

ЗВУКОВЫЕ ДАННЫЕ И ФУНКЦИИ ПРЕОБРАЗОВАНИЯ ФУРЬЕ, БПФ И СПЕКТРОГРАММ ДЛЯ СИСТЕМЫ РАСПОЗНАВАНИЯ РЕЧИ *Половинченко М.И., Елисеев В.С.*

Ключевые слова: преобразование Фурье, аудиоданные, распознавание речи, спектрограмма.
Аннотация. Почти в каждой организации ежедневно генерируется огромное количество аудиоданных. Аудиоданные дают существенную стратегическую информацию, когда они доступны для специалистов по обработке данных для заправки двигателей искусственного интеллекта и аналитики. В этой статье показано, как работать с аудиоданными и несколькими методами анализа звука с нуля. Кроме того, это дает отправную точку для построения систем распознавания речи.

AUDIO DATA AND FOURIER TRANSFORM, FFT, AND SPECTROGRAM FUNCTIONS FOR SPEECH RECOGNITION *Polovinchenko M.I., Eliseev V.S.*

Keywords: Fourier transform, audio data, speech recognition, spectrogram.

Abstract. Almost every organization generates a huge amount of audio data every day. Audio data provides essential strategic information when it is available to data scientists to fuel the engines of artificial intelligence and analytics. This article shows you how to work with audio data and several audio analysis methods from scratch. In addition, it provides a starting point for building speech recognition systems.

Почти в каждой организации ежедневно генерируется огромное количество аудиоданных. Аудиоданные дают существенную стратегическую информацию, когда они доступны для специалистов по обработке данных для заправки двигателей искусственного интеллекта и аналитики. Организации, которые уже осознали силу и важность информации, поступающей из аудиоданных, используют транскрибированные разговоры искусственного интеллекта для улучшения обучения своих сотрудников, обслуживания клиентов и повышения общего качества обслуживания клиентов.

С другой стороны, есть организации, которые не могут лучше использовать свои аудиоданные благодаря следующим причинам. 1) Они не захватывают его. 2) Качество данных плохое. Эти барьеры могут ограничить потенциал решений машинного обучения, которые они собираются реализовать. Это действительно важно, чтобы захватить все возможные данные, а также в хорошем качестве.

Эта статья содержит пошаговое руководство по началу работы с обработкой аудиоданных. Хотя это поможет вам начать работу с базовым анализом, никогда не будет плохой идеей получить базовое представление о звуковых волнах и основных методах обработки сигналов, прежде чем перейти к этой области. Эта статья дает базовое понимание звуковых волн, а также немного объясняет различные аудиокодеки.

Прежде чем идти дальше, давайте перечислим содержание, которое мы рассмотрим в этой статье. Давайте последовательно рассмотрим каждую из следующих тем:

- чтение аудиофайлов;
- преобразование Фурье (FT);
- быстрое преобразование Фурье (БПФ);
- спектрограмма;
- распознавание речи с помощью функций спектрограммы;
- вывод.

1. Чтение аудиофайлов.

LibROSA – это библиотека python, в которой есть почти все утилиты, которые вам понадобятся при работе с аудиоданными. Эта богатая библиотека предлагает большое количество различных функций. Вот краткий обзор этих особенностей:

- загрузка и отображение характеристик аудиофайла;
- спектральные представления;
- извлечение признаков и манипулирование ими;
- частотно-временные преобразования;
- временная сегментация;
- последовательное моделирование.

Поскольку эта библиотека огромна, мы не будем говорить обо всех функциях, которые она несет. Мы просто собираемся использовать несколько общих черт для нашего понимания.

Таким образом, общий период лечения сокращается, и пациентам легко принять лечение из-за меньшего количества последующих наблюдений. Однако для получения доступа к лучшему файлу в этой группе требуются дальнейшие долгосрочные исследования с более длительными последующими наблюдениями.

Визуализация звука.

Мы получили амплитуды и частоту дискретизации от librosa. Мы можем легко построить график этих амплитуд со временем. LibROSA предоставляет служебную функцию `waveplot()`, как показано на рисунке 1.

