KAPITEL 2 / *CHAPTER 2*[2]
AUDIO RECOGNITION PROBLEMS
ПРОБЛЕМЫ РАСПОЗНАВАНИЯ АУДИОСИГНАЛОВ

DOI: 10.30890/2709-2313.2022-08-02-025

Введение

Машинное обучение активно внедряется в разработки компаний уровня Google[1], Facebook[2], Netflix[3] и др. Основополагающими причинами взрывного роста популярности машинного обучения и искусственного интеллекта в сфере обработки информации стали широкий спектр возможностей, впечатляющие результаты и разнообразие вариантов применения в различных предметных областях.

Если рассматривать более узкое направление речевых технологий, в том числе распознавания речи, которое непосредственно связано с темой данной работы, то эта область имеет как богатую историю, так и благоприятные перспективы развития. Широкий спектр применения голосовых систем включает, например, голосовой поиск, голосовое управление, ввод текста, интерфейсы управления умным домом, социальные сервисы для людей с ограниченными возможностями и многое другое. Основное преимущество таких систем заключается в том, что они избавляют конечного пользователя от необходимости использования сенсорных или иных методов ввода данных и команд.

На сегодняшний день задача обработки звуковой информации является актуальной в свете резкого скачка в развитии технологий цифровой обработки сигналов и их распознавания, равно как и совершенствования аппаратных средств, что позволяет обрабатывать большие объемы данных. Многие крупные компании музыкально-технологической и других сфер занимаются исследованиями в данной области, разработкой и совершенствованием новых алгоритмов, технологий и программного обеспечения. Например, компания Google ведет исследования в сфере голосовых технологий, направленные на разработку и совершенствование архитектур и алгоритмов распознавания речи, а также на эксперименты с подходами, ранее считавшимися неоправданно дорогими и ресурсозатратными [4].

Цель данной работы – проектирование и разработка программного модуля компьютерной сетевой системы для распознавания заданного набора голосовых

[2] *Authors: Lvovich I.Y., Lvovich Y. E., Preobrazhenskiy A. P., Preobrazhenskiy Y. P.*

команд с использованием алгоритмов машинного обучения. Задачами являются изучение и сравнительный анализ существующих подходов к решению задач обработки звуковых сигналов, выбор наиболее оптимального для распознавания голосовых команд, разработка программного модуля на основе выбранного подхода и совершенствование разработанного решения для получения максимально возможной точности результатов.

В данной работе применяются следующие общенаучные методы:

- метод формализации для проектирования архитектуры программного модуля;
- эксперимент для проверки предположений о точности результатов работы программы и корректировки параметров;
- дедукция для теоретического обоснования предположений и выводов, полученных эмпирическим путем.

Также можно выделить такие специфичные методы исследования, как алгоритмизация, моделирование и программирование.

1.1. Анализ особенностей распознавания характеристик аудиосигналов

1.1.1. Описание рассматриваемой проблемы

Рассматриваемую в данной работе задачу можно сформулировать следующим образом: реализация системы распознавания голосовых команд (слов) из набора десяти наиболее универсальных голосовых команд на русском языке, которые потенциально могут использоваться во многих системах различной направленности: «старт», «стоп», «пауза», «возобновить», «настройки», «применить», «отменить», «назад», «далее», «выбрать».

В общем смысле распознавание речи можно охарактеризовать как автоматический процесс преобразования речевого сигнала в цифровую информацию. В настоящее время распознавание речи можно свести к трем типам задач: распознавание отдельно произносимых слов, распознавание слитной речи и идентификация по образцу речи. Для каждой из них изучаются и разрабатываются наиболее подходящие методы решения.

1.1.2. Обзор актуального состояния исследований по данной проблеме

На сегодняшний день существуют различные подходы к решению проблемы обработки звуковых сигналов, в частности рассматриваемой в данной работе проблемы распознавания голосовых команд. У истоков

большинства из этих подходов так или иначе стоит спектральный анализ. Если обобщить, спектральный анализ – это совокупность методов качественного и количественного определения состава некоторого объекта, основанная на изучении спектров взаимодействия материи с излучением, включая спектры электромагнитного излучения, акустических волн, распределения по массам и энергиям элементарных частиц и др.

При работе с аудиосигналом, в цифровом виде как правило представляющим собой последовательность амплитудных значений, с помощью, например, преобразования Фурье или вейвлет-преобразования получают частотный спектр этого сигнала, являющийся репрезентативным для последующей обработки или анализа. Спектрограмма звукового сигнала — отражение зависимости спектральной плотности мощности этого сигнала от времени [5].

Также стоит упомянуть о том, что при преобразовании амплитудного спектра сигнала в частотный теряется информация о зависимости частоты от времени. Поэтому для обработки длительных сигналов обычно применяют оконные преобразования, последовательно получая частотный спектр для некоторого малого участка сигнала. Таким образом вместо спектрограммы всего сигнала получают последовательность спектрограмм с зависимостью от временной шкалы.

