

ОСНОВНЫЕ КЛАССЫ БИБЛИОТЕКИ

1. Классы для работы с элементами текста:

- **Text** – для разделения текста на предложения и слова.
- **SentenceProcessing** – для разделения всех предложений текста на слова, слова в начальной форме и на ключевые слова.
- **KeyWords** – для получения ключевых слов текста с помощью интегрального метода.

2. Классы, реализующие методы автоматического реферирования:

- **MethodStatistic** – для получения реферата статистическим методом с использованием статистического алгоритма.
- **MethodSymmetric** – для получения реферата статистическим методом с использованием симметричного алгоритма.
- **MethodPosition** – для получения реферата позиционным методом.
- **MethodPositionAndStatistic** – для получения реферата позиционным-статистическим методом с использованием статистического алгоритма.
- **MethodPositionAndSymmetric** – для получения реферата позиционным-статистическим методом с использованием симметричного алгоритма.
- **MethodIndicator** – для получения реферата индикаторным методом с использованием только индикаторов и с использованием индикаторов и маркеров.
- **MethodIndicatorAndStatistic** – для получения реферата индикаторным-статистическим методом с использованием статистического алгоритма.
- **MethodIndicatorAndSymmetric** – для получения реферата индикаторным-статистическим методом с использованием симметричного алгоритма.
- **MethodIntegration** – для получения реферата интегральным методом.

- **MethodIntegrationRandom** – для получения реферата интегральным методом с применением алгоритма случайного выбора предложений.

- **MethodOfSummarization** – объединяющий класс для упрощения и стандартизации работы с остальными методами.

3. Вспомогательный расчётный класс:

- **MethodSupporting** – для выполнения различных сортировок, вычисления порога пересечения.

ДОСТУПНЫЕ МЕТОДЫ

В первую группу входят методы, предназначенные для работы с текстом:

1. **getWordsInText(String inputText)** – метод для получения всех слов текста. Параметром является исходный текст. Метод возвращает список слов типа List<Word>. Класс Word описан в библиотеке KExtract.

2. **getKeyWords(String inputText)** – метод для получения всех ключевых слов текста интегральным методом [18]. Параметром является исходный текст. Метод возвращает список ключевых слов типа List<Word>.

3. **getNumberAndParagraphs(String inputText)** – метод для разделения текста на абзацы. Параметром является исходный текст. Метод возвращает номера абзацев и сами абзацы типа Map<Integer, String>.

4. **getNumberAndSentence(String inputText)** – метод для разделения абзацев на предложения. Параметром является исходный текст. Метод возвращает номера предложений и сами предложения типа Map<Integer, String>.

5. **getWordsInSentences (String inputText)** – метод для получения слов каждого предложения в форме, в которой они употребляются в тексте. Параметром является исходный текст. Метод возвращает список слов каждого предложения типа List<List<String>>.

6. **getWordsInSentencesInInitialForm (String inputText)** – метод для получения слов каждого предложения в начальной форме. Параметром является исходный текст. Метод возвращает список слов каждого предложения в начальной форме типа List<List<Word>>.

7. **getKeyWordsInSentences(String inputText)** – метод для получения ключевых слов каждого предложения. Параметром является исходный текст. Метод возвращает список ключевых слов каждого предложения типа List<List<Word>>.

Во вторую группу входят методы, предназначенные для получения реферата:

1. **getReferatOfMethodStatistic (String inputText, int procentOfText)** – метод для получения реферата статистическим методом с применением статистического алгоритма. Параметрами являются исходный текст и объём реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

2. **getReferatOfMethodSymmetric(String inputText, int procentOfText)** – метод для получения реферата статистическим методом с применением симметричного алгоритма. Параметрами являются исходный текст и объём реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

3. **getReferatOfMethodPosition(String inputText)** – метод для получения реферата позиционным методом. Параметром является исходный текст. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

4. **getReferatOfMethodPositionAndStatistic(String inputText, int procentOfText)** – метод для получения реферата позиционно-статистическим методом с применением статистического алгоритма. Параметрами являются исходный текст и объём реферата в процентах. Метод

возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

5. **getReferatOfMethodPositionAndSymmetric(String inputText, int procentOfText)** – метод для получения реферата позиционным-статистическим методом с применением симметричного алгоритма. Параметрами являются исходный текст и объём реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

6. **getReferatOfMethodIndicator(String inputText, String markers, int procentOfText)** – метод для получения реферата индикаторным методом. Параметрами являются исходный текст, введённые пользователем ключевые слова и объём реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

7. **getReferatOfMethodIndicatorAndStatistic(String inputText, String markers, int procentOfText)** – метод для получения реферата индикаторным-статистическим методом с применением статистического алгоритма. Параметрами являются исходный текст, введённые пользователем ключевые слова и объём реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

8. **getReferatOfMethodIndicatorAndSymmetric(String inputText, String markers, int procentOfText)** – метод для получения реферата индикаторным-статистическим методом с применением симметричного алгоритма. Параметрами являются исходный текст, введённые пользователем ключевые слова и объём реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

9. **getReferatOfMethodIntegration(String inputText, String markers, int procentOfText)** – метод для получения реферата интегральным методом. Параметрами являются исходный текст, введённые пользователем ключевые слова и объём реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

10. **getReferatOfMethodIntegrationRandom(String inputText, String markers, int procentOfText)** – метод для получения реферата интегральным методом с применением алгоритма случайного выбора предложений. Параметрами являются исходный текст, введенные пользователем ключевые слова и объем реферата в процентах. Метод возвращает реферат в виде списка предложений с их номерами типа Map<Integer, String>.

Для получения списка предложений без их номеров реализован метод **getReferatToString(Map<Integer, String> referatToMap)**. Параметром метода является реферат, полученный вышеописанными методами. Метод возвращает реферат типа String.