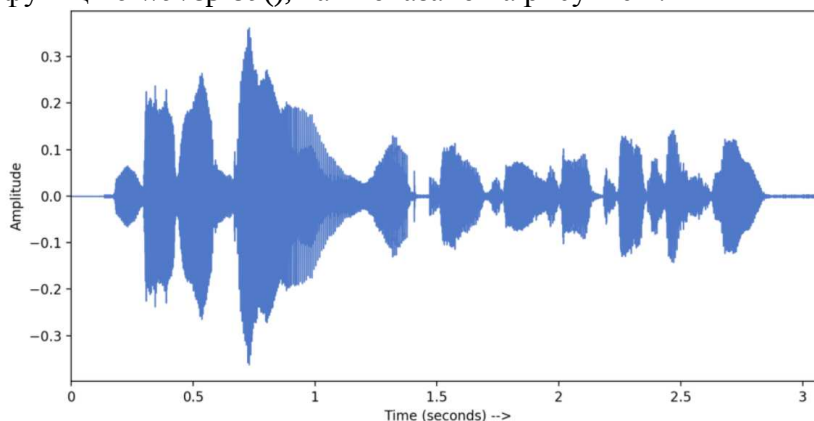


Рис. 1. Визуализация звука

Эта визуализация называется представлением данного сигнала во временной области. Это показывает нам громкость (амплитуду) звуковой волны, изменяющуюся со временем. Здесь амплитуда равна нулю, что представляет собой тишину. (Из определения звуковых волн — Эта амплитуда на самом деле является амплитудой частиц воздуха, которые колеблются из-за изменения давления в атмосфере из-за звука).

2. Преобразование Фурье (FT).

Аудиосигнал — это сложный сигнал, состоящий из множества "одночастотных звуковых волн", которые перемещаются вместе как возмущение в среде [1]. Когда звук записывается, мы фиксируем только результирующие амплитуды этих множественных волн. Преобразование Фурье — это математическое понятие, которое может разложить сигнал на составляющие его частоты. Преобразование Фурье дает не только частоты, присутствующие в сигнале, но и величину каждой частоты, присутствующей в сигнале. Преобразование Фурье представлено на рисунке 2.

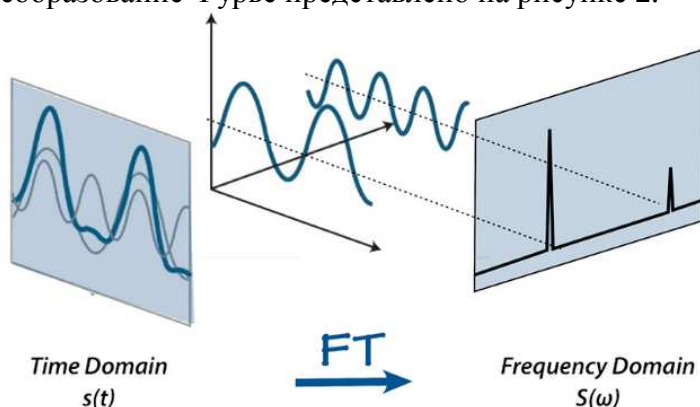


Рис. 2. Преобразование Фурье

Обратное преобразование Фурье прямо противоположно преобразованию Фурье. Он принимает в качестве входных данных частотное представление данного сигнала и математически синтезирует исходный сигнал [2].

3. Быстрое преобразование Фурье (БПФ).

Быстрое преобразование Фурье (FFT) — это математический алгоритм, который вычисляет дискретное преобразование Фурье (DFT) заданной последовательности [3]. Единственное различие между FT и FFT заключается в том, что FT рассматривает непрерывный сигнал, в то время как FFT принимает дискретный сигнал в качестве входного. DFT преобразует последовательность (дискретный сигнал) в ее частотные составляющие точно так же, как FT преобразует непрерывный сигнал. В нашем случае мы имеем последовательность амплитуд, которые были отобраны из непрерывного звукового сигнала. Алгоритм DFT или FFT может преобразовать этот дискретный сигнал во временной области в частотную.

Обзор алгоритма БПФ.

Простая синусоидальная волна для понимания БПФ.

Чтобы понять выход БПФ, давайте создадим простую синусоидальную волну. Следующий фрагмент кода создает синусоидальную волну с частотой дискретизации равной 100, амплитудой равной единице и частотой равной трем. Значения амплитуды вычисляются каждые одну сотую секунды (частота дискретизации) и сохраняются в списке под названием `y1`. Мы передадим эти дискретные значения амплитуды для вычисления ДПФ этого сигнала с помощью алгоритма БПФ [4].

