Московский государственный университет имени М.В. Ломоносова

**Факультет вычислительной математики и кибернетики**



Программа профессиональной переподготовки

«**Разработчик компьютерных технологий**»

Итоговая аттестационная работа

**«Решение задачи диаризации методами машинного обучения»**

Работу выполнил

**Синёв Олег Сергеевич**

Научный руководитель:

Смирнов Илья Николаевич

**Москва**

**2023**

**Содержание**

1. **Введение**
   1. **Постановка задачи** …………………………………………………………………. 3
   2. **Актуальность задачи** ………………………………………………………………. 5
   3. **Обзор существующих решений**  ………………………………………………….. 6
2. **Исследование задачи** ………………………………………………………………... 11
3. **Программная реализация** 
   1. **Случай одного диктора** ………………………………………………………..... 13
   2. **Случай двух дикторов** ………………………………………………………..…. 13
   3. **Случай трех дикторов** …………………………………………………………... 16
4. **Выводы** ……………………………………………………….…………………………. 18
5. **Заключение** …………………………………………………………………………….. 20
6. **Список литературы** ………………………………………………………………….. 21

**1. Введение**

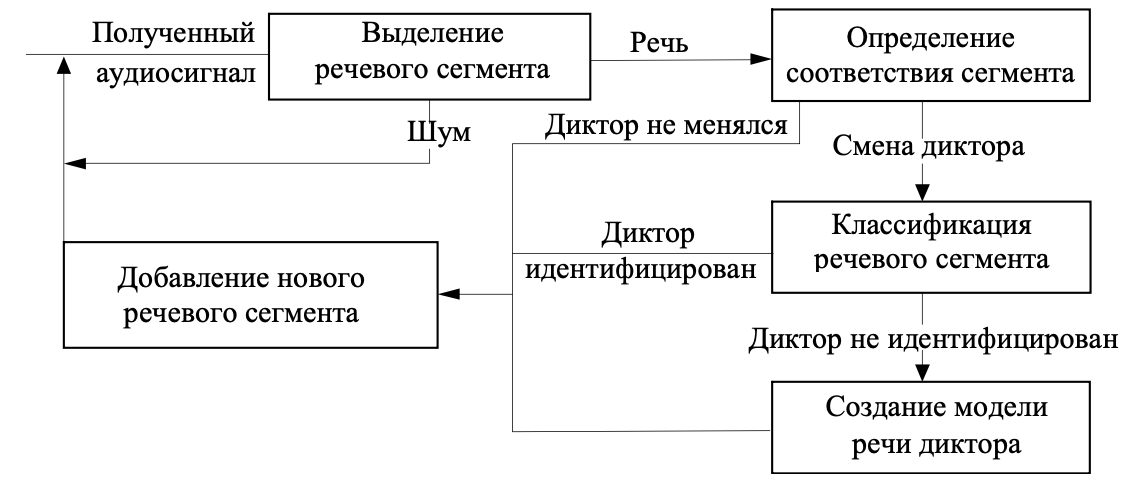
**1.1. Постановка задачи**

Цель работы – разработать алгоритм, автоматически распознающий количество собеседников в аудиозаписи. Алгоритм должен осуществлять диаризацию (Speaker Diarization — SD, которая известна в англоязычной литературе, как “Who Spoke When”).На вход программы подаётся ауидофайл содержащий диалог двух или более человек, на выход выдается количество собеседников и их время в которое они говорят.

Для достижения цели были поставлены следующие задачи:

* Удалить из аудиофайла посторонних шумов и тишины.
* Разбить аудиозапись на фреймы длительностью 1 секунде.
* Выделить из полученных феймов признаков человеческой речи.
* Кластеризовать полученные признаки с помощью модели KMeans.
* Протестировать алгоритм на реальных данных

Диаризация – Это процесс разделения входящего аудиопотока на однородные сегменты в соответствии с принадлежностью аудиопотока тому или иному говорящему. Диаризация повышает качество текстов при автоматическом транскрибировании, а также может использоваться совместно с системой [распознавания речи](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D1%81%D0%BF%D0%BE%D0%B7%D0%BD%D0%B0%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D1%80%D0%B5%D1%87%D0%B8), значительно её улучшая. Диаризация используется для ответа на вопрос «Кто сейчас говорит?». Схематично алгоритм выглядит следующим образом:



*Рис. 1. Структура типовой системы диаризации*

Первым делом, программа очищает аудиозапись от посторонних шумов и тишины, не участвующие в определении признаков речи. Затем аудиозапись делится на фреймы для определения участков с активной речью. Данный сегмент будет использоваться для сравнения с соответствующей моделью речи собеседника. В случае, если соответствие установлено, то выделенный фрейм добавляется к аудиофайлу данного диктора. В противном случае, сравнение будет произведено либо до нахождения соответствующего говорящего, либо создастся модель нового собеседника и сегмент будет помечен, как принадлежащий последнему.

Часто для улучшения работы алгоритма, программисты считаются с такими параметрами, как пол говорящего, его эмоции и качество сигнала.

Предварительное выделение участков, содержащих разного рода помехи, позволяет существенно уменьшить погрешность в работе систем диаризации. Даже различные методы определения речевой активности VAD, которые базируются на оценивании уровня сигнала, его спектра все же не решают всех проблем, которые возникают в задачах автоматического определения количества собеседников в аудиозаписи.



*Рис. 2. VAD алгоритм*

Именно поэтому пользуются популярностью методы, отвечающие за нормализацию многоканального сигнала, ведь существуют скрытые марковские модели, которые содержат содержат не два состояния (речь/тишина), как обычно в VAD-методах, а 2K состояний, где K — количество дикторов. Изюминкой данных моделей является обучаемость.

