Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Метод выделения составных частей научного текста на основе анализа распределения пикселей в сканирующей строке

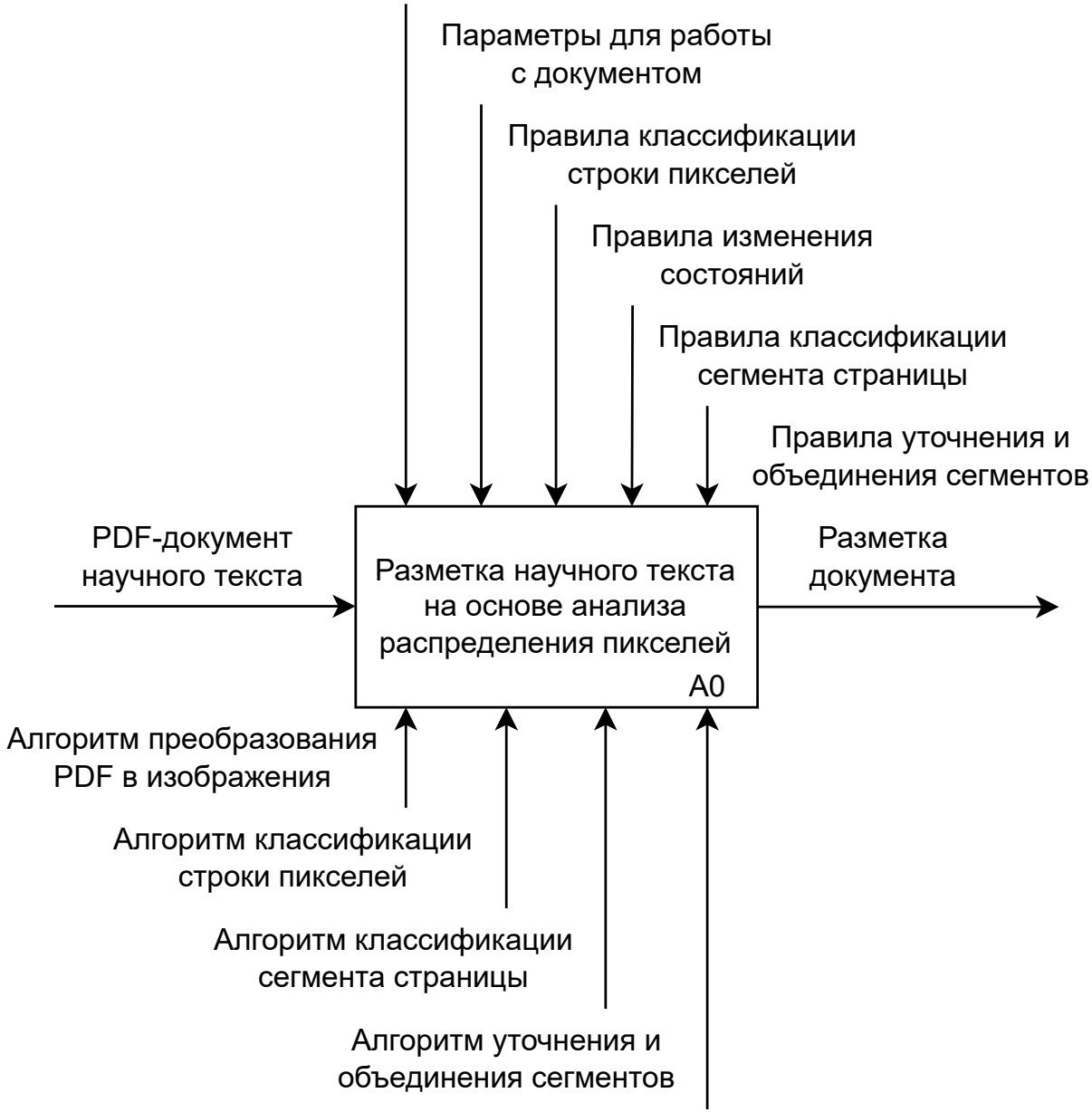
Студент: Рунов Константин Алексеевич, ИУ7-84Б
Научный руководитель: Строганов Юрий Владимирович