Одним из самых старых и наиболее полно изученных подходов является анализ сигналов на основе алгоритмов нечеткого поиска. Он позволяет, к примеру, давать числовую (процентную) оценку схожести двух аудиосигналов: входного сигнала и образца, или определять вхождение отрывка в некоторый звуковой сигнал. Для этого классические алгоритмы нечеткого поиска, исторически предназначенные для обработки текстовой информации, можно модифицировать для анализа числовых последовательностей частотного спектра аудиосигнала. Другой путь заключается в преобразовании набора частот в некоторую текстовую последовательность.

Анализ на основе нечеткого поиска является довольно ресурсоемким подходом и имеет ряд функциональных ограничений, связанных, например, с продолжительностью сигналов, темпом речи произносящего распознаваемую голосовую команду или высотой и тембром голоса. Обход этих ограничений в рамках текущего подхода возможен, но является достаточно комплексным и трудоемким с точки зрения программной реализации.

Более современным и имеющим более широкий спектр применений подходом можно назвать использование нейронных сетей. На сегодняшний

день аудиоанализ представляет собой стремительно развивающийся поддомен сферы машинного обучения [6]. Некоторые из самых популярных и распространенных систем машинного обучения, например виртуальные помощники Alexa, Siri и Google Home, — это продукты, созданные на основе моделей, извлекающих информацию из аудиосигналов.

Для данной работы был избран подход, основанный на глубоком обучении, как наиболее универсальный и перспективный.

Одним из наиболее известных фундаментальных трудов, посвященных оконным функциям для гармонического анализа, является работа Фредерика Дж. Харриса 'On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform', изданная в 1978 году и переведенная на русский язык в журнале ТИИЭР [7].

Опираясь в том числе и на этот знаменитый труд, В.П. Дворкович и А.В. Дворкович в 2017 году опубликовали работу «Оконные функции для гармонического анализа сигналов». Она содержит подробную информацию о параметрах классических оконных функций, а также оконных функций, сконструированных в виде произведений, сумм и сверток различных функций, в виде отдельных участков известных окон различными авторами, и их применении для обработки сигналов с использованием быстрого преобразования Фурье [8].

О преобразовании Фурье, его вариациях и применении в решении различного плана задач было издано множество трудов. Одной из наиболее содержательных и полезных для данной работы стала книга Г. Нуссбаумера «Быстрое преобразование Фурье и алгоритмы вычисления сверток», русский перевод которой был издан в 1985 году издательством «Радио и Связь». Помимо рассмотрения разновидностей собственно преобразования Фурье, книга предоставляет теоретический базис по элементам теории чисел и полиномиальной алгебры, алгоритмам быстрой свертки и линейной фильтрации [9].

Одним из лучших алгоритмов по распознаванию и классификации изображений на сегодняшний день считается сверточная нейронная сеть. Использование сверточных нейросетей также показывает свою высокую эффективность в обработке и анализе звуковых (в том числе и речевых) сигналов, основанных на работе с изображениями спектрограмм.

Например, в 2019 вышла статья Мирко Раванелли и Йошуа Бенджио 'Speaker Recognition from Raw Waveform with SincNet', в которой описана end-to-end архитектура нейронной сети для распознавания говорящего по голосу.

Ключевая особенность этой архитектуры — специальные одномерные сверточные слои, которые имеют два параметра с четкой интерпретацией [10].

Еще одной примечательной работой является «Нейросетевой анализ и сопоставление частотно-временных векторов на основе краткосрочного спектрального представления и адаптивного преобразования Эрмита» (авторы Жирков А.О., Корчагин Д.Н., Лукин А.С., Крылов А.С., Баяковский Ю.М.) Работа была издана в 2001 году и рассматривает метод распознавания речи/дикторов на основе представления речевой информации в виде потока двухмерных частотно-временных векторов. Классификация векторов осуществляется нейронной сетью, на вход к которой поступают низкочастотные двумерные вейвлет-преобразования участков спектрограмм. Исходными представлениями звука являются сонограммы краткосрочного преобразования Фурье и адаптивного преобразования Эрмита[11].

Также задача распознавания речи рассматривалась в статье Р.Ю. Белоруцкого и С.В. Житника «Распознавание речи на основе сверточных нейронных сетей», изданной в 2019 году[12].

1.1.3. Анализ исторического опыта

Первое устройство для распознавания речи, способное распознавать произнесенные человеком цифры, появилось в 1952 году [13]. Оно было разработано в Bell Telephone Laboratories и использовало метод распознавания, основанный на сравнении входного сигнала с заранее записанными образцами.

В 1956 году была разработана «фонетическая» печатная машинка, которая могла распознавать десять отдельных символов. Принцип ее работы также базировался на вышеописанном подходе [14].

В 1962 году на ярмарке компьютерных технологий в Нью-Йорке был продемонстрирован компьютер IBM Shoebox, способный выполнять математические операции и распознавать речь. Устройство было разработано Уильямом К. Дершем в лаборатории подразделения Advanced Systems Development Division в IBM. Оно распознало 16 произнесенных слов, в том числе цифры от 0 до 9. Управление компьютером осуществлялось с помощью микрофона, который преобразовывал звуки голоса в электрические импульсы. Измерительный контур классифицировал эти импульсы на основе различных типов звуков и активировал присоединенную вычислительную машину через систему реле [15].