Теперь у нас есть последовательность амплитуд, хранящаяся в списке `y1`. Мы передадим эту последовательность алгоритму БПФ, реализованному `scipy`. Этот алгоритм возвращает список `yf` комплекснозначных амплитуд частот, найденных в сигнале. Первая половина этого списка возвращает положительные частотные термины, а вторая половина возвращает отрицательные частотные термины, которые аналогичны положительным. Вы можете выбрать любую половину и вычислить абсолютные значения, чтобы представить частоты, присутствующие в сигнале. Следующая функция берет образцы в качестве входных данных и строит график частоты.

На следующем рисунке 3 мы построили частоты для нашей синусоидальной волны, используя приведенную выше функцию `fft_plot`. Вы можете видеть, что этот график ясно показывает одночастотное значение, присутствующее в нашей синусоидальной волне, которое равно трем. Кроме того, он показывает амплитуду, связанную с этой частотой, которую мы сохранили равной единице для нашей синусоидальной волны.

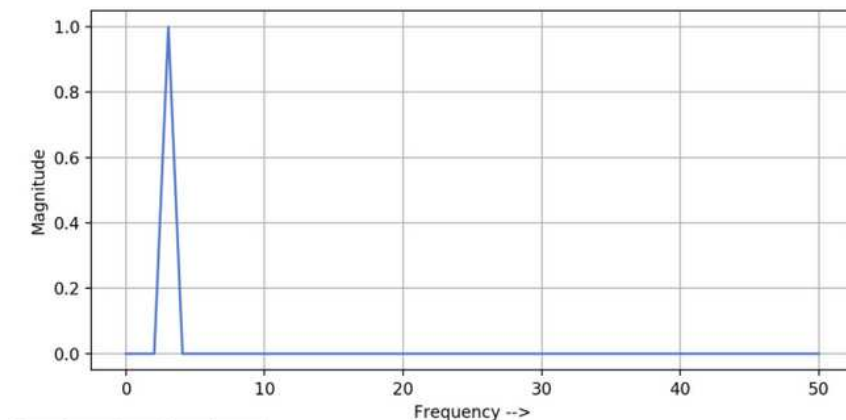


Рис. 3. Частоты синусоидальной волны

Чтобы проверить выход БПФ для сигнала, имеющего более одной частоты, давайте создадим еще одну синусоидальную волну. На этот раз мы будем держать частоту дискретизации равной 100, амплитуду равной двум и значение частоты равное одиннадцати. Сгенерированная синусоидальная волна выглядит следующим образом, как показано на рисунке 4. Это было бы более очевидно и наглядно показано, если бы мы увеличили частоту

дискретизации. Мы сохранили частоту дискретизации равной сто, потому что позже мы добавим этот сигнал к нашей старой синусоиде.

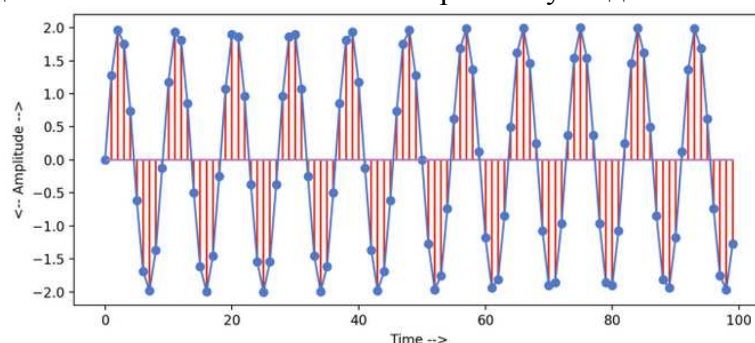


Рис. 4. Сгенерированная синусоидальная волна

Очевидно, что функция БПФ также покажет один всплеск с частотой равной одиннадцати для этой волны. Но мы хотим посмотреть, что произойдет, если мы добавим эти два сигнала с одинаковой частотой дискретизации, но с разными значениями частоты и амплитуды. Здесь последовательность u_3 будет представлять результирующий сигнал.

БПФ на нашем аудиосигнале.

Теперь, когда мы увидели, как этот алгоритм БПФ дает нам все частоты в данном сигнале. давайте попробуем передать наш оригинальный звуковой сигнал в эту функцию. Мы используем тот же самый аудиоклип, который мы загрузили ранее в python с частотой дискретизации равной шестнадцать тысяч.