Цель и задачи

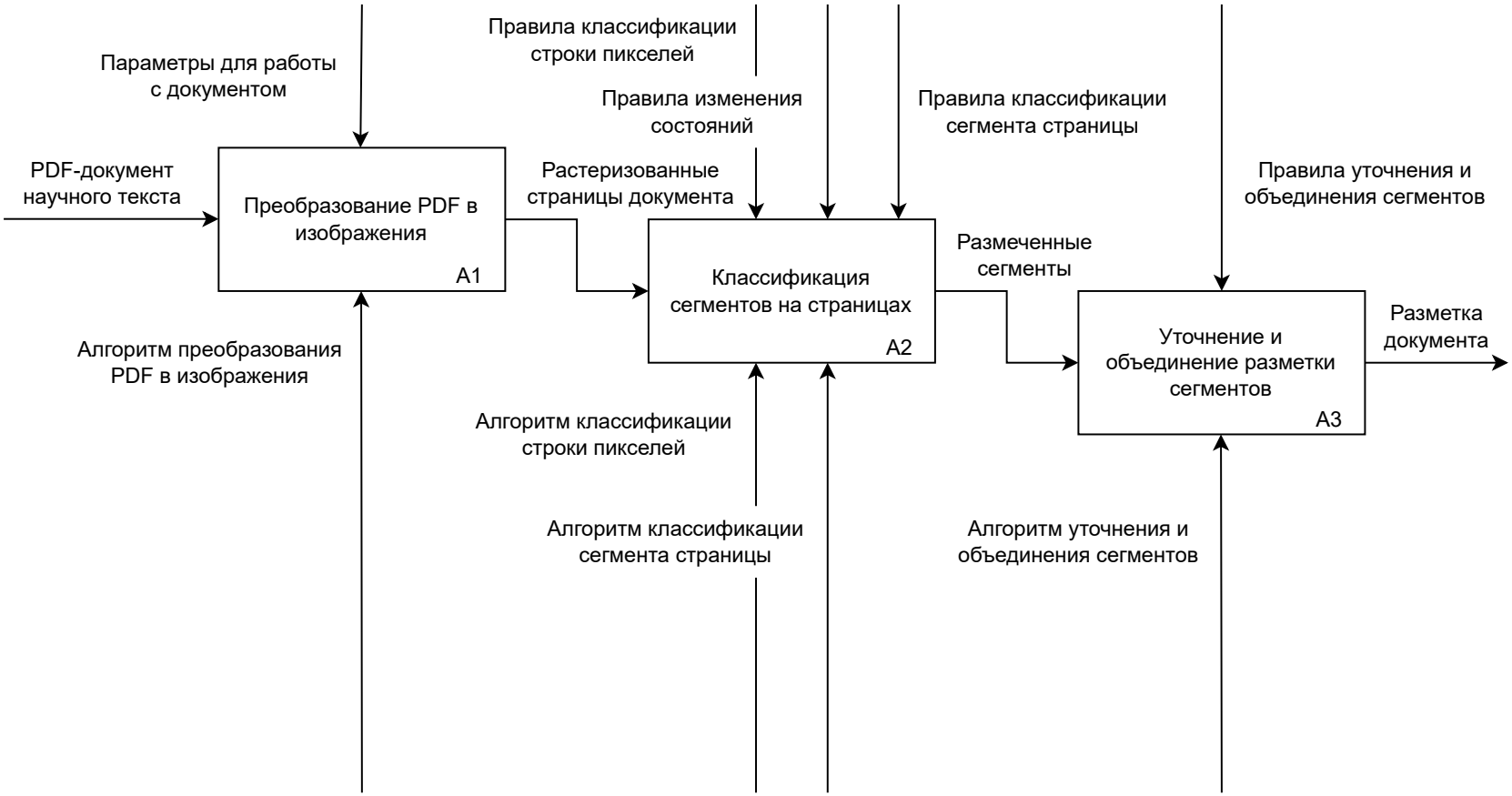
Целью работы является разработка метода для автоматического выделения составных частей научного текста, использующего только простые эвристики для классификации сегментов документа.
Задачи:

- Рассмотреть и сравнить существующие методы сегментации документов;
- Формализовать постановку задачи;
- Разработать описанный метод;
- Разработать программное обеспечение, реализующее данный метод;
- Провести исследование скорости разметки и максимального объема используемой памяти в зависимости от количества процессов, участвующих в разметке.

Постановка задачи



Постановка задачи



Классификация методов

Метод	Скорость	Гибкость	Ус-ть	СпецТреб
CCA	2	2	3	Нет
PPA	2	3	3	Нет
RLSA	1	3	3	Нет
ML	3	1	1	Да
PPA + CCA	2	3	2	Нет
Разраб.	1	3	3	Да

Гибкость — способность метода адаптироваться к различным типам макетов документов;

Устойчивость — способность метода адаптироваться к шумам и искажениям текста.

Специальное требование — позволяет сегментировать не только текст, но и такие составные части научного текста, как таблицы, листинги, схемы, рисунки, графики и прочее.

