(Слайд 2) Большой объем аудиоинформации по всему миру позволяет существовать множеству направлений обработки музыки. Одно из них - поиск по фрагменту. Самое известное приложение, решающее эту задачу, - “Shazam”, но можно столкнуться с проблемой сопоставления оригинальной записи и мелодии, напетой или сыгранной на каком-либо инструменте. Отсюда появляется задача создания аудио-поисковой системы. Одно из решений – это распознавание нот. Наиболее технический способ оценки музыки – это гармонический анализ. Так, задача распознавания нот может быть решена комбинацией из гармонического анализа и нейронных сетей за счет знания законов музыки и высоких вычислительных мощностей. С помощью главной мелодии и банка записей нот различных музыкальных инструментов можно заменить звучащий инструмент или голос на необходимый.

(Слайд 3) Таким образом, целью исследования будем рассматривать разработку метода распознавания нот мелодии путем извлечения основных гармоник. При этом необходимо свести обработку звука к работе с изображением.

(Слайд 4) Для исследования необходимо выбрать архитектуру нейронной сети. Обычно в них используются сверточные слои и вариации слоев подвыборки (пулинга). Для устранения переобучения чаще применяется слой Dropout. На выходе - слой softmax, который переводит наборы чисел в вероятности (1), сумма которых равна 1. Итоговому значению соответствует максимальная вероятность.

Для примера рассмотрим сеть LeNet5, схема которой показана на рисунке 1. Она включает 7 слоев, в дополнение к входному слою, 3 из них – сверточные, два – слои подвыборки. На выходе – слой softmax с 10 возможными значениями, соответствующими цифрам от 0 до 9.

(Слайд 5) Не будем опираться на различие октав, поэтому основываемся на 12 основных полутонах. Таким образом, в нашем распоряжении 12 классов.

При многоклассовой классификации для линейных моделей осуществляется переход к набору бинарных классификаций, где целевая переменная кодирует принадлежность к положительному или отрицательному классу.

Регрессия же предсказывает некоторое непрерывное действительное значение. Логистическая регрессия оценивает, к какой категории данных принадлежит конкретная точка из вероятности принадлежности к классу. Поэтому это частный случай линейного классификатора. Для многоклассовой задачи строятся линейные модели (2) по количеству классов, каждая из которых дает оценку принадлежности к конкретному классу. С помощью softmax эти оценки преобразовываются в вероятность путем нормировки вектора оценок. Вероятность -го класса будет тогда вычисляться по формуле (3).

(Слайд 6) Далее будем придерживаться показанной на слайде схемы действий для получения основной гармоники мелодии.

(Слайд 7) Сначала найдем базы данных. Формат данных указан на слайде. Приведение длительности к времени окончания и высоты тона к частоте показано формулами (4) и (5). Частота мелодий стандартно равна 44,1 кГц, то есть 44100 отсчета в секунду. Значит, для перевода времени в отсчеты воспользуемся формулой (6).

(Слайд 8) В рамках проделанной работы рассматривалось 2 набора данных, при этом вторая база также разделялась на два подкаталога английских и корейских песен. Этапы их обработки указаны на слайде.

(Слайд 9) Разбиение аудиофайлов на интервалы делаем в отсчетах. Получившиеся фрагменты распределяем на каталоги, соответствующие нотам, что осуществимо из соответствия нот и частот. Формула частоты для всего звукоряда имеет вид (7). Количество полутонов, на которое заданная частота отстает от камертона, вычисляется по формуле (8). Для сведения всех нот к 12 основным, необходимо по формуле (9) найти значение , из которого определяем ноту по таблице 1.

(Слайд 10) Далее генерируем спектрограммы на основе полученной базы данных.

Спектрограмма — изображение, показывающее зависимость спектральной плотности мощности сигнала от времени. Они применяются в различных областях музыки, радио- и т. д. Спектрограмма - матрица размера , где соответствует определенной частоте, а каждый из столбцов – спектр фрагмента аудиозаписи. Как видно по рисунку 2, чаще спектрограмма показывается как двумерная диаграмма: по горизонтали представлено время, по вертикали — частота; третье измерение с указанием амплитуды на определенной частоте во времени представлено интенсивностью или цветом каждой точки изображения.

Спектрограмму можно получить с помощью оконного преобразования Фурье (STFT), представленного формулой (10). Для упрощения мы предполагаем, что на коротких временных масштабах аудиосигнал статически не сильно меняется. Наиболее популярное окно - окно Хэмминга (11).

(Слайд 11) Вместо спектра самого сигнала можно использовать спектр его мощности. Оценка спектрограммы выполняет работу, аналогичную работе ушной улитки, определяя, какие частоты присутствуют в кадре.