В 60-е годы проводились эксперименты с техниками временной нормализации в попытке минимизировать погрешность распознавания речи

различных людей, а также определять моменты времени, когда речь начинает и заканчивает звучать. В 1966 году Д.Р. Рэдди предпринял попытку разработать систему, способную распознавать непрерывную речь при помощи динамического выделения отдельных фонем [16].

Тремя годами ранее, в 1963, в США были презентованы распознающие устройства «Септрон» (англ. «Sceptron») с оптоволоконным запоминающим устройством, выполняющие некоторую последовательность действий в ответ на произнесенные человеком-оператором определенные фразы [17]. Септроны были пригодны для применения в сфере проводной связи для автоматизации набора номеров, могли применяться в военной сфере для голосового управления сложными образцами военной техники, авиации, автоматизированных системах управления и др.[16-18]

В 1983 году был представлен интерактивный комплекс «умной авионики» для вертолетов «Apache», распознающий голосовые команды пилота, преобразующий их в сигналы управления бортовым оборудованием и отвечающий ему голосом относительно возможности реализации поставленной задачи [18].

В начале 90-х годов начали появляться первые коммерческие программы для распознавания речи, такие как Dragon NaturallySpeaking или VoiceNavigator. С увеличением вычислительных мощностей мобильных устройств стало возможным создавать мобильные программы с функцией распознавания речи, например, Microsoft Voice Command.

Сегодня технологии распознавания речи все больше внедряются во многие сферы человеческой жизни и находят широкое применение в различных сферах бизнеса.

Еще более полувека назад в этой технологии видели большой потенциал. В наши дни на фоне лавинообразного технологического развития, появления новых разработок в сферах программного и аппаратного обеспечения, для нее возникают и новые (в основном коммерческие) области применения.

2.2. Теоретические основы анализа аудиосигналов

2.2.1. Анализ предмета исследования

Обработка звуковой информации неразрывно связана с рядом этапов предобработки обрабатываемых данных. Аналоговый сигнал оцифровывается в некотором цифровом формате, с которым затем следуют произвести ряд

манипуляций, прежде чем он будет пригоден для корректного распознавания.

Рассмотрим последовательно технологии и методы распознавания звуковых сигналов, а также методы предварительно преобразования аудиопотока.

2.2.1.1. Выбор целевого цифрового аудиоформата

Формат аудиофайла определяет структуру и особенности представления звуковых данных при хранении на запоминающем устройстве. Для устранения избыточности аудиоданных используются аудиокодеки, при помощи которых производится сжатие аудиоданных. Выделяют три группы звуковых форматов файлов:

- без сжатия, такие как WAV, AIFF;
- со сжатием без потерь (APE, FLAC);
- со сжатием с потерями (MP3, Ogg).

Рассмотрим несколько популярных аудиоформатов.

WAV – обычно используется для хранения несжатых аудиозаписей (PCM), идентичных по качеству звука записям на компакт-дисках (audio-CD). В среднем одна минута звука в формате wav занимает около 10 мегабайт.

MP3 (MPEG Layer-3) – наиболее распространенный в мире звуковой формат. Как и многие другие форматы с потерей качества, урезает звук, не воспринимаемый человеческим ухом, уменьшая тем самым итоговый размер файла.

WMA (Windows Media Audio) – формат, принадлежащий компании Microsoft. Изначально данный формат был представлен, как замена MP3, имеющая, по заявлению Microsoft, более высокие характеристики сжатия.

AAC – запатентованный аудио-формат, имеющий большие возможности (количество каналов, частоты дискретизации) по сравнению с mp3 и дающий несколько лучшее звучание при том же размере файла.

FLAC – популярный формат сжатия без потерь. Он не вносит изменений в аудиопоток, и закодированный с его помощью звук идентичен оригиналу. Как правило используется для прослушивания звука на звуковых системах высокого уровня.

Самым популярным форматом, используемым для анализа и обработки, считается WAV. Он содержит несжатое аудио в формате линейной импульсно-кодовой модуляции (LPCM). Существует широкий выбор программных решений и библиотек для работы с данным форматом в различных языках программирования, например, в C или Python. Причина его популярности в

удобном для обработки и анализа формате представления данных в файле.

WAV-файл использует стандартную RIFF-структуру, которая группирует содержимое файла из отдельных секций (chunks) – формат выборок аудиоданных, аудиоданные, и т. п. Каждая секция имеет свой отдельный заголовок и отдельные данные секции. Заголовок указывает на тип секции и количество содержащихся в ней байт. Такой принцип организации позволяет программам анализировать только необходимые секции, пропуская остальные, которые не известны или не требуют обработки. Некоторые определенные секции могут иметь в своем составе подсекции (sub-chunks).