Формализация задачи

Документ $D = \{P_1, P_2, \ldots, P_n\}$ состоит из страниц $P_1, \ldots, P_n$, а каждая страница P_i содержит множество объектов $O_{i,1}, \ldots, O_{i,m}$.

Объект $O_{i,j}$ — кортеж $(x_{i,j}, y_{i,j}, w_{i,j}, h_{i,j})$, где $(x_{i,j}, y_{i,j})$ — координаты верхнего левого угла, $w_{i,j}$ — ширина, $h_{i,j}$ — высота объекта.

Требуется построить отображение

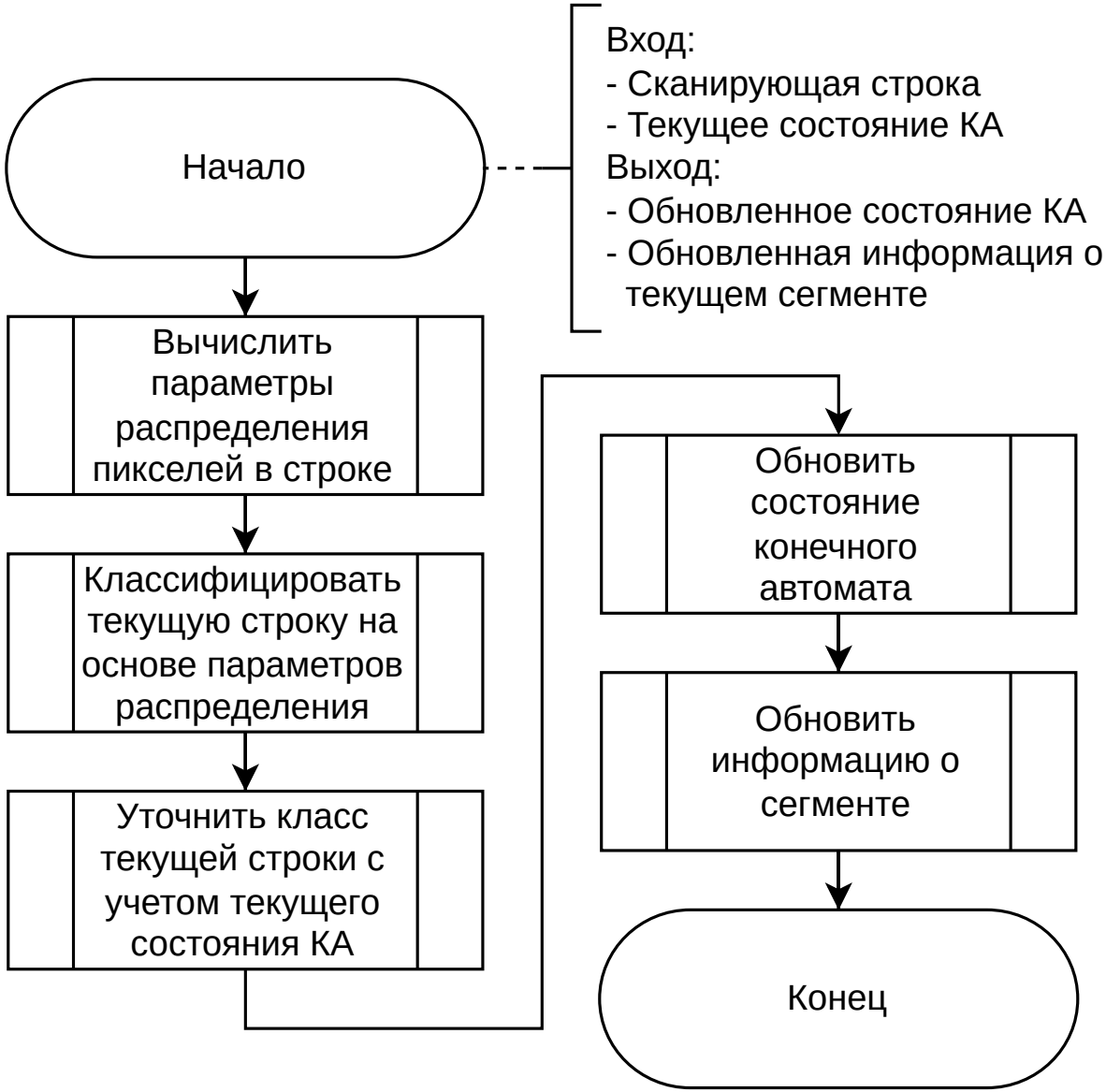
$$F : D \to \{(O_{i,j}, C_{i,j})\},$$

где каждому объекту $O_{i,j}$ ставится в соответствие класс

$$C_{i,j} = C_{i,j}(O_{i,j}),$$

область допустимых значений которого определяется исходя из требований к разметке.