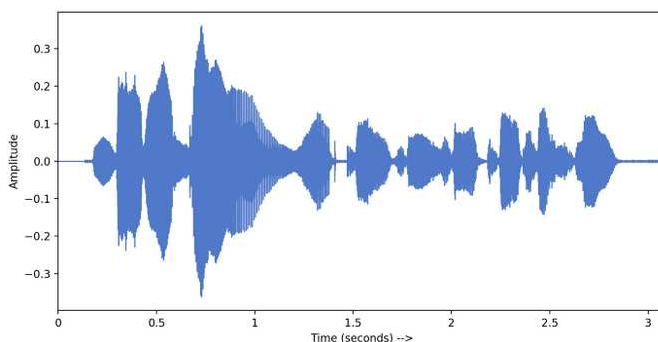


Рис. 5. Частотный график

Теперь взгляните на следующий частотный график (рисунок 5). Этот трех секундный сигнал состоит из тысяч различных частот. Величины значений частоты более двух тысяч очень малы, так как большинство этих частот, вероятно, связано с шумом. Мы строим графики частот в диапазоне от нуля до восьми кГц (рисунок 6), потому что наш сигнал был отобран с частотой дискретизации шестнадцать тысяч и, согласно теореме выборки Найквиста, он должен обладать только частотами меньшими или равными восьми тысячам Гц (Потому что шестнадцать тысяч разделить на два получается восемь тысяч).

Сильные частоты колеблются от нуля до одного кГц только потому, что этот аудиоклип был человеческой речью. Мы знаем, что в типичной человеческой речи этот диапазон частот доминирует.

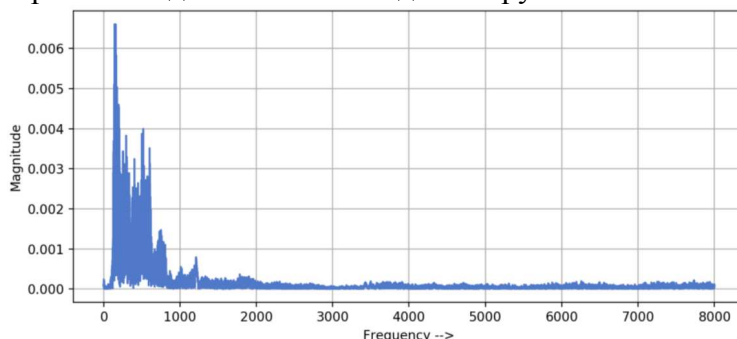


Рис. 6. График частот

4. Спектрограмма.

Визуальное представление частот данного сигнала со временем называется спектрограммой. На графике представления спектрограммы — одна ось представляет время, вторая ось представляет частоты, а цвета представляют величину (амплитуду) наблюдаемой частоты в определенный момент времени [5].

Создание и построение спектрограммы.

Идея состоит в том, чтобы разбить аудиосигнал на более мелкие кадры (окна) и вычислить DFT (или FFT) для каждого окна [6]. Таким образом, мы будем получать частоты для каждого окна, а номер окна будет представлять время. Как окно один приходит первым, окно два следующим, и так далее. Это хорошая практика, чтобы эти окна перекрывались, иначе мы можем потерять несколько частот. Размер окна зависит от проблемы, которую вы решаете.

Для типичной задачи распознавания речи рекомендуется окно длиной от двадцати до тридцати мс. Человек не может говорить больше одной фонемы в этом временном окне. Таким образом, сохраняя окно настолько меньшим, мы не потеряем ни одной фонемы при классификации. Перекрытие кадра (окна) может варьироваться от 25% до 75% в зависимости от ваших потребностей, как правило, оно сохраняется на 50% для распознавания речи.

В нашем расчете спектрограммы мы будем держать продолжительность окна двадцать мс и перекрытие 50% между окнами. Поскольку наш сигнал дискретизируется на частоте шестнадцать тысяч, то каждое окно будет иметь $(16000 * 20 * 0.001) = 320$ амплитуд. Для перекрытия 50% нам нужно пройти вперед на $(320/2) = 160$ значений амплитуды, чтобы добраться до следующего окна. Таким образом, наше значение шага составляет 160.

Взгляните на функцию спектрограммы на рисунке 11. В строке – 18 мы делаем весовое окно и умножаем его на амплитуды, прежде чем передать его в функцию БПФ в строке 20. Окно взвешивания используется здесь для

обработки разрыва этого малого сигнала(малого сигнала из одного кадра) перед передачей его алгоритму DFT.