Заголовки WAV-файла используют стандартный формат RIFF. Первые 8 байт файла - стандартный заголовок секции RIFF, который имеет ID секции "RIFF" и размер секции, равный размеру файла минус 8 байт, используемых для RIFF-заголовка. Первые 4 байта данных в секции "RIFF" определяют тип ресурса, который можно найти в секции. WAV-файлы всегда используют тип ресурса "WAVE". После типа ресурса (ID "WAVE") идут секции звукового файла, которые определяют аудиосигнал.

Существует много типов секций, заданных для файлов WAV, но большинство WAV-файлов содержат только две из них – секцию формата ("fmt") и секцию данных ("data"). Это именно те секции, которые необходимы для описания формата выборок аудиоданных, и для хранения самих аудиоданных. Хотя официальная спецификация не задает жесткий порядок следования секций, наилучшей практикой считается размещение секции формата перед секцией данных. Многие программы ожидают именно такой порядок секций, и он наиболее разумен для передачи аудиоданных через медленные, последовательные источники наподобие Интернет. Иначе если формат придет после данных, то перед стартом воспроизведения необходимо считать и запомнить все аудиоданные, только после получения формата запускать воспроизведение[19].

Секция данных Wave (Wave Data Chunk) содержит данные цифровых выборок аудиосигнала, которые можно декодировать с использованием формата и метода компрессии, указанных в секции формата Wave (Wave Format Chunk).

WAV-файлы обычно содержат только одну секцию данных, но секций может быть несколько, если они содержатся в секции списка Wave (Wave List Chunk "wavl").

Аудиовыборки многоканального цифрового аудио сохраняются как чередуемые (interlaced) данные, которые просто означают последовательные

аудиовыборки нескольких каналов (таких как стерео и каналы окружения surround). Выборки каналов сохранены последовательно друг за другом, перед тем как произойдет переход к следующему времени выборки. Это сделано с целью возможности последовательного проигрывания файла даже тогда, когда еще не весь файл прочитан целиком.

2.2.1.2. Методы предварительного преобразования аудиопотока

Цифровой звук — это аналоговый звуковой сигнал, представленный посредством дискретных численных значений его амплитуды.

Звуковая волна имеет три основных характеристики: амплитуда, частота и фаза. Частота и фаза являются функциями времени, амплитуда определяет динамический диапазон. Отсюда следует, что для корректного представления звукового сигнала в цифровой форме необходимо сохранить изменения амплитуды как функцию времени[20].

При этом стоит отметить основной проблемой задачи спектрального анализа голосовых команд то, что голоса разных дикторов могут иметь многочисленные отличия. Для выделения этих отличий рассмотрим основные акустические характеристики голосового сигнала, влияющие на подходы к спектральному анализу этого сигнала.

В первую очередь голоса дикторов разного пола и возраста различаются по *высоте*. Базовая частота основного тона – частота вибрации голосовых связок, которая является акустическим коррелятором тона голоса, индивидуальна и определяется не только возрастной и половой принадлежностью говорящего, но и особенностями строения голосовых связок и гортани, а также обуславливается интонациями и эмоциональной окраской речи. В среднем для мужского голоса частота основного тона составляет 80-210 Гц, для женского – 150-320 Гц [21].

Сила (громкость, уровень звукового давления) голоса определяется амплитудой вибрации голосовых связок.

Индивидуальная комбинация амплитуды и частоты связок формирует *вибрато* – периодические изменения высоты, громкости и/или тембра звука. Оно более значимо в контексте анализа певческого голоса, нежели устной речи, и как правило оказывает на спектральные характеристики речевого сигнала совсем незначительное влияние.

Тембр является, пожалуй, наиболее комплексным параметром голоса. Он определяет его индивидуальную окраску путем сложения основного тона и обертонов, сумма звучания которых придает голосу индивидуальную

совокупность высоты и интенсивности голоса, а также шумов, возникающих в момент формирования звука в голосовых связках.

Тембр напрямую связан с такой акустической характеристикой звуков (прежде всего гласных) как *форманта*. В частотном спектре форманта выглядит как отчетливо выделяющаяся область усиленных частот. Это обусловлено тем, что в определенной частотной области вследствие резонанса в голосовом аппарате усиливается некоторое число гармоник тона, производимого голосовыми связками. Для характеристики звуков речи обычно выделяют четыре форманты, пронумерованных в порядке возрастания их частоты. Для разных звуков речи характерны определенные частотные диапазоны формант [21].

Существует подход к распознаванию голосовых сигналов на основе анализа изображений формант, однако он показывает менее эффективные результаты, чем анализ изображений спектрограмм [10].

В упрощенной форме процесс представления звукового сигнала в цифровой форме связан с замером значений амплитуды аналогового сигнала (дискретизация по амплитуде) в определенные и постоянные моменты времени (дискретизация по времени).

Представление звукового сигнала в виде набора отсчетов амплитуд удобно для его хранения и преобразования обратно в аналоговый сигнал. Однако, в силу того что реальный звуковой сигнал складывается из составляющих его частот с определенной амплитудой и фазой, применение многих операций обработки звукового сигнала (например, частотные фильтры) требует преобразования его в частотный спектр.

Спектрограмма — отражение зависимости спектральной плотности мощности сигнала (распределения мощности сигнала, приходящейся на единичный интервал частоты) от времени.