Разработка алгоритма. Первичная разметка



Разработка алгоритма. Первичная разметка. Состояния конечного автомата

Разработка алгоритма. Первичная разметка. Примеры правил

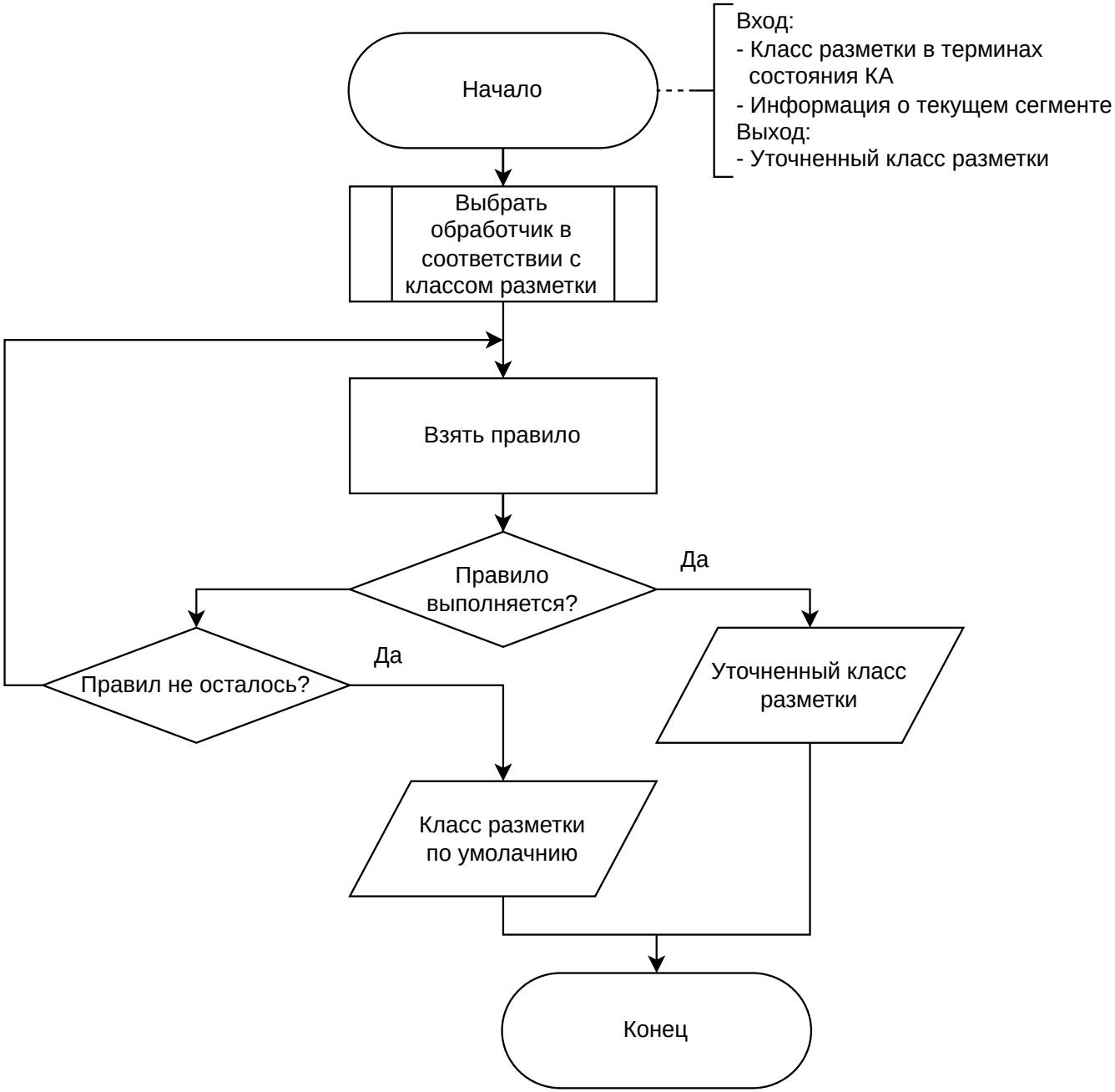
Классификация сканирующей строки:

- Если в сканирующей строке большое количество комнонент из смежных черных пикселей, вероятно строка относится к классу «Много текста»;
- Если сканирующая строка содержит цветные пиксели, вероятно она относится к классу «Цвет»;
- Если сканирующая строка содержит единственную компоненту из смежных черных пикселей длиной почти во всю ширину документа, вероятно она относится к классу «Длинная черная линия».