Выход алгоритма БПФ представляет собой список комплексных чисел ($\text{size} = \text{window_size} / 2$), которые представляют амплитуды различных частот внутри окна. Для нашего окна размером 320 мы получим список из 160 амплитуд частотных ячеек, которые представляют частоты от 0 Гц до 8 кГц (так как наша частота дискретизации составляет шестнадцать тысяч) в нашем случае.

Далее вычисляются и нормируются абсолютные значения этих комплекснозначных амплитуд. Полученная двумерная матрица – это ваша спектрограмма. В этой матрице строки и столбцы представляют номер оконной рамы и частоту, а значения представляют силу частот.

Это сводит его к задаче классификации изображений. Это изображение представляет вашу произнесенную фразу слева направо своевременно. Или рассматривайте это как изображение, где ваша фраза написана слева направо, и все, что вам нужно сделать, это идентифицировать эти скрытые английские символы.

Имея параллельный корпус английского текста, мы можем обучить модель глубокого обучения и построить собственную систему распознавания речи.

6. Заключение.

В этой статье показано, как работать с аудиоданными и несколькими методами анализа звука с нуля. Кроме того, это дает отправную точку для построения систем распознавания речи. Хотя вышеприведенные исследования показывают очень многообещающие результаты для систем распознавания, все же многие не рассматривают распознавание речи как решенную проблему из-за следующих подводных камней:

- модели распознавания речи, представленные исследователями, действительно большие (сложные), что затрудняет их обучение и развертывание;

- эти системы плохо работают, когда говорят несколько человек;

- эти системы плохо работают, когда качество звука плохое.

Они действительно чувствительны к акценту говорящего, поэтому требуют тренировки для каждого отдельного акцента.

В этой области исследований имеются огромные возможности. Улучшения могут быть сделаны с точки зрения подготовки данных (путем создания лучших функций), а также с точки зрения архитектуры модели (путем представления более надежной и масштабируемой архитектуры глубокого обучения).

Список литературы

1. Рассел Джесси. Быстрое преобразование Фурье. – М.: VSD, 2012. – 908 с.
2. Рассел Джесси. Дискретное преобразование Фурье. – М.: VSD, 2013. – 196с.

3. Рассел Джесси. Преобразование Фурье. – М.: VSD, 2012. – 650 с.
4. Vintsiuk Taras K. Generalized Automatic Phonetic Transcribing of Speech Signals // Труды Пятой Всеукраинской международной конференции «Оброблення сигналів і зображень та розпізнавання образів» / УАсОІРО. – Київ, 2000.
5. Young T.S., Evermann G., Kershaw D. and others. The HTK Book – Cambridge University Engineering Department, 2002.
6. Ле Н.В. Распознавание речи на основе искусственных нейронных сетей / Н.В. Ле, Д.П. Панченко // Технические науки в России и за рубежом: материалы междунар. науч. конф. – М., 2011.

References

1. Russell Jesse. Fast Fourier transform. – М.: VSD, 2012. – 908 p.
2. Russell Jesse. Discrete Fourier transform. – М.: VSD, 2013. – 196p.
3. Russell Jesse. Fourier transform. – М.: VSD, 2012. – 650 p.
4. Vintsiuk Taras K. Generalized Automatic Phonetic Transcribing of Speech Signals // Proceedings of the Fifth All-Ukrainian International Conference "Obroblennya signaliv i zobrazhen ta rozpiznavannya obraziv" / UAsO1RO. - Kiev, 2000.
5. Young T.S., Evermann G., Kershaw D. and others. The HTK Book – Cambridge University Engineering Department, 2002.
6. Le N.V. Speech recognition based on artificial neural networks / N.V. Le, D.P. Panchenko // Technical sciences in Russia and abroad: materials of the International Scientific Conference. – Moscow, 2011.

Половинченко Марина Игоревна – магистрант, m.polovin4enko@yandex.ru	Polovinchenko Marina Igorevna – master's student, m.polovin4enko@yandex.ru
Елисеев Владислав Сергеевич – магистрант	Eliseev Vladislav Sergeevich – master's student
Донской государственный технический университет, Ростов-на-Дону, Россия	Don state technical University, Rostov-on-don, Russia

Received 08.05.2021