В контексте рассматриваемой задачи анализ входных аудиосигналов производится именно на основе анализа изображений их спектрограмм.

Для получения спектра звукового сигнала наиболее часто используются дискретное преобразование Фурье и вейвлет-преобразование. Интегральное преобразование и ряды Фурье лежат в основе спектрального анализа. Однако несмотря на популярность преобразования Фурье для частотного представления сигнала, существует ряд фундаментальных ограничений, которые привели к появлению оконного преобразования Фурье и стимулировали развитие вейвлет-преобразования.

Для рассматриваемой задачи для предварительной обработки входных

сигналов было решено применять оконное преобразование Фурье.

2.2.1.3. Выбор оконной функции для оконного преобразования Фурье на этапе формирования спектрограммы

Преобразование Фурье раскрывает элементарную периодичность сигнала, раскладывая сигнал на составляющие его синусоидальные частоты и определяя величины и фазы этих составляющих частот.

Преобразование Фурье функции f вещественной переменной является интегральным и задаётся следующей формулой:

$$\hat{f}(\alpha) = \int_{\mathbb{R}} f(x) e^{-i\alpha x} dx$$

При получении частотного спектра из цифрового (или оцифрованного аналогового) аудиопотока теряется информация о временных характеристиках для той или иной частоты. Простыми словами, получив частотный спектр звукового сигнала, невозможно определить, в какой именно момент времени на протяжении длительности данного сигнала прозвучала та или иная частота. Например, преобразование Фурье не различает сигнал, представляющий собой сумму двух синусоид с различными частотами, от сигнала, состоящего из тех же синусоид, но звучащих последовательно одна за другой [22].

В контексте распознавания звуковых сигналов эта проблема может быть решена применением оконного преобразования Фурье. Временной интервал сигнала разделяется на подынтервалы, и преобразование Фурье выполняется последовательно на каждом из этих интервалов. Таким образом осуществляется переход к частотно-временному представлению сигналов. Результатом оконного преобразования является семейство спектров, которым отображается изменение спектра сигнала по интервалам сдвига окна преобразования.

Оконное преобразование Фурье определяется следующим образом:

$$F(t, \omega) = \int_{-\infty}^{\infty} f(\tau) W(\tau - t) e^{-i\omega\tau} d\tau,$$

где $W(\tau - t)$ – некоторая оконная функция.

На практике нет возможности получить сигнал на бесконечном интервале, так как нет возможности узнать, какой был сигнал до включения устройства и какой он будет в будущем. Ограничение интервала анализа равносильно произведению исходного сигнала на прямоугольную оконную функцию. Таким образом, результатом оконного преобразования Фурье является не спектр исходного сигнала, а спектр произведения сигнала и оконной функции. В результате возникает эффект, называемый растеканием спектра сигнала. Существует проблема, которая заключается в том, что боковые лепестки

сигнала более высокой амплитуды могут маскировать присутствие других сигналов меньшей амплитуды.

Для борьбы с растеканием спектра применяют гладкую оконную функцию, спектр которой имеет более широкий главный лепесток и низкий уровень боковых лепестков. Спектр, полученный при помощи оконного преобразования Фурье, является сверткой спектра исходного идеального сигнала и спектра оконной функции [23].

2.2.1.4. Генерация спектрограммы

Преобразование Фурье позволяет получить частотный спектр из массива амплитуд, которым по сути является входной аудиосигнал.[6] Для этого к каждому окну Гаусса, полученному из входного аудиопотока, применяется быстрое преобразование Фурье, после чего для каждого полученного набора частот строится спектрограмма. Эти изображения объединяются в одно, составляя визуализацию зависимости частотных характеристик сигнала от времени.

2.2.1.5. Предварительная обработка входных данных для распознавания

Для минимизации ошибки распознавания необходимо осуществить некоторую предварительную обработку входных данных:

- ограничение сигнала по временной шкале с целью выделения его информативной части;

- ограничение частотного диапазона до 8 кГц, что является достаточным для качественного распознавания, но не перегружает вычислительные мощности компьютера [24];

- нормировка интенсивности спектра для корректной обработки сигналов различной громкости;

- уменьшение разрешения изображений спектрограмм с целью снизить нагрузку на вычислительные мощности и повышение их контрастности для выделения информативных составляющих сигнала на фоне шумов, так как запись образцов речи производится с использованием непрофессионального оборудования.

Произведя данные манипуляции с тестовым набором данных, можно будет приступать к обучению нейронной сети. Впоследствии при использовании программного модуля входные аудиосигналы будут проходить через те же стадии предварительной обработки.

2.2.1.6. Обзор теоретических основ архитектуры нейронных сетей

Искусственная нейронная сеть – математическая модель, а также ее программная или аппаратная реализация, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма.

Искусственная нейронная сеть представляет собой систему соединенных и взаимодействующих между собой элементарных процессоров – искусственных нейронов.

Возможность обучения за счет применения решений множества сходных задач – одно из главных преимуществ нейросетей перед традиционными алгоритмами. Технически обучение заключается в нахождении коэффициентов связей между нейронами.