Обновление состояния конечного автомата:

- Если КА находится в состоянии «Фон» и встречает строку, содержащую черные пиксели, и их распределение не похоже ни на текст, ни на черные линии, КА переходит в состояние «Не определено».

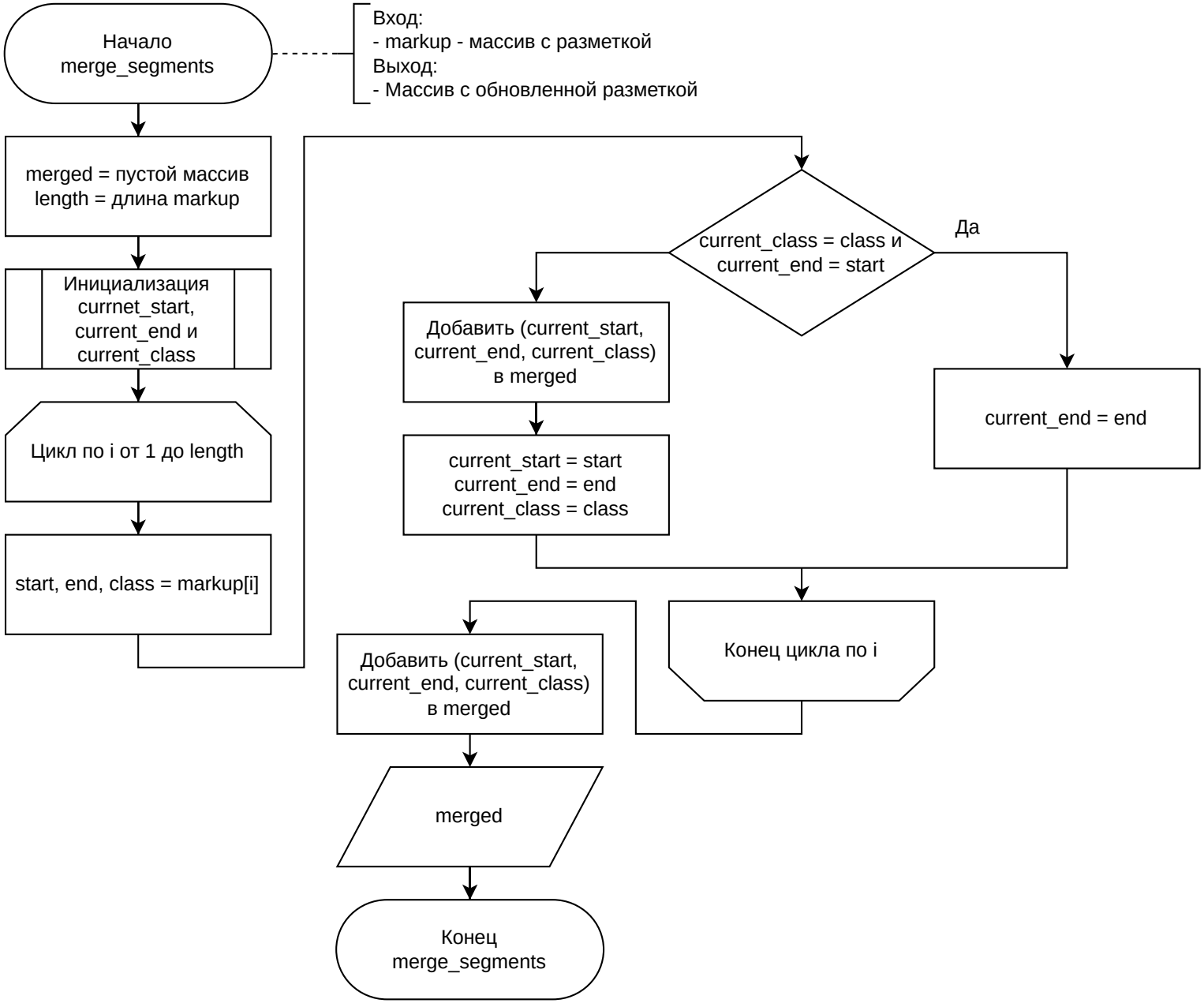
Разработка алгоритма. Уточненная разметка



Разработка алгоритма. Уточненная разметка. Примеры правил

- Если сегмент был классифицирован, как «Не определено», при этом его высота небольшая ИЛИ много строк в сегменте было классифицировано, как «Немного текста», то уточненный класс сегмента будет «Текст»;
- Если сегмент был классифицирован, как «Много текста», но при этом из информации о сегменте видно, что в нем содержится больше двух столбцов черных пикселей высотой с сегмент, то его уточненный класс «Таблица»;
- Если сегмент был классифицирован, как «Цвет», и содержит одну вертикальную линию, а также количество белых пикселей в сегменте преобладает, то его уточненный класс «График».