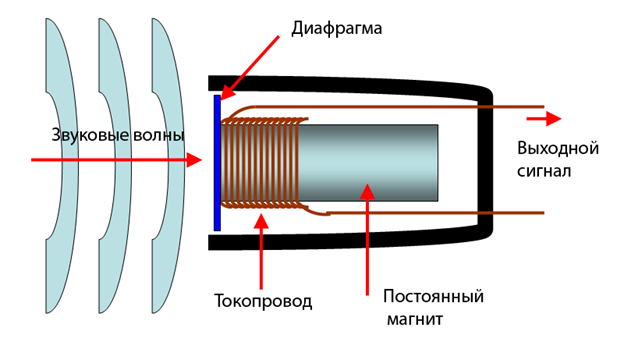
Большая часть передовых систем построения алгоритмов определения количества собеседников пользуется моделями гауссовых смесей (Gaussian Mixture Models — GMMs). Они позволяют добиться высокой точности, хоть и требуют предварительного обучения моделей, что в то же время препятствует применимости метода в режиме реального времени.

Также имеет место быть и метод параметрического представления звукового сигнала с использованием признаков MFCC (Mel Frequency Cepstral Coefficients).

**1.2. Актуальность задачи**

Несмотря на уникальность голоса человека и существенное развитие информационных технологий, данная задача все еще остается актуальной и нет пока алгоритма, работающего на все 100%. Основными источниками ошибок могут являться следующие факторы:

* Внешняя среда (шум, посторонние звуки, частоты и т.д.);
* Особенности речи и дикции (дефекты речи, паузы, тембр и т.д.);
* Устройство (искажения микрофона, погрешности кодирования аудио и т.д.).



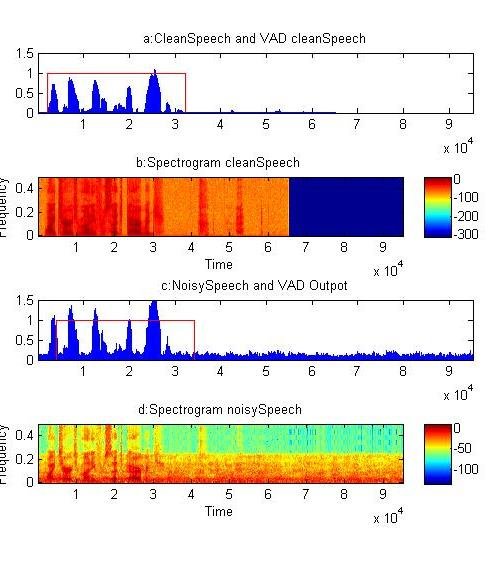
*Рис. 3. Тембры голосов Рис. 4. Пример работы микрофона*

Результаты данной работы востребованы в биометрическом поиске, голосовой верификации, разграничении прав доступа к информации, создании субтитров к видеозаписям, системах умного дома, криминалистике, в качестве помощи к системам распознавании речи и т.д.

**1.3. Обзор существующих решений**

**1. VAD**

Задача VAD метода в определении участков с активной речью от участков с тишиной и выделение первых в отдельный сегмент. Для такого разделения на фреймы применяется извлечение признаков, таких как Мел-Кепстральные коэффициенты, а также их дальнейшая классификация. Для нее могут использоваться такие модели, как Глубинная нейронная сеть (DNN - deep neural network), Гауссова смесь распределений (GMM) и Скрытые Марковские модели (CMM).



*Рис. 5. Пример работы VAD алгоритма*

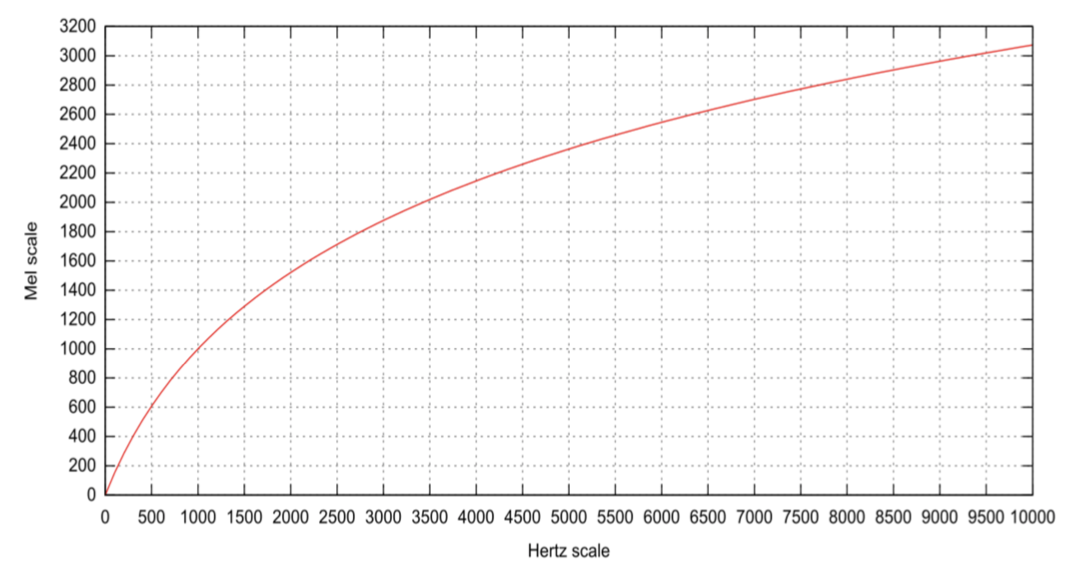
1. **Классификация алгоритмов диаризации по промежуточным признакам**

Векторные характеристики представляют собой уникальный отпечаток голоса диктора, извлечённый с помощью GMM или DNN набором фильтров.

* *I­vector: их* получают через извлечение Мел­Кепстральных коэффициентов. Вычисление косинусного расстояния для двух i-векторов различных сегментов речи, позволяет принимать решение об их принадлежности одному диктору. Как правило, при извлечении векторов аудиофайл делят на сегменты длиной от 20мс до 40мс (на фреймах такой длины изменениями голоса несущественны), зачастую пересекая соседние сэмплы друг с другом от 50% до 75% своей длины;