По мере увеличения частот сложнее различить разницу между двумя близкими частотами. Поэтому возьмем скопления блоков спектрограммы и просуммируем их, чтобы получить представление о том, сколько энергии существует в различных частотных областях. Это выполняется с помощью банков фильтров Мела. Шкала мела направлена на то, чтобы имитировать нелинейное восприятие звука человеческим ухом, будучи более различимой на более низких частотах и менее различимой на более высоких. Мы будем использовать банк треугольных фильтров (14) с откликом 1 на центральной части и линейным уменьшением к 0 в центральных частотах двух соседних фильтров. На рисунке 3 показан пример банка треугольных фильтров. От энергии фильтрующего банка берем логарифм, получая кепстр или мел-кепстральные коэффициенты MFCC — т.е. представление сигнала в виде особого спектра, из которого удалены незначительные для человеческого слуха компоненты.

(Слайд 12) В качестве архитектуры сети рассмотрим сеть Филипа Корзениовски, показанную на рисунке 4. Входные данные - спектрограммы. Сеть состоит из 3 скрытых слоев из 512 узлов с функцией активации ReLU и выходного слоя из 12 узлов по количеству полутонов. Параметры нейронной сети показаны на слайде.

(Слайд 13) Для получения спектрограмм используем 2 подхода: применение спектра сигнала и мел-кепстра. Назовем эти модели 1 и 2, соответственно. Данные будем подавать пакетами по 64 вектора.

Первоначальный размер окна для STFT возьмем равным 2048 отсчетов, что соответствует 21,5 Гц – величине, близкой к границе слышимости. Шаг смещения фрейма при этом возьмем равным половине размера кадра. После STFT получаем множество коротких спектров, длина которого вычисляется по формуле (17). Множество таких спектров составит одну спектрограмму. На вход нейросети, как предлагал Корзениовски, подаем 15 последовательных спектрограмм. Для коротких аудио дополним спектрограмму последним столбцом до нужного количества. Таким образом, размерность входа для нейросетевой модели – 1025х15. При получении результатов, номера классов будем сопоставлять с нотой в соответствии с таблицей 1.

(Слайд 14) Вся дальнейшая работа производилась с помощью языка программирования Python. Сначала рассмотрим модель 1. На рисунке 5 (а) представлена спектрограмма, соответствующая некоторой аудиозаписи ноты ре-диез. Ось Х - номера отсчетов, ось Y – номера спектрограмм. На тестовой выборке модель с такими входными данными, обученная на 100 эпохах, дает примерно 95% точности. Но графики штрафа и точности для тренировочной и проверочной выборок показывают наличие переобучения. Тестирование распознавания ноты «ми» было успешно пройдено, что видно по гистограмме 5 (г). Исходя из матрицы ошибок, можно сказать, что в основном модель работает правильно. Время работы модели составило 12 минут 45 секунд.

(Слайд 15) Далее рассмотрим результаты модели 2. На рисунке 6 (а) представлена спектрограмма, соответствующая той же аудиозаписи ноты ре-диез. На ось X - номера спектрограмм, ось Y - номера треугольных фильтров. Точность на тестовой выборке при обучении на 100 эпохах чуть меньше 92%, что ниже результата модели 1. По графикам штрафа и точности для тренировочной и проверочной выборок видно, что и при таком подходе присутствует переобучение, но оно начинается позже, чем у модели 1. При проверке распознавания ноты «ми» из гистограммы 6 (г) заметим, что есть небольшая вероятность спутать эту ноту с ее соседней – нотой «фа». А матрица ошибок показывает, что модель 2 дает больше неверных ответов. Однако, время работы модели составило 3 минуты 25 секунд, что почти в 4 раза меньше, чем для модели 1.

(Слайд 16) Также рассмотрим результаты для демонстрационного примера, который представляет из себя моно-запись короткой гитарной мелодии длительностью 29 секунд. На слайде - графики сравнения реальных нот и нот, полученных в результате распознавания моделями 1 и 2. Для распознавания аудио-сигнал разделялся на равные по времени фрагменты, количество которых равно количеству нот в мелодии (в оригинальной записи расстояния между нотами не равны). Как можно заметить, модель 1 дает более точные результаты, хотя сама точность обоих моделей довольно низкая.

(Слайд 17) Таким образом, можно на данный момент сделать 2 промежуточных вывода: оба подхода дают неплохую точность распознавания нот при обучении, но при демонстрации их точность низкая, и при использовании обоих подходов присутствует переобучение. В дальнейших исследованиях будут проводиться работы в отношении модели 2, показанные на слайде.