Искусственный нейрон (*математический нейрон Маккалока-Питтса*) – узел искусственной нейронной сети, являющийся упрощенной моделью естественного нейрона.

Обычно искусственный нейрон представляется как некоторая нелинейная функция от линейной комбинации всех входных сигналов. Данную функцию называют *функцией активации* или *функцией срабатывания*, *передаточной функцией*.

Каждая связь, по которой выходной сигнал одного нейрона поступает на вход другого, характеризуется своим весом. Связи с положительным весом называются возбуждающими, с отрицательным – тормозящими. Нейрон имеет один выход, сигнал с которого может поступать на произвольное число выходов других нейронов. Функция активации определяет зависимость сигнала на выходе нейрона от взвешенной суммы сигналов на его входах.

Стоит отдельно упомянуть основные параметры нейронной сети, чтобы иметь возможность в дальнейшем оперировать этим понятийным аппаратом.

Эпоха – этап работы программы, когда весь датасет прошел через нейронную сеть в прямом и обратном направлении один раз. Одна эпоха приводит к недообучению, избыток – к переобучению. С увеличением числа эпох веса сети изменяются все большее количество раз. При переобучении исчезает обобщающая способность нейронной сети. Для различных датасетов оптимальное количество эпох будет отличаться, оно напрямую связано с разнообразием данных.

Поскольку одна эпоха как правило слишком велика для аппаратного обеспечения, датасет делится на маленькие партии – *батчи*. Размер батча – общее число тренировочных данных, представленных в одном батче.

Итерации – число батчей, необходимое для завершения одной эпохи.

Можно выделить восемь общих этапов решения задач с помощью нейронных сетей:

- сбор данных для обучения;
- подготовка и нормализация данных;
- выбор топологии сети;
- экспериментальный подбор характеристик сети: числа эпох, итераций и батчей;
- экспериментальный подбор параметров обучения;
- обучение;
- проверка адекватности обучения;
- корректировка параметров, окончательное обучение.

Нейронные сети можно классифицировать по различным признакам.

Характер обучения

Можно выделить три характера обучения нейросетей:

- *обучение с учителем*: выходное пространство решений нейронной сети известно;
- *обучение без учителя*: сеть формирует выходное пространство решений только на основе входных воздействий. Такие сети называют самоорганизующимися;
- *обучение с подкреплением*: система назначения штрафов и поощрений от среды.

Архитектура

Также существует классификация нейросетей по архитектуре.

Многослойными (*многосвязными*) называются нейросети, в которых нейроны сгруппированы в слои. Каждый нейрон предыдущего слоя связан со всеми нейронами следующего слоя. Внутри слоев связь между нейронами отсутствует. Слой содержит совокупность нейронов с едиными входными сигналами.

Среди многослойных сетей в свою очередь выделяют:

- сети без обратных связей (прямого распределения): сигналы передаются строго по направлению от входного слоя к выходному;
- сети с обратными связями: информация может передаваться с последующих слоев на предыдущие.

Слабосвязные сети (*сети с локальными связями*) – слоистые сети с небольшим количеством связей.

В *полносвязных* сетях каждый нейрон передает свой выходной сигнал всем

остальным нейронам сети, в том числе самому себе.

Топология

Существует множество разновидностей топологий нейронных сетей, применяемых в силу своих особенностей для решения разных классов задач. Автокодировщик (разновидность сети прямого распространения) используется для кодирования информации, сверточные сети – для распознавания и классификации образов, разверточные сети – для генерации выходных данных по заданному критерию, сети типа deep belief – для отображения данных в виде вероятностной модели и др.

Для обработки изображений и аудио чаще всего используются сверточные нейронные сети и глубинные сверточные нейронные сети. Зачастую они применяются для классификации изображений и показывают один из лучших результатов в данной области [25].

2.2.1.7. Обзор принципов работы сверточных нейронных сетей

Сверточные нейронные сети работают на основе фильтров, которые предназначены для определения некоторых конкретных характеристик изображения (например, прямых линий). *Фильтр* — это набор кернелов (ядер); иногда в фильтре используется одно ядро. *Ядро* — это матрица чисел, называемых весами, которые “обучаются” (подстраиваются) с целью поиска на изображении определенных характеристик. Фильтр перемещается вдоль изображения и определяет, присутствует ли некоторая искомая характеристика в конкретной его части.

Если некоторая искомая характеристика присутствует во фрагменте изображения, операция свертки на выходе будет выдавать число с относительно большим значением. Если же характеристика отсутствует, выходное число будет маленьким. Результатом перемещения данного фильтра вдоль всего изображения есть матрица, состоящая из результатов единичных сверток.

Для того, чтобы обучение весов было эффективным, в результаты сверток вводятся некоторое смещение (bias) и нелинейность.

Смещение — это операция сложения каждого элемента выходной матрицы с величиной смещения. Это может быть необходимо для того, чтобы вывести нейронную сеть из тупиковых ситуаций, имеющих сугубо математические причины.