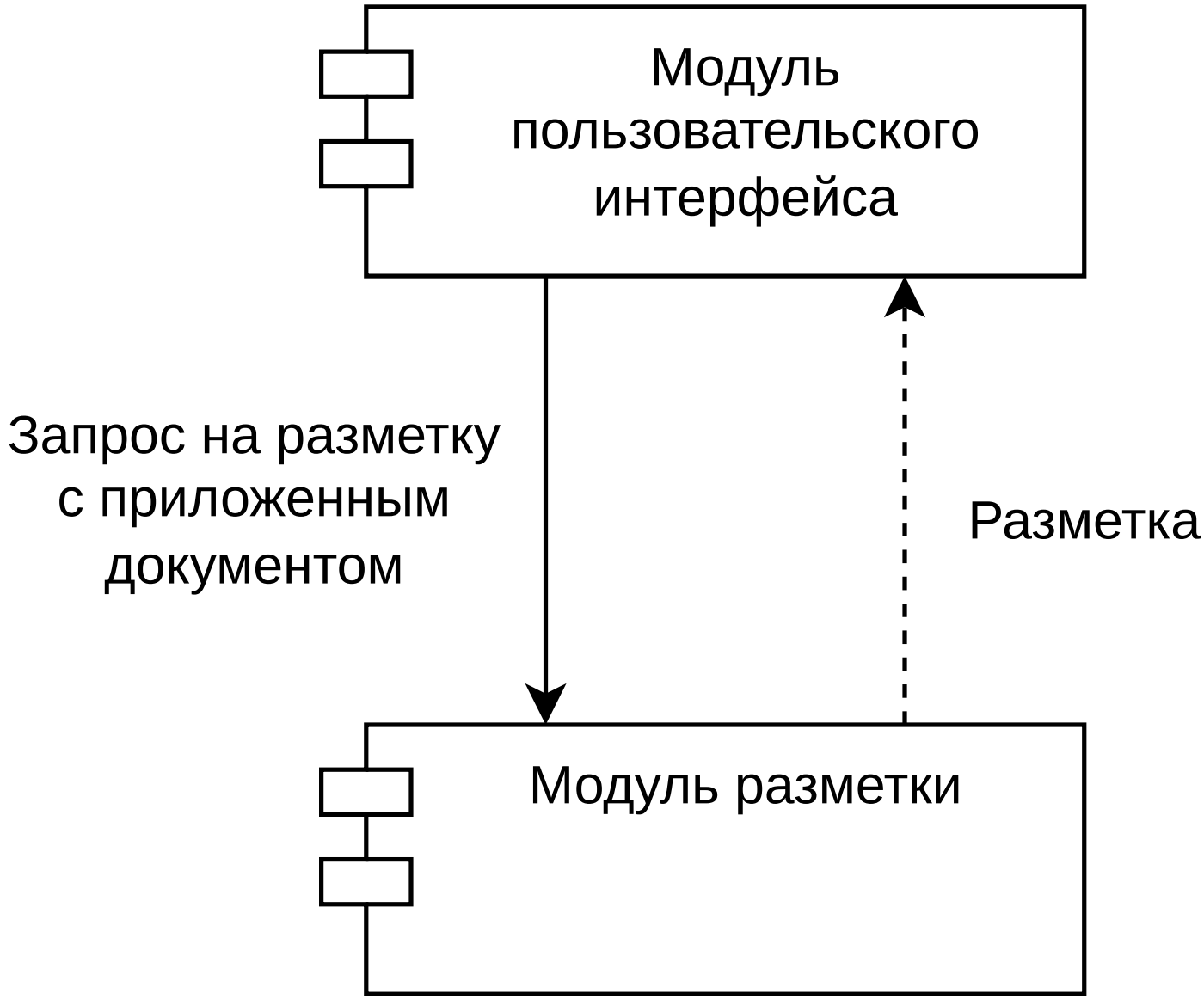
Разработка алгоритма. Объединение сегментов



