# Речь выступления

## Слайд 1

Благодаря продолжающийся информационной революции и поисковым системам, объемы цикрулирующей информации достигает 1 билиона веб страниц. Указанное выше обуславливает увеличение сотава и сложности программных решений в области обработки текстов на естественных языках, в основе которых лежит ряд базовых алгоритмов, в том чиссле выделение или извлечение ключевых слов.

Извлечение ключевых слов (Keyword extraction) - это задача по автоматическому определению набора терминов которые наилучшем образом описывают объект документа.

Методы способные на извлечения КС из русского языка представляют в основном своем большинстве объемное програмнное обеспечение, требующее предварительного сбора и обработки корпуса текстов относящихся к одной области, что влечет за собой узконаправленость методов, что ограничывает область применения.

## Сладй 2

Целью данной работы являетя разработка программного обеспечения для излвечения ключевых слов и словосочетаний из электронного документа на русском языке.

Для выполнения поставленной цели необходима выполнить следующие задачи: * Провести анализ существующих решений излвечения КС; * Выбрать оптимальный метод и изучить его; * Разработать архитектуру программного обеспечения; * Выбрать инструменты наимболее подходящие для решения поставленной цели; * Реализовать програмнное обеспечение с повошью выбранного метода, архитектуры и иснтрументов * Провести тестирование и эксперементы

## Слайд 3

Современная классификация методов извлечения ключевых слов подразумевает не разделение по группам, а выделение определенных признаков. На слайде под номером 3 выделены следующие признаки: * По обучению - данные свойсво говорит нам требует ли метод перед своим ипользование обучения на размеченном корпусе текстов или нет. * Лингвистические ресурсы - использует ли метод словари, анталогии и так же сборники размеченных текстов. * Матаппарат распознования - матаппрат позволяет нам понять каким образом происходит вычисление КС из документа: выделяют статистичекие, структурные, лингвистически и нейросетевые.

В рамках данной работы не расматривались методы которые используют нейросетевые подходы.

## Слайд 4

На основе выделенной класификации для алгоритмов были выделены следующие критерии: * Не требует наличиция корпуса текстов; * Умеет алгоритм работать с многокомпонентными КС; * Не привязан к предметной области На слайде под номером 4 отображена фильтрация алгоритмов по выше перечисленным критериями. Исходя из таблицы, можно сделать вывод, что оптимальным решением было выбрать метод Rake, но было бы ошибкой, так как данный метод оперирует только совместным появлением кандидатов, что приводит к очень низким результатам, что будет предемонтрировано в эксперементальном разделе.

## Слайд 5

В итоге для реализации был выбран алгорит Yake (Another Keyword Extraction Method). Особеностями данного метода являются: * Он учитывает расположение кандидата в ключевое слова в рамках предложения и текста в целом * Учитывает связь кандидата с контекстом путем построения матрицы соответствий. * Учитывает форму написания термина

Так же стоит выделить метод до этого не использовался с документами на русском языке.

## Слайд 6

Так как целью данной работы является извлечение ключевых слов и словосочетаний, а выбранный алгоритм не может с ними работать, необходимо провести модификацию. Модификация будет проводиться будет добавления н-грамм.

В нашем случаии н-грамма это произвольная цепочка слов длинной N. В записимости от параметра N название разнится. При н = 1 это униграмма, при н = 2 она биграмма, при н = 3 триграмма, а при н > 3 нграмма.

На слайде под номер 6 отображен пример разбиения предложения на н-граммы. Исходным текстом явлсяетя выражение: Автоматическое извлечение ключевых слов. Результат выделения представлен для N равного 1 до N равного 4. При н = 1 мы видим, что предложение разбито на отдельные слова, а вот при н = 2 отдельные ранее слова преобразуются в словосочетания. При дальнейшем увеличении н мы увидим более широкие словосочетания

## Слайд 7

Перейдем к самому алгоритму, на слайде с номер 7 представлена idef0 диаграмма для метода Yake. На вход ожидается текст размером не меньше 50 слов, описывающий одну предметную область. Для работы данного алгоримта необходимо передать параметры и шумовые слова. Результатов работы является список состоящий из кортежей включающий в себя Ключевое слово и его оценку.

