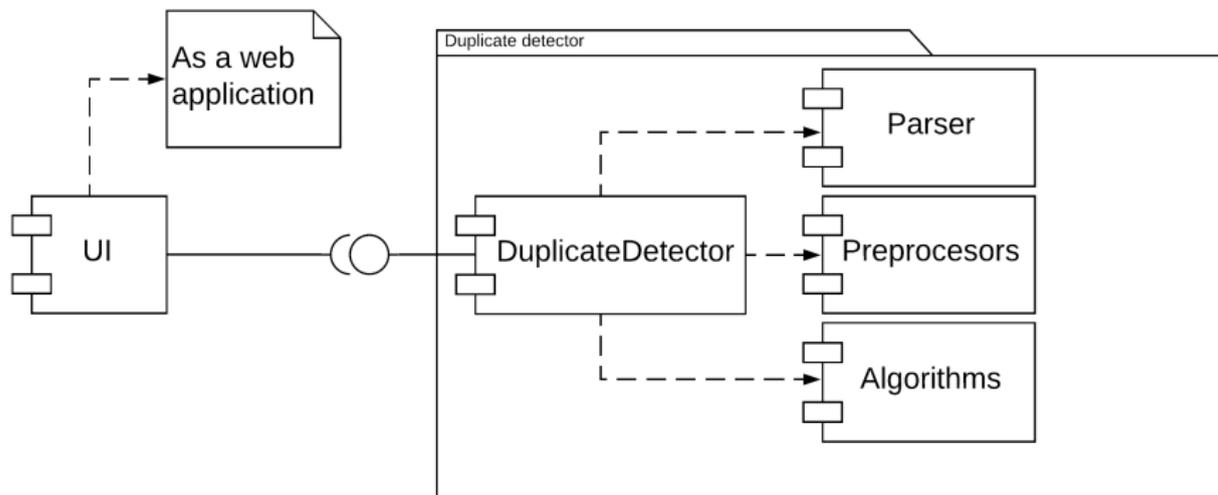
Задачи:

- 1 Исследовать существующие теоретические алгоритмы решения задач полулокального поиска наибольшей общей подпоследовательности и выравнивания строк и реализовать их на практике в виде *библиотеки алгоритмов*
- 2 Адаптировать алгоритмы решения полулокальных задач поиска *LCS* и *SA* к задаче поиска повторов в *JavaDoc* документации и реализовать соответствующее приложение на их основе
- 3 Провести экспериментальное исследование реализованных алгоритмов и анализ результатов

- Необходимые структуры данных
- Алгоритмы для решения *semi-local LCS* и *SA*:
 - 1 Распутывание кос
 - 2 Матричное умножение через тропическую алгебру
 - 3 Умножение кос
- Алгоритмы, решающие задачи на основе *semi-local*:
 - 1 Window-substring
 - 2 Поиск шаблона в тексте
 - 3 Локальное выравнивание с ограничениями:
Tiskin A. *Bounded-Length Smith-Waterman Alignment* (WABI 2019)
 - 4 ...

- Поиск по шаблону
 - ① Улучшенная асимптотическая версия алгоритма из диссертации
 - ② Алгоритмы на основе анализа решения *semi-local*
- Поиск групп повторов в *JavaDoc*
 - ① Графовые алгоритмы с матрицей расстояний на основе *semi-local* и её производных

Приложение для поиска повторов в *JavaDoc*



Визуализатор

Groups No highlighting show with variation Group name «Class MethodValidator. (MethodValidator.java)»

Current node

Class TestClass (TestClass.java)

@deprecated Included for backwards compatibility with JUnit 4.4. Will be removed in the next major release. Please use [{@link BlockJUnit4ClassRunner}](#) in place of [{@link JUnit4ClassRunner}](#).