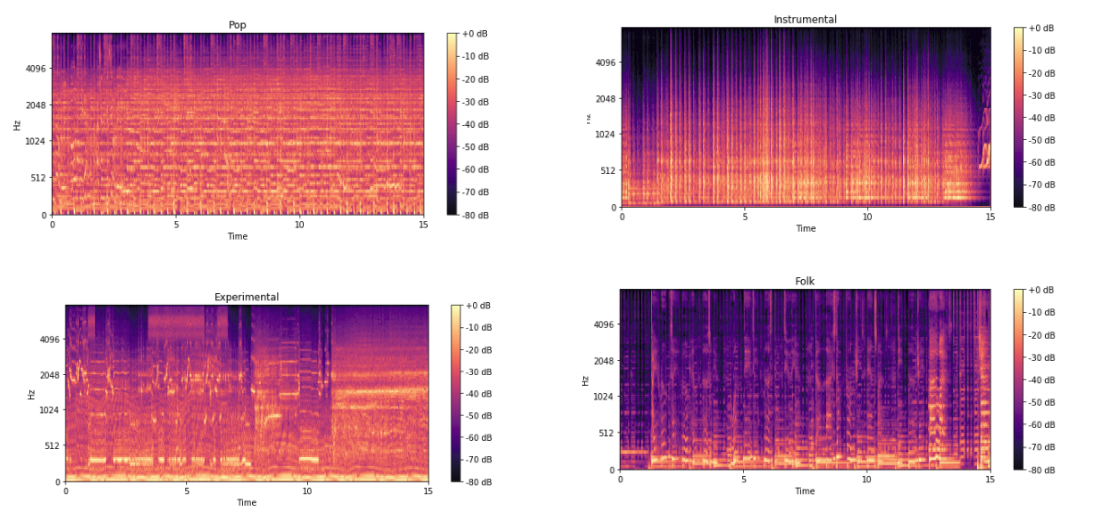


*Формула, описывающая отношение высоты чистого тона (мел) с фактической измеренной частотой (Гц).*



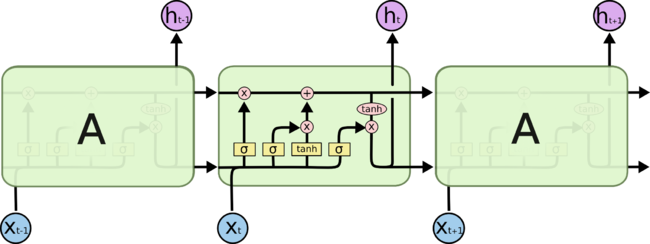
*Рис. 6. Шкала Мел*

* *X­vector:* такие векторы извлекают спектрограммы из аудиофайла и возвращают с помощью моделей глубокого обучения представление о сегменте. Несомненное превосходство метода в том, что нейронная сеть обучается специально для извлечения признаков, принадлежащих говорящему, что помогает улучшить результаты алгоритма диаризации;



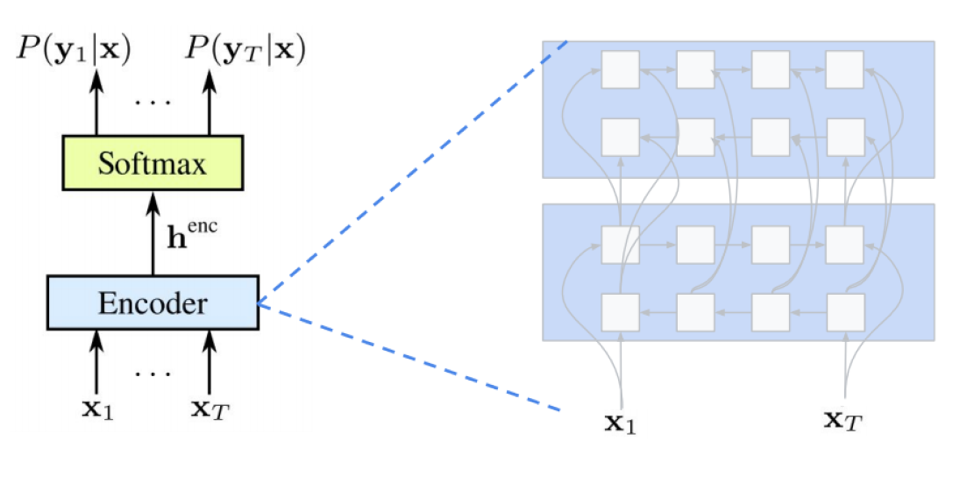
*Рис. 7. Примеры спектрограмм на музыкальных жанрах*

* *D-vector*: Данная характеристика извлекается с помощью обучения LSTM. В современном мире, алгоритмы, использующие d-векторы для описания дикторов, показывают наилучшие результаты. Даже при небольшом количестве доступных примеров речи конкретного диктора, эту характеристику можно успешно применять для его описания;



*Рис. 8. Пример работы LSTM*

* *end­to­end системы:* они не используют промежуточных представлений звука, лишь принимают на входе спектрограмму, а на выходе возвращают метку диктора.