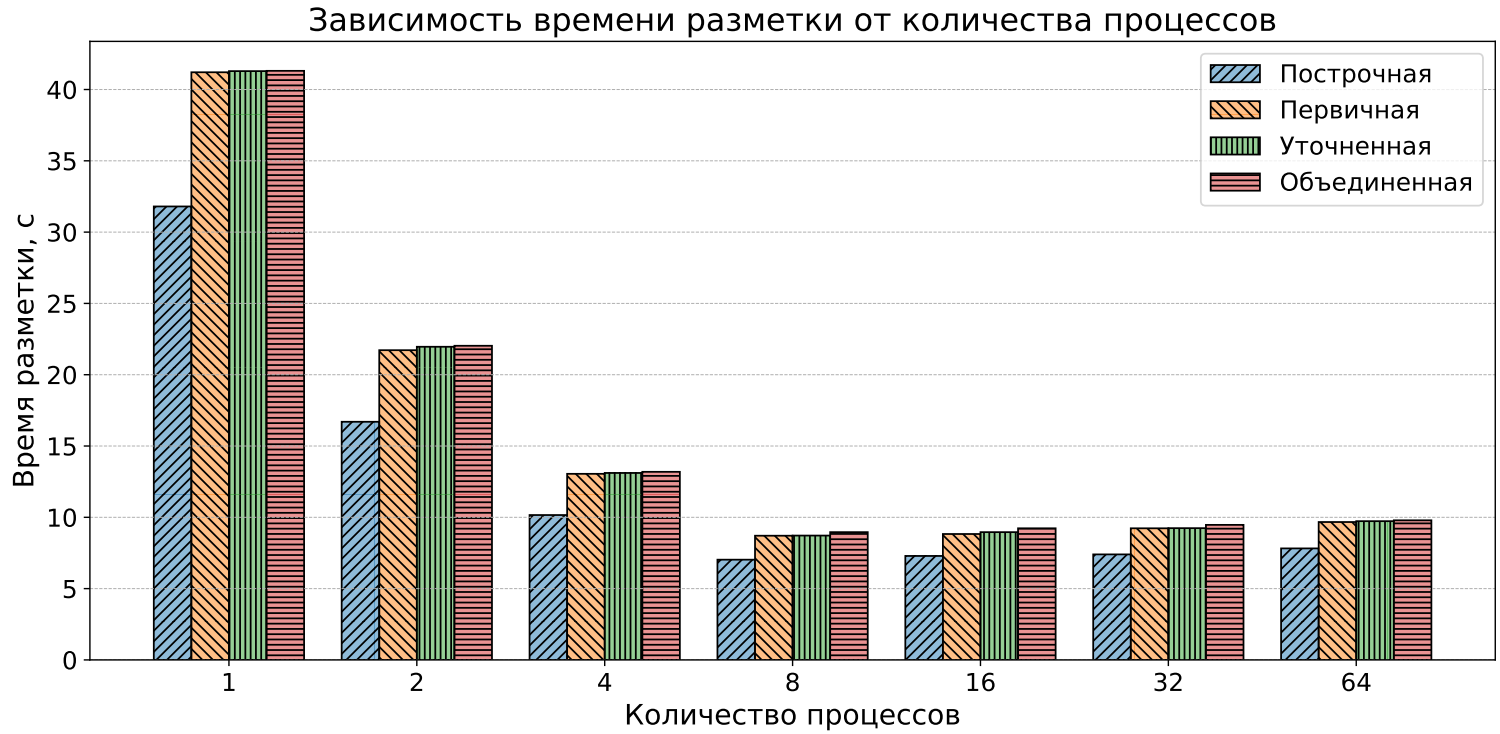
Разработка алгоритма. Объединенная разметка. Примеры правил

- Маленькие (меньше 30 px) фоновые сегменты сливаются с наибольшим соседним;
- Фоновые сегменты сливаются с соседними, если у соседей одинаковый класс;
- Небольшие (меньше 200 px) фоновые сегменты меняют класс на «Не определено»;
- Небольшие (меньше 200 px) неопределенные сегменты сливаются с наибольшим соседним.

