**Слайд 1**

Здравствуйте уважаемая комиссия. Вашему вниманию представляется мой дипломный проект, тема которого – «Разработка системы кластеризации документов на базе методов машинного обучения».

В наше время трудно представить себе жизнь современного человека без машинного обучения. оно используется во многих сферах:

1. разработка приложений;

2. Рекламные компании;

3. Маркетинговые исследования и тд

И задачи кластеризации и классификации документов так же являются частью машинного обучения.

*Тут я переключаю на слайд 2*

Кластеризация — задача группировки множества объектов на подмножества (кластеры) таким образом, чтобы объекты из одного кластера были более похожи друг на друга, чем на объекты из других кластеров по какому-либо критерию.

Классификация — это система распределения обънетов на классы или разделы.

Задачи кластеризации и классификации по сей день являются довольно актуальными. К примеру, за последние десятилетки лет, стремительным ростом накапливаются большие объемов электронных документов, поэтому особое значение приобретает разработка программных средств поиска информации.

**Слайд 3**

Целью дипломной работы является создание программного средства, способного кластеризовать и классифицировать различные наборы документов.

Для достижения этой цели были выполнены следующие задачи:

* проведен аналитический обзор предметной области;
* выбрана математическая модель и метод ее решения;
* создан программный комплекс, реализующий выбранную математическую модель;
* проведен ряд экспериментов, показывающих эффективность разработанного программного комплекса

В программе предусмотрено 2 режима работы: кластеризация и классификация.

Давай начнем с кластеризации.

**Слайд 4-6**

Прочитать с экрана…

**Слайд 7**

Прочитать с экрана…

**Слайд 8**

А теперь перейдем к примеру решения кластеризации.

На этом слайде показаны файлы с их содержимым. На них я и буду демонстрировать работу алгоритма.

**Слайд 9**

*Шаг 1*. Готовим данные к обработке:

*Шаг 1.1.* Считать содержимое всех файлов;

*Шаг 1.2.* Объединить слова в единый список (словарь);

*Шаг 1.3.* Удалить слова, которые не несут смысловой  нагрузки;

*Шаг 1.4.* Получитьоснову каждого слова

Проделав эти действия получаем словарь, состоящий из 15 слов следующего вида: *вмест, готов, заказл, зачет, компьютер, костюм, куп, мног, нов, нужн, отел, перенесл, почин, стар, телефон*.

*В конце добавить, что основа слова берется с помощью стеммера Потера.*

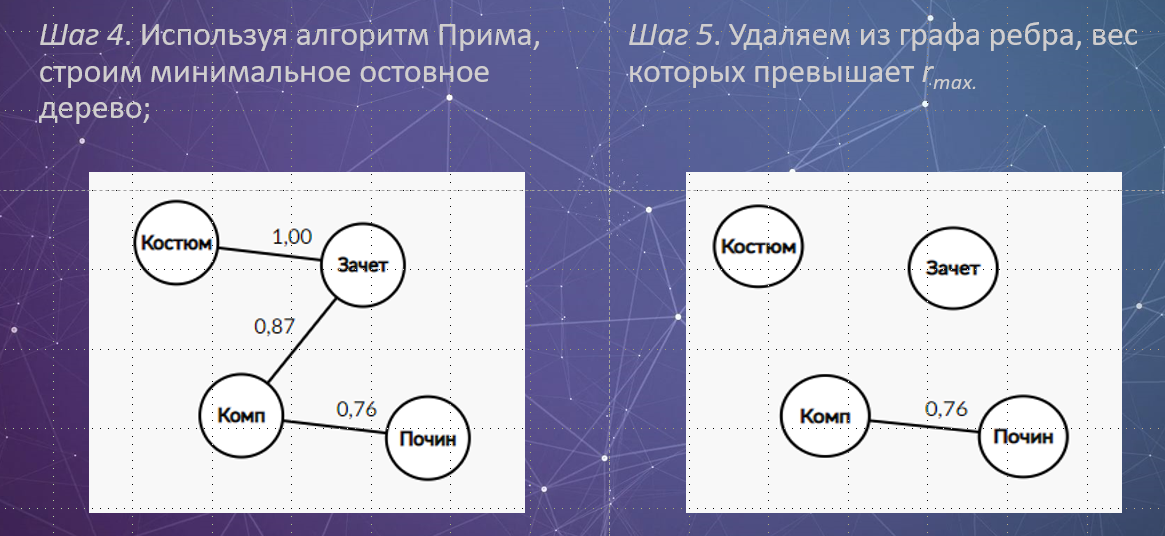
**Слайд 10**



Далее на шаге 2 мы строим матрицу схожести документов, сравнивая каждый документ со всеми остальными.