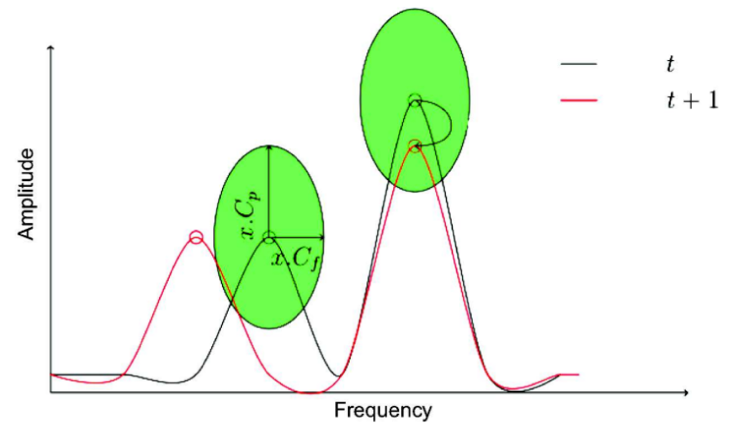


*Рис. 9. Пример работы end­to­end системы*

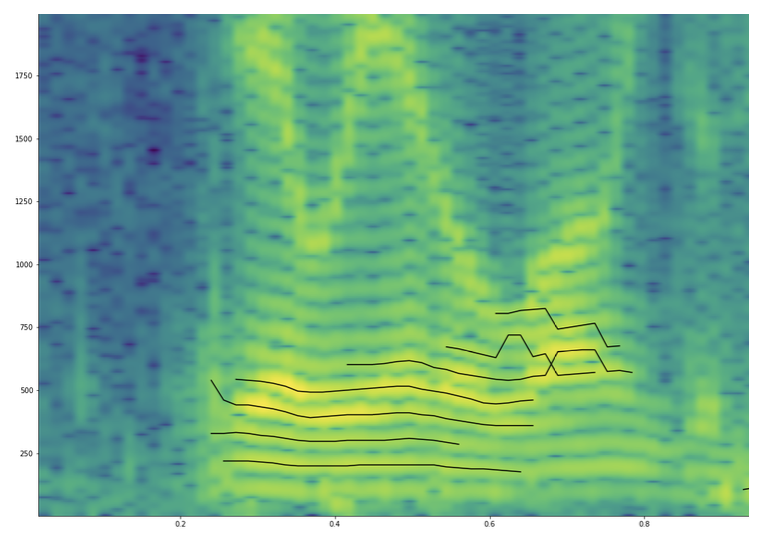
1. **Синусоидальная модель**

Идея модели заключается в использовании алгоритма отслеживания треков в амплитудно­частотной области между фреймами.

* В каждом окне делается преобразование Фурье, на базе полученной амплитудно­частотной характеристике (АЧХ) берутся n точек с наивысшей амплитудой;
* полученные точки сопоставляются с точками из предыдущих и следующих окон.



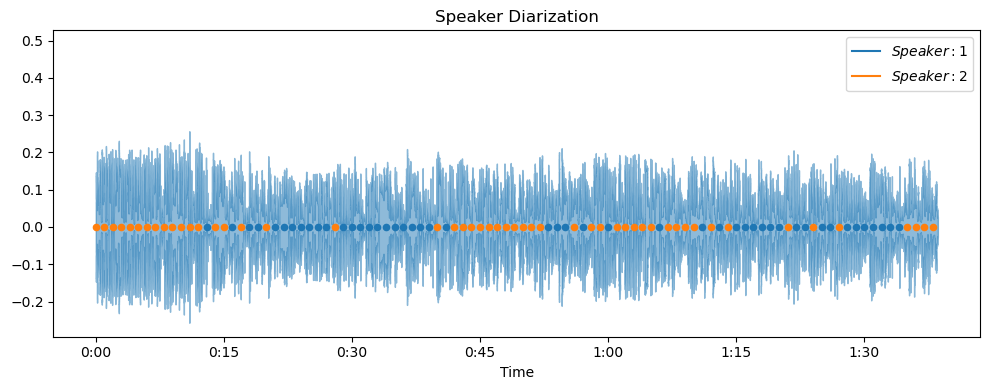
*Рис. 10. Сопоставление пиков*



*Рис. 11. Сипнусоидальная модель*

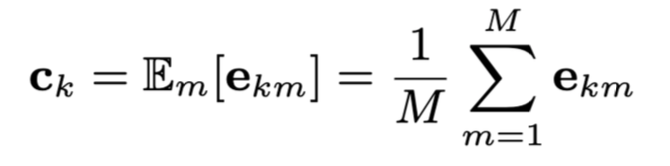
1. **Исследование задачи**

а) Первый этап - извлечение характеристик, то есть данных, на основании которых можно получить информацию о голосе того или иного человека, идентифицировать его и использовать в алгоритме для решения поставленной задачи диаризации. Суть ее - разделить голоса отдельных дикторов, потому в первую очередь необходимо заняться вопросом обнаружения участков активной речи. Предварительно аудиофайл разделяется на фреймы фиксированной длины с небольшим наложением;

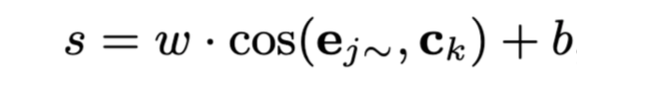


*Рис. 12. Работа алгоритма диаризации*

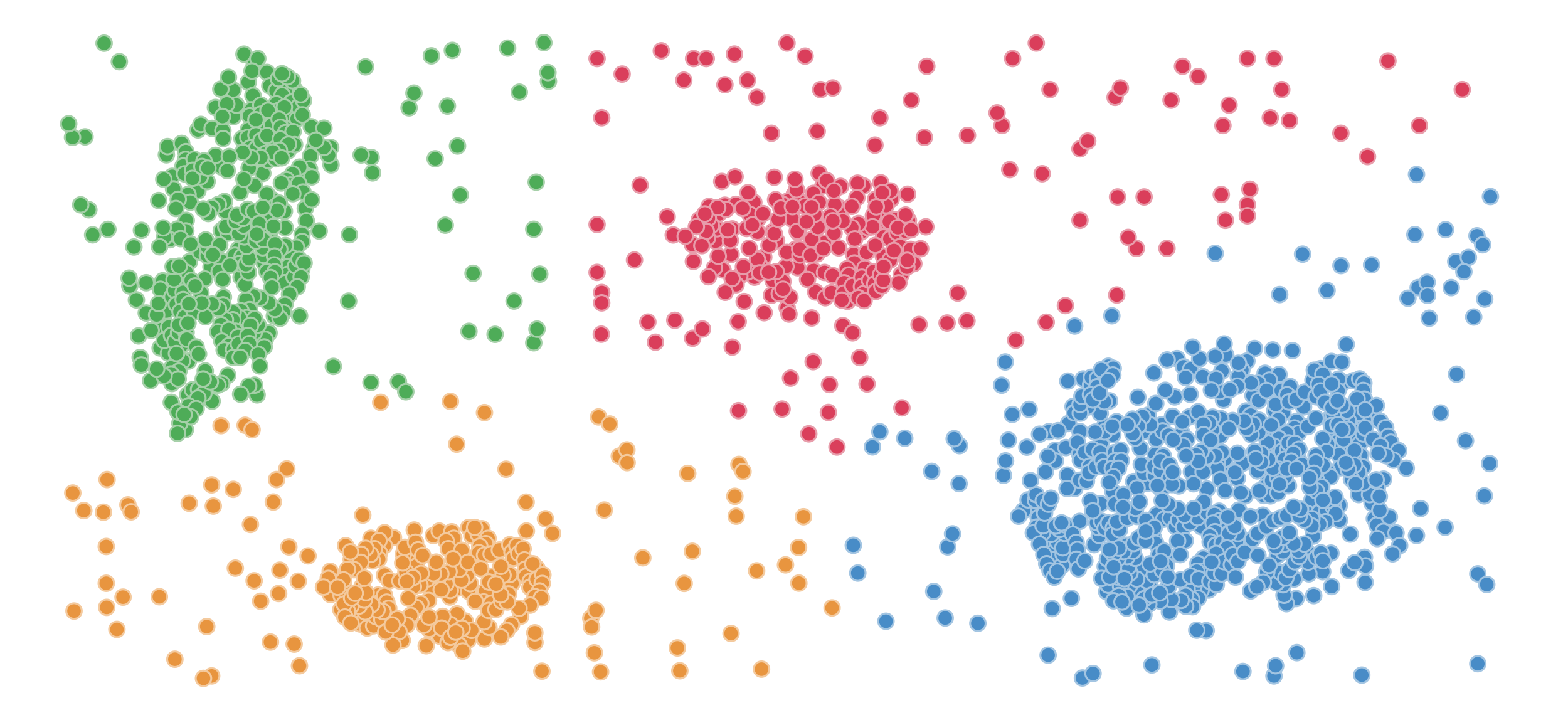
б) Данная характеристика извлекается с помощью обучения LSTM на фреймах, полученных из исходной аудиозаписи. К полученным значениям последнего слоя нейросети применяется L2 нормализация. Центроида полученного кортежа (где e – встраивание соответствующего фрейма, представляет собой уникальный голосовой отпечаток для M встраиваний;