Структура ПО

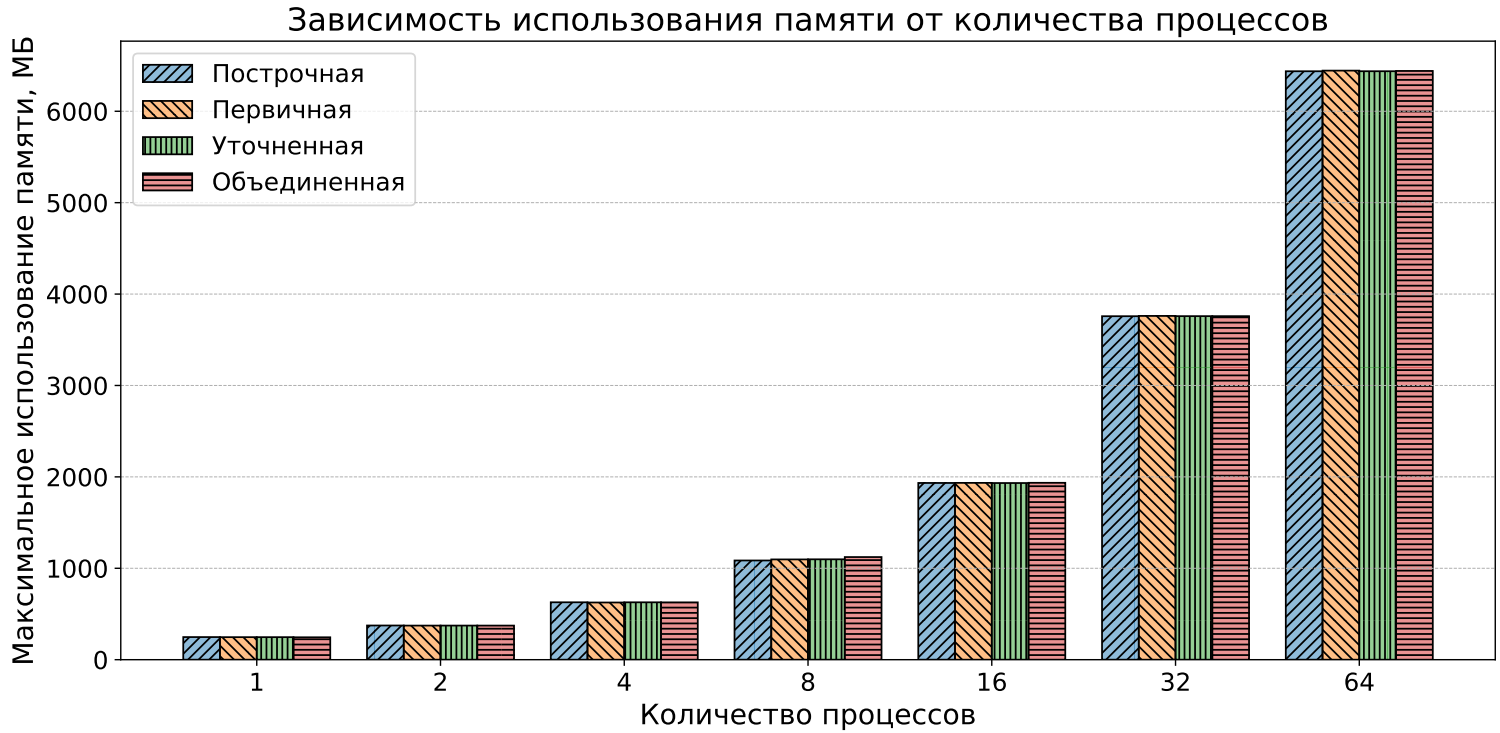


Зависимость времени разметки от количества рабочих процессов



Количество процессов	Построчн., с	Первичн., с	Уточн., с	Объед., с
1	31.80	41.21	41.29	41.31
2	16.70	21.72	21.96	22.03
4	10.16	13.05	13.11	13.19
8	7.03	8.71	8.72	8.96
16	7.29	8.83	8.96	9.23
32	7.40	9.23	9.24	9.47
64	7.82	9.67	9.73	9.79

Зависимость максимального объема используемой памяти от количества рабочих процессов



Количество процессов	Построчн., МБ	Первичн., МБ	Уточн., МБ	Объед., МБ
1	247.39	246.58	246.56	245.09
2	374.05	373.12	373.26	373.16
4	627.94	625.74	627.79	627.11
8	1085.41	1096.95	1098.18	1122.85
16	1933.34	1934.43	1932.78	1935.12
32	3757.83	3760.95	3758.53	3758.38
64	6437.75	6444.76	6438.00	6440.77

Заключение

Поставленная цель была достигнута. Для ее достижения были решены следующие задачи:

- Рассмотрены и сравнены существующие методы сегментации документов;
- Формализована постановка задачи;
- Разработан описанный метод;
- Разработано программное обеспечение, реализующее данный метод;
- Проведено исследование скорости разметки и максимального объема используемой памяти в зависимости от количества процессов, участвующих в разметке.

Дальнейшее развитие

- Добавление новых правил для увеличения точности классификации;
- Поддержка классификации формул;
- Поддержка работы с двухколоночными документами.