Нелинейность представляет из себя функцию активации. Благодаря ей

картина, формируемая с помощью операции свертки, получает некоторое искажение, позволяющее нейронной сети более ясно оценивать ситуацию. Обычно в сверточных слоях используется более одного фильтра. Когда это имеет место, результаты работы каждого из фильтров собираются вдоль некоторой оси, что в результате дает трехмерную матрицу выходных данных. С целью ускорения процесса обучения и уменьшения потребления вычислительных ресурсов производят даунсемплинг исходных и/или промежуточных данных [26].

Далее, после нескольких сверточных слоев и блока даунсемплинга, трехмерное представление изображения разворачивается в вектор, который далее будет передан в многослойный перцептрон — полносвязную нейронную сеть.

Выходной слой отвечает за формирование вероятностей принадлежности входного образа тому или иному классу. Для этого выходной слой должен содержать количество нейронов, соответствующих количеству классов. Взвешенные и просуммированные сигналы далее модифицируются с помощью функции активации [10].

2.3. Практическая реализация разработанных алгоритмов

2.3.1. Проектирование и разработка программного модуля

Программный модуль для распознавания голосовых команд предполагается содержащим несколько функциональных компонентов, которые будут последовательно передавать обрабатываемые данные из одного компонента в другой. Работа модуля будет начинаться с запуска компонента записи и сохранения аудиосигнала, затем этот сигнал будет передаваться компоненту извлечения из сигнала последовательности данных и обработки их для получения частотных характеристик сигнала и построения спектрограммы. Далее полученное изображение спектрограммы передается компоненту обработки изображений для сжатия, увеличения контрастности и нормировки гистограммы. После этапа обработки изображение поступает на вход компоненту распознавания, содержащему обученную сверточную нейронную сеть. На выходе этого компонента пользователь получает результаты распознавания входного сигнала.

Также в модуль заложена возможность расширения заложенного в него набора распознаваемых команд. Для этого пользователь должен добавить в

датасеты для обучения, проверки и тестирования фиксированный набор реализаций (записей) новой команды, а затем инициировать повторную генерацию и обучение распознающей нейросети на обновленных данных с корректировкой ее параметров, таких как количество сверточных слоев или количество нейронов на выходном слое сети. Наряду с добавлением существует возможность замены и удаления команд, которое осуществляется по тому же алгоритму.

Доступ к этой функциональности должна иметь только определенная группа пользователей с расширенным набором разрешений (назовем их «администраторами»). Реализация различных уровней доступа является второстепенной в контексте рассматриваемой в данной работе задачи, поэтому подробности программирования этого сегмента программы здесь не описываются.

Также стоит отметить, что, предоставляя доступ к редактированию датасетов, разработчик не может гарантировать корректность добавляемых администратором данных. Некорректные данные (например, реализации различных команд в одном наборе) могут негативно повлиять на качество обучения нейросети независимо от ее архитектуры и параметров. Эта уязвимость не является критичной в контексте разработки опытного образца, однако в случае интеграции программного модуля в какую-либо систему с реальными пользователями необходима верификация данных, помещаемых в обучающие датасеты.

Разработка программной реализации модуля предобработки и распознавания аудиосигналов осуществляется на языке программирования Python с использованием библиотек **pyaudio** для получения сигналов с микрофона, встроенного или подключенного к компьютеру, на котором запущена программа, **wave** для преобразования записанного звука в формат wav и разбора аудиофайлов, **scipy** для осуществления оконного преобразования Фурье, **matplotlib** для генерации спектрограмм, **pillow** для обработки изображений спектрограмм и **keras** для синтеза нейронной сети.

Несмотря на то, что домен администрирования подробно не рассматривается в рамках данной работы, его репрезентация важна для корректного представления об общей архитектуре и функциональности программы, а также возможностях ее интеграции в реальные системы в дальнейшем.

Рассмотрим последовательно каждый из функциональных компонентов разрабатываемой программы.

2.3.1.1. Запись звука

Библиотека **pyaudio** позволяет записать звук из любого доступного источника, например микрофона или микшера. Источник может быть как встроен в компьютер или ноутбук, на котором работает программа, так и быть подключенным к нему либо находиться с ним в одной локальной сети. Данные записываются в виде потока объектов типа bytes и могут затем быть сохранены в виде wav-файла. Перед записью инициализируется поток pyAudio, затем в цикле получаемые из потока данные блоками записываются в массив, откуда после окончания записи сохраняются в виде wav-файла.

При необходимости пользователь может отменить сохранение записанного образца и перезаписать его. В этом случае на следующий этап обработки будет передан последний записанный пользователем сигнал, а данные, записанные во время предыдущей попытки, будут автоматически очищаться в момент начала новой записи.

В результате мы получим одноканальный wav-файл с записанным в него сигналом (голосовой командой), готовым к последующей обработке. Таким образом записывались датасеты команд для обучения, проверки и тестирования распознающей нейросети. Таким же образом будут записываться команды при непосредственном использовании программного модуля.