в) Для встраиваний вычисляется косинусное расстояние и производится сравнивание с некоторым фиксированным пороговым значением;



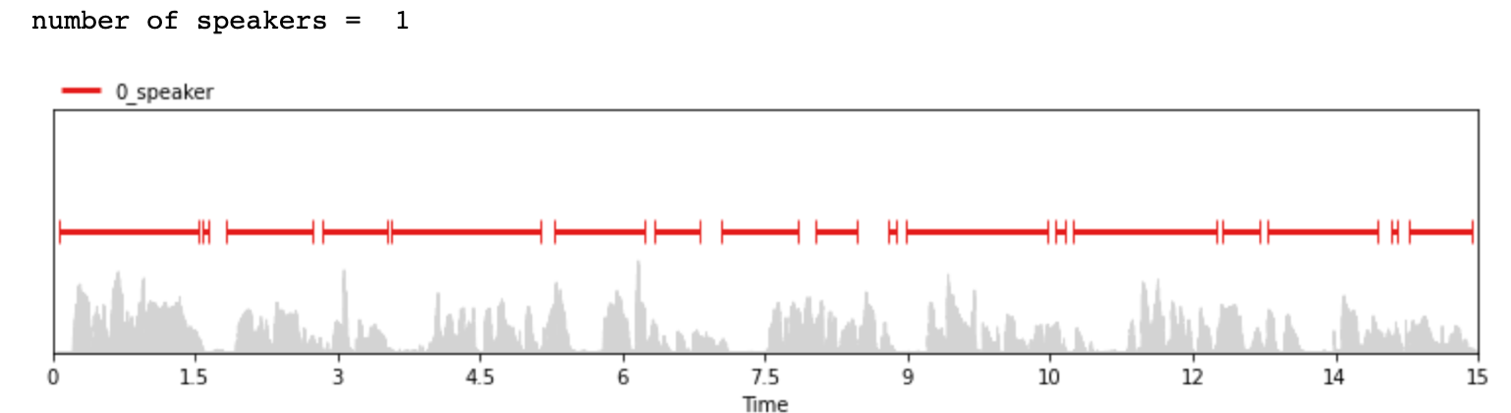
г) Делается вывод о принадлежности речи тому или иному диктору и дальнейшая кластеризация фрагментов.



*Рис. 13. Кластеризация*

1. **Программная реализация**
   1. **Случай одного диктора**

В аудиофайле содержится русская речь одного диктора женского пола. Длительность файла составляет 15 секунд, посторонних звуков практически нет. Алгоритм делит входную аудиозапись на фреймы длительностью 2 секунды с 12% наложением, threshold составляет 50%. Далее каждый сегмент проходит через VAD, предварительно пройдя шумоподавление. Следующий этап – встраивание дикторов с помощью d-векторов для каждого сегмента и последующее вычисление косинусного расстояния, за которым следует кластеризация и определение количества собеседников.



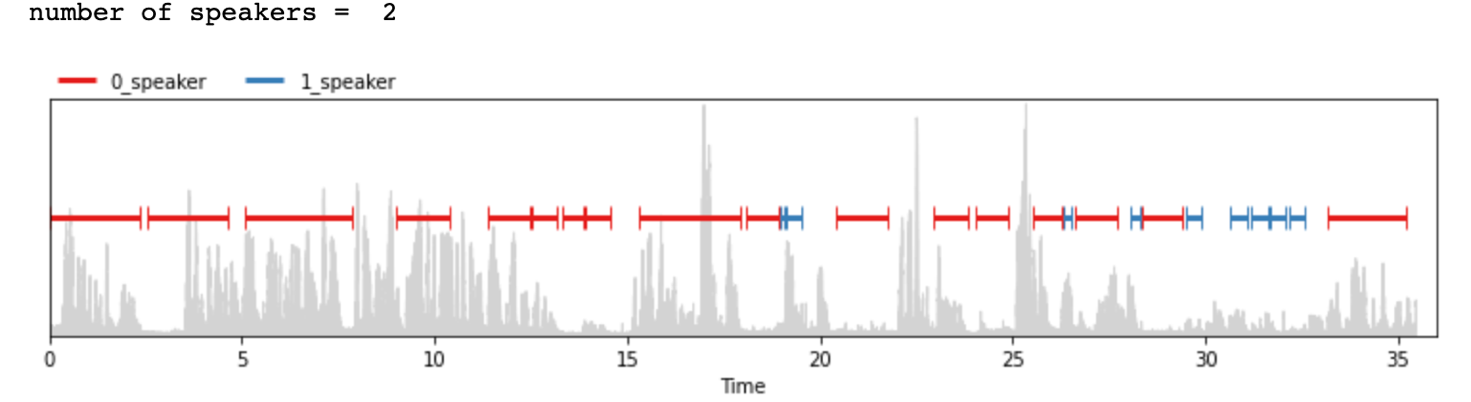
*Рис. 14*

На рисунке 14 мы видим вывод программы и его визуализацию. Светло-серым оттенком изображена исходная дорожка (оригинальная аудиозапись), красным показаны кусочки речи.