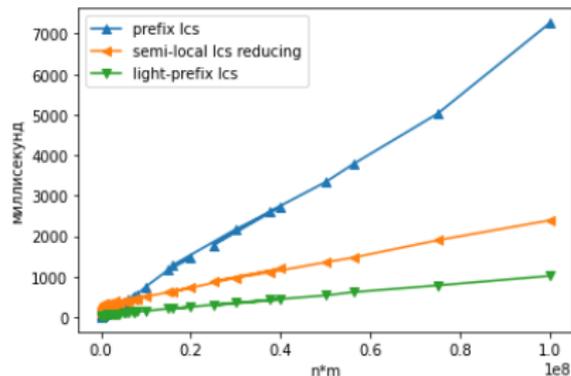
Parents

Childrens

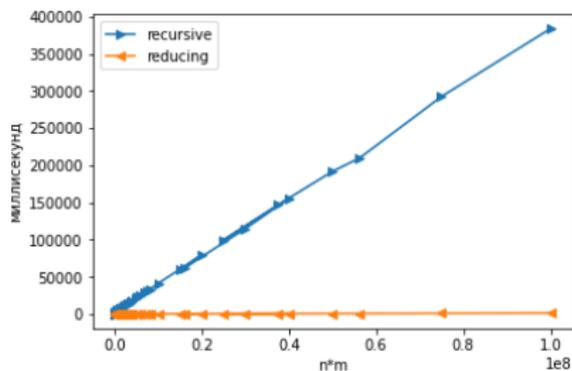
- + 🌲 Class TestMethod. (TestMethod.java)
- + 🌲 Class FailedBefore. (FailedBefore.java)
- 🌲 Class ClassRoadie. (ClassRoadie.java)
- + 🌲 Class TestClass. (TestClass.java)
- 🌲 Class MethodRoadie. (MethodRoadie.java)
- + 🌲 Class TestClass. (TestClass.java)
- + 🌲 Class JUnit4ClassRunner. (JUnit4ClassRunner.java)

Рис.: Пример визуализации для группы повторов

Апробация: Применимость на практике алгоритмов решения задач *semi-local*



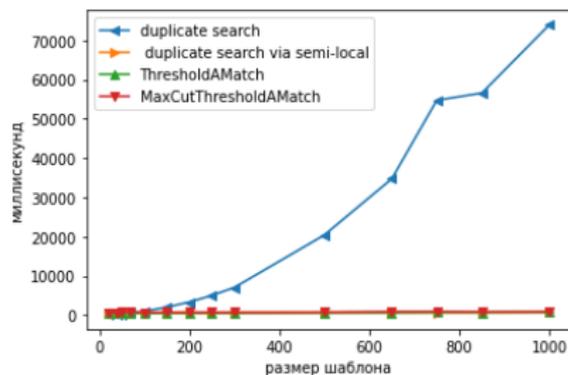
(a) semi-local lcs vs prefix lcs



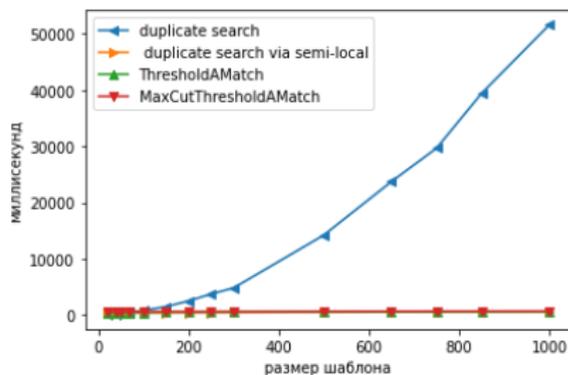
(b) recursive vs reducing approach

Рис.: Сравнение скорости реализаций, решающих полулокальные задачи

Апробация: Применимость *semi-local* к задаче поиска по шаблону



(а) Сценарий с малым размером алфавита



(б) Сценарий с большим размером алфавита

Рис.: Сравнение скорости различных алгоритмов решения задачи поиска по шаблону

Апробация: Применимость *semi-local* к задаче поиска групп повторов

Название проекта	Кол-во комментариев	Кол-во повторов	Кол-во групп	Время исполнения (сек)
slf4j	188	157	25	8
apache commons io	1284	1180	92	569
apache commons collection	610	495	50	408
gson	498	356	81	96
junit	680	539	87	163
mockito	2979	2812	164	2012
guava	4340	3662	418	8505

Рис.: Результат работы приложения

Апробация: Применимость *semi-local* к задаче поиска групп повторов

Groups No highlighting show with variation

Group name «void clear()»

Current node

`void putAll(RangeMap rangeMap)`

Puts all the associations from {@code rangeMap} into this range map (optional operation).

Parents

Childrens

`void clear()`

< Removes all associations from this range map (optional operation). >

- void putAll(RangeMap rangeMap)
 - void clear()
 - void remove(Range range)
 - void clear()
 - void remove(Range range)
 - void removeAll(RangeSet other)
 - void addAll(RangeSet other)

```
graph TD; A[void removeAll(RangeSet.C&gt; other)] --> B[void removeAll(RangeSet.C&gt; other)]; B --> C[void remove(RangeSet.C&gt; range)]; C --> D[void clear()]; D --> E[void clear()]; E --> F[void putAll(RangeMap&lt;K, V&gt; rangeMap)]; F --> G[void remove(RangeSet.K&gt; range)];
```

Рис.: Пример группы из проекта *guava*

Никита Мишин (СПбГУ)

Реализация и применение алгоритмов

13 июня 2020г.

13 / 14

- Исследованы существующие теоретические алгоритмы решения задачи полулокального поиска наибольшей общей подпоследовательности и выравнивания строк и реализованы в виде *библиотеки алгоритмов* на языке *Kotlin*
- Создано приложение на языке *Kotlin* для поиска повторов в *JavaDoc* документации на основе адаптации алгоритмов решения полулокальных задач
- Проведено экспериментальное исследование реализованных алгоритмов и анализ результатов