**СЛАЙД 2**

Здравствуйте уважаемые члены аттестационной комиссии.

Я представляю вам работу по теме «Решение задачи диаризации методами машинного обучения.»

Цель итоговой работы использовать машинное обучения для анализа аудиозаписи и получения количества спикеров, в ней.

Данный метод называется Диаризация или Speaker Diarization, выражение обозначает выделение говорящих и определение кто когда говорит.

**Слайд 3**

В соответствии с поставленной целью в работе уделялось особое внимание представленных на данном слайде. А именно:

**Слайд 4**

Для начала я хочу объяснить актуальность задачи. Зачем нам анализировать аудиозапись и выделять человеческую речь.

В настоящий момент не существует алгоритма способного без ошибочно определить человека на записи.

Выделение дикторов из аудиозаписи используется в голосовой верификации, Анализе телефонных звонков, превращении аудиозаписи в текст, например для сабтитров в фильме или анализе операторов связи и в криминалистике, для поиска преступников с помощью распознавания речи в аудиосвязи.

**Слайд 5**

Перед разработкой программы, необходимо опередить какие есть подходы к решению поставленной задачи.

Так как работа реализована на языке Python, я использовал библиотеку Librosa, с помощью который можно извлекать признаки из аудиозаписи, присущие речи человека.

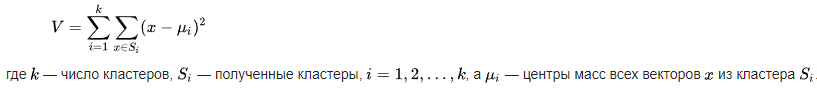
**Слайд 6**

Признаки о которые мы будем извлекать это Мел-Кепстральные коэффициенты. Это набор данных об отношении высоты звука к частоте волны аудиозаписи преобразованный с помощью специальных фильтров и представленный на мел шкале за единицу времени.

**Слайд 7**

Для кластеризации в работе использовался метод К-средних, он же KMeans. Каждый кластер с набором данных и есть отдельный говорящий на аудиозаписи.

*К средних стремится суммарное квадратичное отклонение точек кластеров от центров этих кластеров:*



**Слайд 8**

Теперь рассмотрим этапы работы.

Вначале необходимо разделить файл на части (фреймы) длинной одна секунда или менее, это необходимо для получения признаков для последующего сравнения.

Из каждого фрейма получаем набор признаков. В такой схеме мы сохраняем информацию в какой момент времени взят тот или иной признак из общей записи.

**Слайд 9**

После, полученные признаки представляем в виде точек на графике. Из пространства точек берем случайные точки для центройдов будущих кластеров. С помощью метода К средник рассчитываем в каких кластерах будут располагаться наши признаки.

**Слайд 10**

Так как модель K сердник принимает заранее выбранное количество кластеров, я использовал метод локтя для подбора оптимального количества кластеров.

Для этого было рассчитано среднее расстояние от центройда до всех точек и было произведено сравнение изменения расстояния от 2 до 10 кластеров.

На графике видно, что максимальное изменение расстояния было от первого до второго кластера.

Так программа определяет, что обучение модели K средних нужно выполнять на двух кластерах.

**Слайд 11**

После выполнение обучения модели K средних на двух кластерах мы получаем наборы признаков с маркировкой кластера. Тем самым устанавливаем, что в представленной записи есть два голоса.

**Слайд 12**

Так как заранее признаки были получены из отдельных фреймов, можно установить в какой момент времени говорил один или другой человек на записи.

На данных графиках представлены результаты анализа двух радиоинтервью, содержащих два и три голоса соответственно.

**Слайд 13**

По окончанию работы были сделаны следующие выводы:

Если уменьшать длительность фреймов качество кластеризации возрастает, но увеличивается время обработки данных

Если на записи собеседники перебивают друг друга, есть посторонние шумы это уменьшает качество кластеризации.

Использование Python библиотек, таких как librosa и matplotlib значительно упрощают реализацию данной задачи.

При данном подходе не имеет значения на каком языке представлена аудиозапись.