* 1. **Случай двух дикторов**

Рассмотрим несколько аудиодорожек:

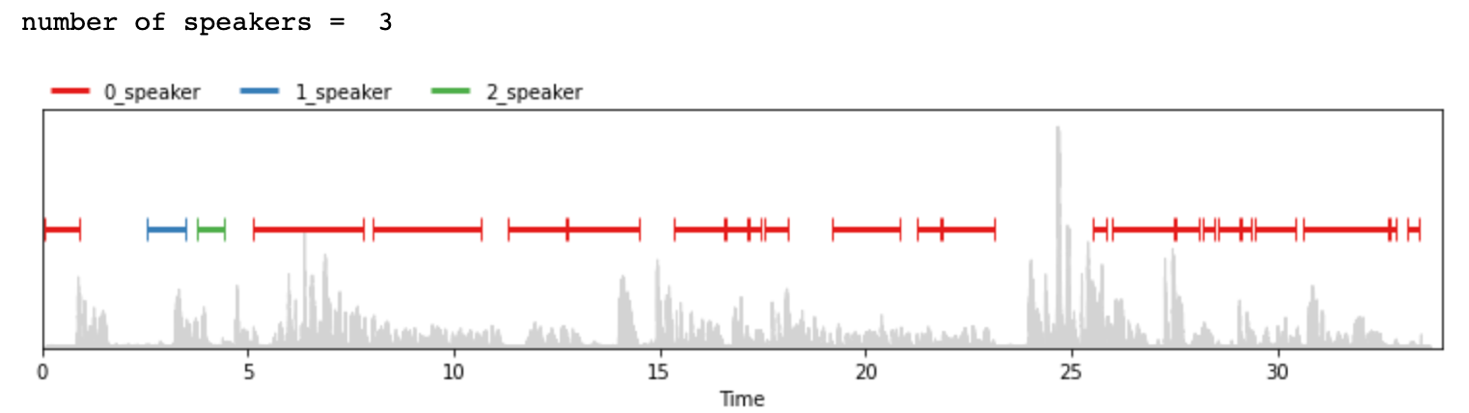
а) Аудиофайл содержит диалог двух женщин, качество записи среднее. По аудиозаписи понятно, что девушки сидят на расстоянии друг от друга, одну слышно громче, чем другую, хотя у первой есть небольшие дефекты в речи. Присутствуют несущественные шумы, помехи. Длительность составляет 35 секунд. В силу этого, некоторые параметры были изменены: деление на фреймы длительностью 3 секунды, наложение 15%, threshold 30%.



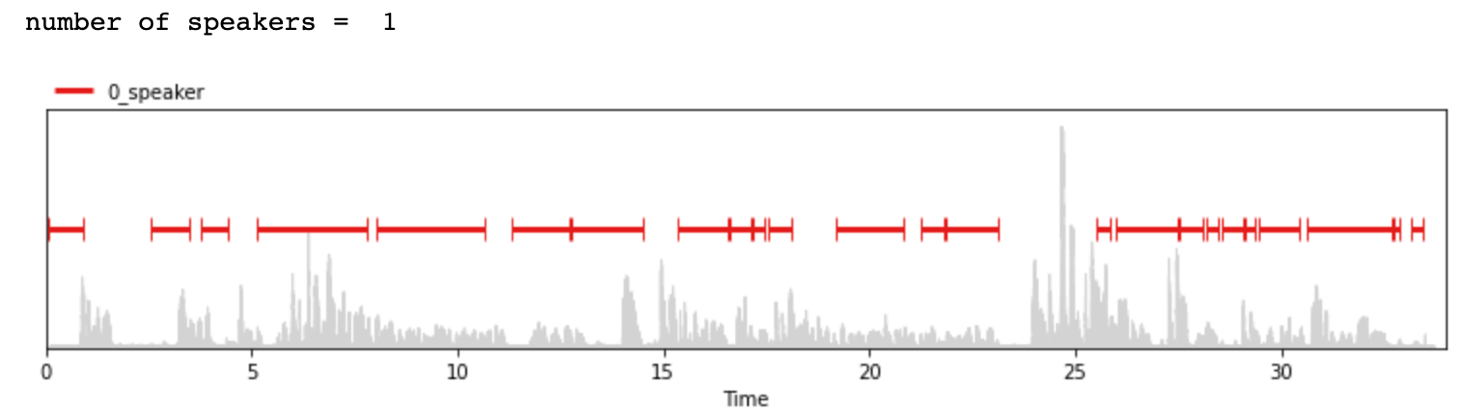
*Рис. 15*

На основании рисунка 15, можно сделать вывод, что алгоритм сработал верно. Количество собеседников двое, женщина синего цвета действительно говорила достаточно мало и больше к концу аудиофайла.

б) Рассмотрим следующую аудиозапись, содержащую беседу двух мужчин – братьев, чьи голоса достаточно похожи. Шумов меньше, чем в предыдущем случае, однако собеседники также сидят на расстоянии друг от друга. Длительность аудиозаписи около 35 секунд. Сделаем два теста. На рисунке 7 параметры сохранены, на рисунке 8 изменен только threshold с 30% до 40%.