На шаге 3 мы нормализуем полученную на предыдущем шаге матрицу.

**Слайд 11**



После того как мы нормировали матрицу схожести документов, на ее основе строится граф и ищется минимальное остовное дерево. Оно выглядит следующим образом (и показываю на шаг 4).

**Слайд 12-13**

Теперь перейдем к постановке задачи классификации.

Читаю с доски…

**Слайд 14**

На 1 шаре мы считываем файл, содержащий информацию о классах, берем их ключевые слова и объединяем в словарь, предварительно получив их основы. Затем считываем файлы, которые нужно классифицировать и в каждом файле оставляем только те слова, которые входят в словарь.

На 2-ом шаге строим матрицу, у который по горизонтали классы и файлы, по вертикали количество повторений слов документа, входящих в словарь.

На шаге 3 мы нормализуем полученную на предыдущем шаге матрицу.

На 4-ом шаге подбираем наиболее подходящий класс.

**Слайд 15**

А теперь перейдем к примеру решения классификации.

На этом слайде слева показаны файлы с их содержимым. На них я и буду демонстрировать работу алгоритма. А справа показаны классы со своими ключевыми словами и их повторением.

**Слайд 16**

*Шаг 1*. Готовим данные к обработке:

*Шаг 1.1.* Считать содержимое файла классов;

*Шаг 1.2.* Объединить ключевые слова в единый список (словарь);

*Шаг 1.3.* Считать содержимое файлов;

*Шаг 1.4.* Получитьоснову каждого слова;

Проделав эти действия получаем словарь, состоящий из 4 слов следующего вида: *зачет, компьютер, костюм, телефон*;

**Слайд 17**

Далее на шаге 2 строим матрицу, у который по горизонтали классы и файлы, а по вертикали количество повторений слов, входящих в словарь.

На шаге 3 мы нормализуем полученную на предыдущем шаге матрицу.

**Слайд 18**

На 4-ом шаге определяем класс каждого документа. Из приведенной таблицы наглядно видно, какой класс больше всего подходит каждому документу.

**Слайд 19**

А теперь перейдем к интерфейсу программы.

У меня будет какая-нибудь палочка, чтоб я могу указывать на нужно место?

На этом слайде показан интерфейс программы для режима кластеризации.

В левом верх.углу расположен переключатель режима работы.

Под ним drag-and-drop область для файлов

……….

**Слайд 20**

На этом слайде показан интерфейс программы для режима классификации.

…….

**Слайд 21-22**

Создание / редактирование файлов класс

…….

**Слайд 23**

А теперь перейдем к результатам экспериментов.

Суть 1 эксп. заключалась в выявлении зависимость времени кластеризации от количества слов в файле.

На левом графике показана сама зависимость.

На правом – ее аппроксимирующая функция. Это полином 3 степени.

**Слайд 24**

Суть 2 эксп. заключалась в выявлении зависимость времени кластеризации от количества файлов.

На левом графике показана сама зависимость.

На правом – ее аппроксимирующая функция. Это полином 3 степени.

**Слайд 25**

Суть 3 эксп. заключалась в выявлении зависимость времени классификации от количества слов в файле.

На левом графике показана сама зависимость.

На правом – ее аппроксимирующая функция. По ней видно, что зависимость линейная.

**Слайд 26**

Суть 4 эксп. заключалась в выявлении зависимость времени классификации от количества файлов.

На левом графике показана сама зависимость.

На правом – ее аппроксимирующая функция. По ней видно, что зависимость линейная.

**Слайд 27**

**Спасибо за внимание**