## Слайд 8

Перейдем к слайду под номером 8. Здесь представлено с помощь idef диаграмма модуля извлечения ключевых слов. Прямоугольником выделена часть которая была добавлена при модификации метода.

Сам метод можно разделить на несколь этапов: 1. Предварительная обработка текста на данном этапе происходить разделение текта на предложения с

последующим разбиением на группы и отдельные слова. 1. Вычисление свойств термин-кандидатов - здеь происходит вычисление основных весов на основе которых будет происходить подсчет оценки кандидата. 1. Следующий блок говорит сам за себя, здесь происходит оценка каждого кандидата в термины 1. В блоке с номером A4 происходит генерация н-грамм из ранее полученных блоков текта и производится вычисление из оценок 1. В последнем блоке производится объединение результатов и подготавливается списко терминов и их оценок на вывод

# Слайд 9

Рассмотрим по ближе работу алгоритма с помощью предтавленных диаграмм на слайдах с 9 - 11 представлена диаграммы описывающий шаг за шагом работу алгоритма от начала до конца.

В начале происходит этап предварительно обработки текста, в котором просход

## Слайд 10

## Слайд 11

## Слайд 12

Для реализации программного обеспечения была выбрана архитектура MVC (модель-представление-контролер): * Модель - компонента отвечающая за взаимодействие с данными, так же предоставляет к ним доступ ниже перечисленным модулям; * Представление - это отображение состояния внутреннией системы. * Контроллер - является связующим звеном мужду представлением и моделью. Обрабатывает действия пользователя, полученные от представления и отдает команды модели. Данное архитектурное решение был выбрана из за предоставляющийся гибкости в разработке ПО. По скольку каждый модель становитяс полностью или почти независимым. Что позволяет без проблем менять фреймворки, библиотеки и другие иснтрументы.

## Слайд 13

На слайде под номером 13, продемострирована структура ПО. В представлении находиться графический интерфейс с помощью которого пользователь взаимодействует с нашей программой. Контролером являетя командный модуль, который обрабатывает все действия пользователя и перенаправляет результат работые в предсставление. Сама модель состоит из Графического модуля, модуля извлечения ключевых слов, связанного с контроллером методов так же здесь присутсвует модуль загруки данных обращающийся к файловой ситеме устройства

## Слайд 14

По результатам работы было получено программное обеспечение реализующее извлечение ключевых слов и словосочетаний из электронных документов. На слайде под номером 14 отображены условия в которых проводились исследования с разработанным методов. Для проведения эксперементов было выбрано 30 электронных документов на рускомя языке формата PDF. Текст должен соблюдать только 1 предметную область и написан на русском языке и содержать не меньше 50 слов.

Критериями оценки работы метода были выбраны следующие вечеличины: * Процент пересечения ключевых слов указанных авторами тексто с КС полученные в результате работы метов * Среднее значение; * Максимальный показатель * Минимальный

Для сравнения были взята готовые реализации 2 методов: * Textrank * Rake

## Слайд 15

Результаты работы 3 методов отображены на слайде под номером 15. Где сплошной серой линией отображено количество ключевых словом в документе. Интервальной линией отображен график алгоритма yake. Пунктирной с точкой результат пересения алгоритма textrank и точеченой линией результат работы алгоритма Yake.

## Слайд 16

Обратимся к слайду под номером 16. На данном слайде отображены результы полученные в ходе проведения эксперемента. Как мы можем видеть среднее значение пересечения ключевых слов полученных алгоритмом yake составляет 42% при 25% у алгоритма texrank и 2% у алгоритма Rake. В следствии чего стоит сделать вывод, что алгоритм, справляется лучше чем его коллеги.

## Слайд 17

Также проведено исследование алгоритма при работе с многокомпонентными терминами. На 17 слайде приведено описание документа, на основе которого проводилось тестирование.

## Слайд 18

## Слайд 19

## Слайд 20