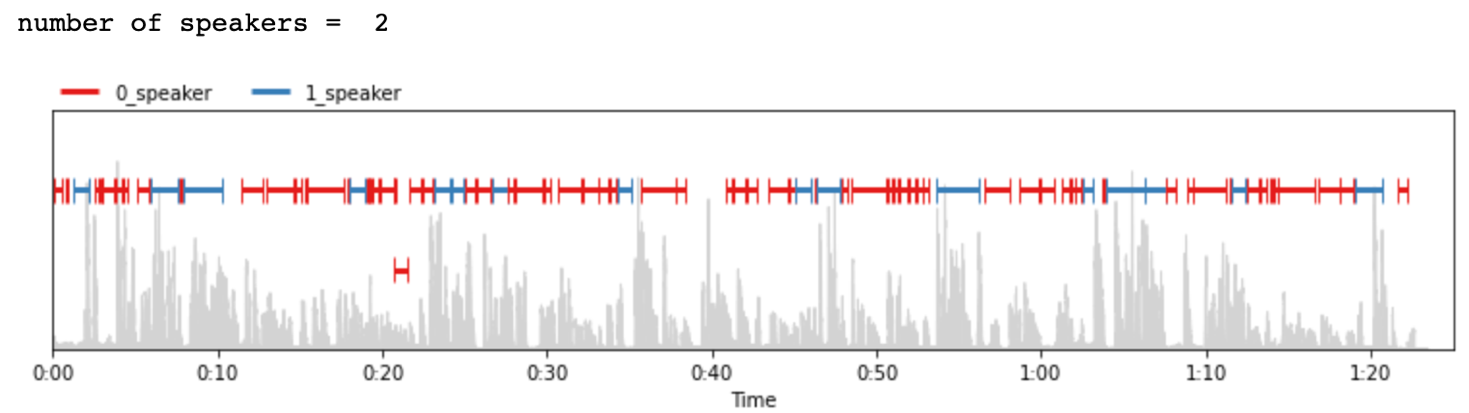
*Рис. 16*



*Рис. 17*

Вывод: в силу сильной схожести голосов, показатель threshold напрямую влияет на результат работы программы, однако не дает верного ответа. Для корректного результата требуется использование больших характеристик и более глубокое изучение задачи диаризации.

в) Аудиофайл содержит диалог мужчины и женщины на английском языке. Речь практически беспрерывна с минимальным шумом и паузами. Аудиозапись длится практически 1.5 минуты. Параметры: деление на фреймы длительностью 3 секунды, наложение составляет 15%, threshold 30%.



*Рис. 18*

Ответ получен верный, однако некоторые сегменты с речью были проигнорированы программой как фрагменты тишины/шума.

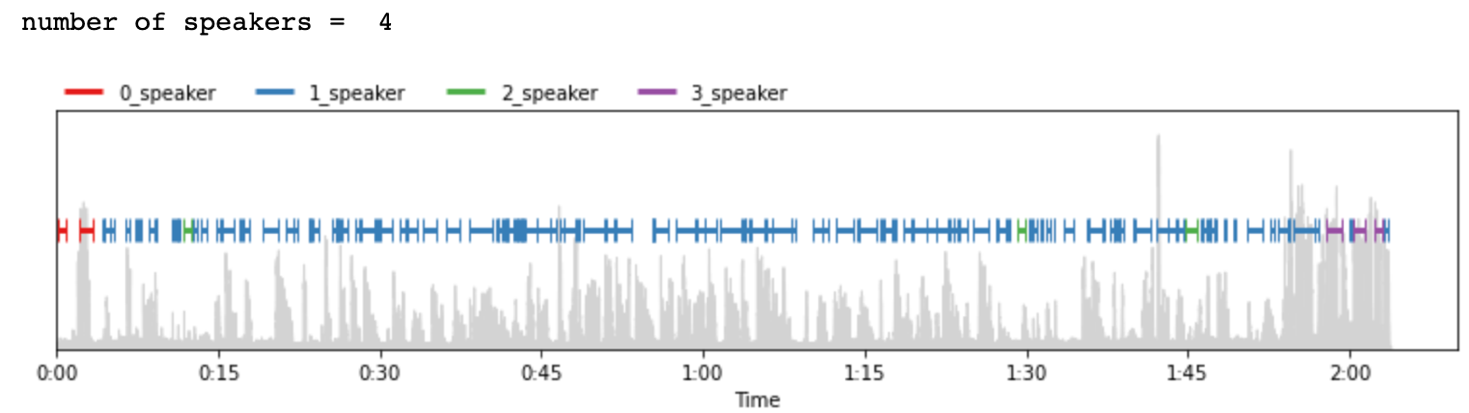
* 1. **Случай трех дикторов**

Данная аудиозапись также представлена на английском языке, однако в беседе участвуют трое. Два из них говорят на протяжении всей аудиозаписи длительностью в 2 минуты, а вот третий говорит всего одну реплику в начале. Протестируем задачу на разных параметрах.