2.3.1.2. Разбор аудиофайла

Для построения спектрограммы необходимо сначала получить последовательность амплитудных отсчетов из обрабатываемого файла, затем применить к ней оконное преобразование Фурье, получив, таким образом, набор частотных спектров для последовательности участков исходного сигнала. На основе этих спектров будут генерироваться спектрограммы.

В результате получаем массив амплитудных отсчетов обрабатываемого сигнала.

2.3.1.3. Генерация спектрограммы

Для генерации изображений спектрограмм используется библиотека matplotlib. В ее методе построения спектрограммы инкапсулировано оконное преобразование Фурье с различными оконными функциями, что позволяет не программировать вычисления вручную.

2.3.1.4. Обработка изображения спектрограммы

Для того, чтобы минимизировать вероятность ошибки распознавания и

уменьшить нагрузку на вычислительные мощности компьютера, перед передачей изображения спектрограммы на вход распознающей нейронной сети необходимо произвести некоторые манипуляции с этим изображением. Для этого в данной работе используется библиотека pillow, предоставляющая богатый инструментарий для обработки изображений средствами языка Python.

Разрешение исходного изображения спектрограммы уменьшается до 60x60 пикселей. При этом на первом шаге изображение сжимается до 60 пикселей по высоте, соответствующей оси частотных характеристик сигнала, без изменения отношения сторон для сохранения пропорциональности репрезентации частотного диапазона сигнала. Далее высота изображения фиксируется, а ширина, отражающая временную шкалу сигнала, уменьшается (в редких случаях увеличивается) до 60 пикселей. Такой двухступенчатый процесс сжатия изображения позволяет свести к минимуму ошибку распознавнаия, связанную с различиями в темпе (длительности) произносимых команд.

Контрастность сжатого изображения повышается на 60% для более четкого выделения информативной части спектра, цветность опускается до черно-белого спектра, производится процедура эквализации гистограммы для нормировки яркости изображений. Последнее важно для минимизации ошибки распознавания из-за различий в громкости записываемых сигналов.

Гистограмма изображения – это диаграмма распределения пикселей с различной яркостью. Горизонтальная ось диаграммы отражает яркость, а вертикальная – количество точек с конкретным значением яркости. Если посмотреть на гистограмму одного и того же черно-белого фото с различными параметрами яркости, можно заметить, что чем темнее изображение, тем больше ненулевые значения гистограммы концентрируются в левой части гистограммы (ближе к минимальным уровням яркости).

2.3.2. Обучение нейронной сети. Корректировка параметров сети

Произведем обучение спроектированной нейронной сети в 30 эпох. Так как объем обучающих и тестовых данных сравнительно небольшой, разбивать их на батчи нет смысла.

Результаты распознавания проверочных данных после обучения составили 78.3%. Этот результат можно считать близким к удовлетворительному, однако он ниже среднего показателя распознавания, которого достигают сети подобной конфигурации.

Далее были проведены тесты с последовательным наращиванием количества сверточных слоев и эпох обучения. Результаты распознавания

конфигураций сетей с двумя, тремя и четырьмя сверточными слоями при обучении в 30, 40, 50 и 60 эпох представлены в таблице 1.

Таблица 1 – сравнение результатов распознавания для сетей разной конфигурации и с разным количеством эпох обучения

Кол-во эпох / Кол-во Слоев	30	40	50	60
2	88%	89,4%	91,1%	90%
3	91,6%	93,2%	97,2%	93,8%
4	88,7%	91%	94,5%	90,7%

На основании полученных данных можно сделать вывод, что наиболее оптимальной конфигурацией нейронной сети является сеть с тремя сверточными слоями, обученная на протяжении 50 эпох.

Также эти цифры наглядно подтверждают озвученные ранее факты о работе нейронных сетей:

- увеличение количества эпох обучения повышает качество распознавания;
- излишнее количество эпох приводит к переобучению;
- введение дополнительных сверточных слоев также ведет к повышению качества распознавания, но излишнее количество слоев ведет к переобучению.

Выводы

В рамках данной работы удалось спроектировать и разработать программный модуль для сетевой информационной системы, способный распознавать десять голосовых команд на русском языке с точностью более 97% в тестовой выборке. Этот модуль разработан на языке программирования Python и пригоден для использования в системах различной конфигурации в качестве самостоятельного приложения или компонента какого-либо комплексного программного продукта. Программа является машстабируемой, имеется возможность при необходимости менять конфигурацию нейронной сети, а также размер и состав обучающих датасетов.

На подготовительных этапах работы были изучены и проанализированы теоретические материалы по обработке звуковых сигналов, основам

спектрального анализа, машинного обучения, принципам построения и функционирования нейронных сетей. Рассмотрен исторический опыт в контексте проблемы распознавания звуковых сигналов, краткая справка по истории исследований и разработок в данной области.

Был разработан комплексный подход к предварительной обработке изображений спектрограмм голосовых команд, позволяющий свести к минимуму влияние на результаты распознавания особенностей голоса диктора, таких как основной тон, громкость и тембр.

Также в ходе разработки и тестирования программы удалось подтвердить рассмотренные на этапе проектирования архитектуры программного модуля факты о работе сверточных нейронных сетей.