1. Длительность фреймов = 2.5

Наложение = 15%

Threshold = 30%

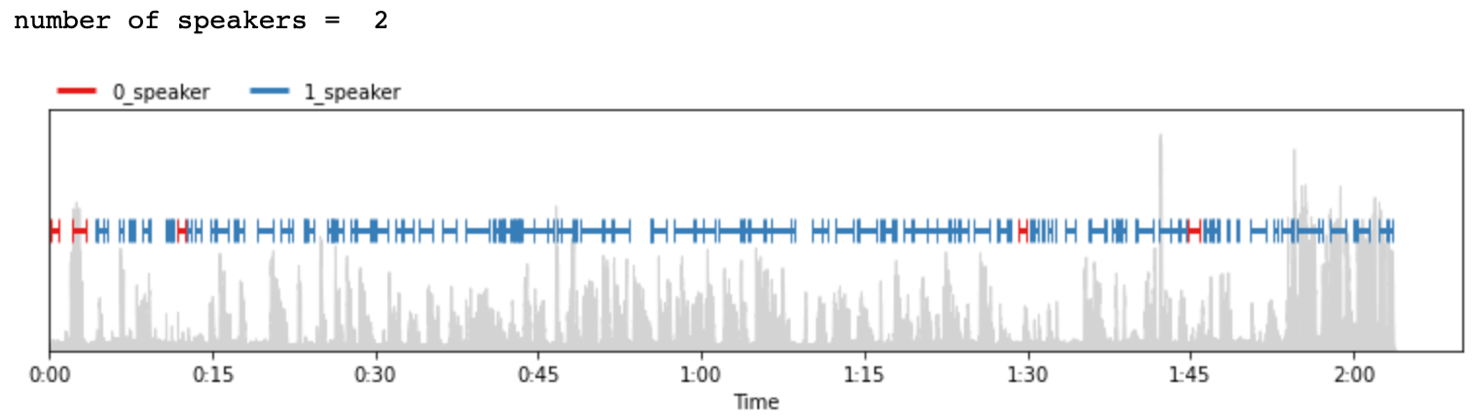


*Рис. 19*

1. Длительность фреймов = 2.5

Наложение = 15%

Threshold = 40%

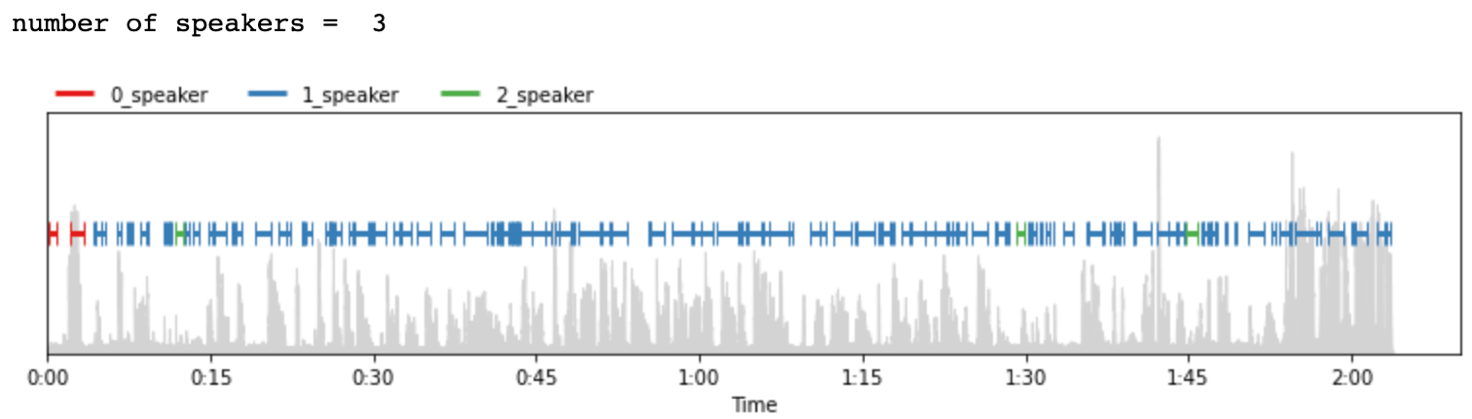


*Рис. 20*

1. Длительность фреймов = 2.5

Наложение = 15%

Threshold = 35%



*Рис. 21*

Как мы видим, на рисунках 19-21 представлена зависимость ответа от показателя threshold. Наиболее близким к ответу оказался случай 3, однако ни один из случаев не определил максимально точно границы смены дикторов.

1. **Выводы**

Таким образом, алгоритм определения количества собеседников действительно актуальный, требующий серьезных знаний не только в сфере программирования, но и в сфере музыки, а именно знаний о голосе, тембре, тонах и многом другом.

Проведенное мною исследование показывает, что задача разрешима даже в рамках курсовой работы, однако содержит в себе определенную погрешность, преодолеть которую возможно с увеличением характеристик голоса, более глубоко анализа и обучения нейронных систем.

Благодаря тому, что программа не привязана к тексту разговора, она может работать для любого иностранного языка, что является ее несомненным преимуществом. Однако, алгоритм сильно зависим от устанавливаемых в коде параметров, таких как: процент наложения одного фрейма на другой, уровень threshold-а, длительность сегментов, на которые делится аудиофайл.

На примерах мы можем видеть, что программа зачастую игнорирует участки с речью, а шумы, напротив, включает в сегменты. Это дает серьезную погрешность и дальнейшие ошибки в определении количества собеседников в аудиозаписи. Тем не менее, данная проблема может быть разрешена посредством совершенствования алгоритма деления фреймов. Например, можно использовать не фиксированную величину разрезания на сегменты, а варьирующуюся в зависимости от частот или фигурирования в записи речи/шумов, а всяческие помехи незамедлительно удалять. Также, программа дает менее качественные результаты для аудиозаписей с большей длительностью, а монолог одного диктора распознается лучше другого, речь которого наименее представлена в аудиозаписи. Несмотря на это, алгоритм дает показатели, которые в целом удовлетворяют поставленным условиям.

В дальнейшем развитии данной работы, необходимо уделить внимание добавлению характеристик, которые решали бы проблему идентификации дикторов, чьи голоса очень похожи и, возможно, принадлежат родственникам. Также, можно усовершенствовать алгоритм шумоподавления и найти оптимальные значения параметров, которые давали бы удовлетворительные результаты для большинства тестируемых данных.

1. **Заключение**

В рамках данной работы была исследована задача диаризации, раскрыта ее актуальность, а также представлены различные подходы к ее решению и описаны значимые характеристики и методы, необходимые для эффективной реализации алгоритма автоматического определения количества собеседников в аудиозаписи.

Программная реализация раскрыта на тестах реальных данных, на основании анализа которых даны выводы о качестве работы кода и предложены пути улучшения и доработки алгоритма.

1. **Список литературы**
2. Б. М. Нутфуллин, Е. А. Ильюшин. Применение синусоидального моделирования речи к задаче диаризации звука. // International Journal of Open Information Technologies ISSN: 2307­8162 vol. 9, no.7, 2021
3. А.В. Судьенкова. Обзор методов извлечения акустических признаков речи в задаче распознавания диктора. // СБОРНИК НАУЧНЫХ ТРУДОВ НГТУ. – 2019. – No 3–4 (96). DOI: 10.17212/2307-6879-2019-3-4-139-164
4. А. Л. Ронжин, В. Ю. Будков. Анализ современных методов и систем диаризации дикторов. // ИЗВ. ВУЗОВ. ПРИБОРОСТРОЕНИЕ. 2012. Т. 55, No 11. УДК 004.896
5. П.А. Меньшаков, И.А. Мурашко. Методика голосовой идентификации на основе нейронных сетей. // ДОКЛАДЫ БГУИР 2017, No 4 (106). УДК 004.822:514
6. О.Ж. Мамырбаев, А.С. Кыдырбекова, Б.Ж. Жумажанов, Д.О. Оралбекова. Распознавание голоса с использованием Х-векторов. // ВЕСТНИК АЛМАТИНСКОГО УНИВЕРСИТЕТА ЭНЕРГЕТИКИ И СВЯЗИ, ISSN 1999-9801