

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

РАБОТА ПРОВЕРЕНА

Рецензент

_____ 2022 г.
«__»_____

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой ЭВМ

_____ Д.В. Топольский

«__»_____ 2022 г.

Применение компьютерных средств распознавания речи
при дистанционном обучении иностранному языку

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К МАГИСТЕРСКОЙ ДИССЕРТАЦИИ
ЮУРГУ-090401.2022.211 ПЗ МД

Руководитель работы,

к.т.н., доцент каф. ЭВМ

_____ И.Л. Кафтаников

«__»_____ 2022 г.

Автор работы,

студент группы КЭ-222

_____ Г.А. Рыкунов

«__»_____ 2022 г.

Нормоконтролёр, старший

преподаватель каф. ЭВМ

_____ С.В. Сяськов

«__»_____ 2022 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра «Электронные вычислительные машины»

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

_____ Д.В. Топольский

«___» _____ 2022 г.

ЗАДАНИЕ

на выполнение магистерской диссертации

студенту группы КЭ-222

Рыкунову Георгию Андреевичу

обучающемуся по направлению

09.04.01 «Информатика и вычислительная техника»

1. **Тема работы:** «Применение компьютерных средств распознавания речи при дистанционном обучении иностранному языку» утверждена приказом по университету от 25.04.2022 № 697-13/12(приложение №73)
2. **Срок сдачи студентом законченной работы:** 8 июня 2022 г.
3. **Исходные данные к работе:**
 - техническое задание;
 - язык программирования JavaScript;
 - фреймворк Vue.JS;
 - серверная платформа Node.JS;
 - система распознавания речи Web Speech Api.

4. Перечень подлежащих разработке вопросов:

- введение и обзор литературы;
- изучение особенностей и распознавания речи;
- проведение сравнительного анализа аналогов системы;
- проектирование прототипа обучающей системы;
- разработка прототипа обучающей системы;
- тестирование обучающей системы.

5. Дата выдачи задания: 1 декабря 2021 г.

Руководитель работы _____ /И.Л. Кафтанников/

Студент _____ /Г.А. Рыкунов/

КАЛЕНДАРНЫЙ ПЛАН

Этап	Срок сдачи	Подпись руководителя
Введение и обзор литературы	1.03.2022	
Изучение существующих методов распознавания речи	1.04.2022	
Обработка и агрегирование данных	25.04.2022	
Проектирование прототипа обучающей системы	1.05.2022	
Разработка прототипа обучающей системы	10.05.2022	
Компоновка текста работы и сдача на нормоконтроль	31.05.2022	
Подготовка презентации и доклада	7.06.2022	

Руководитель работы _____ /И.Л. Кафтанников/

Студент _____ /Г.А. Рыкунов

Аннотация

Г.А. Рыкунов. Применение компьютерных средств распознавания речи при дистанционном обучении иностранному языку. – Челябинск: ФГАОУ ВО «ЮУрГУ (НИУ)», ВШ ЭКН; 2022, 47 с., 11 ил., библиогр. список – 10 наим.

Работа посвящена разработке прототипа системы распознавания речи для обучения иностранным языкам, в работе описывается применение системы для обучения английскому языку. Описаны основные принципы распознавания речи. Для проектирования и разработки прототипа использован язык JavaScript и его фреймворк Vue.JS, также использована серверная платформа Node.JS.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	7
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	10
1.1. Распознавание образов.....	10
1.2. Постановка задачи распознавания речи.....	12
1.3. Обзор аналогов.....	13
Результаты и выводы по первой главе.....	16
2. ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ.....	17
2.1. Функциональные требования.....	17
1.1.1. Реализуемые задачи.....	17
1.1.2. Требования к ПО.....	18
2.2. Нефункциональные требования.....	18
2.2.1. Условия эксплуатации.....	18
2.2.2. Пользователи.....	18
2.2.3. Характеристики сервера.....	19
2.2.4. Характеристики клиента.....	19
Результаты и выводы по второй главе.....	20
3. ПРОЕКТИРОВАНИЕ.....	21
3.1. Варианты использования системы.....	21
3.2. Архитектура компонентов.....	23
3.3. Проектирование алгоритма сравнения и анализа текста.....	23
Результаты и выводы по третьей главе.....	25
4. РЕАЛИЗАЦИЯ.....	26
4.1. Общее описание принципов реализации.....	26
4.2. Реализация алгоритма анализа текста.....	27
4.4. Реализация клиента и сервера.....	28
4.5. Реализация модулей системы.....	30
Основные результаты и выводы по четвертой главе.....	34
5. ТЕСТИРОВАНИЕ.....	35
5.1. Описание методов тестирования.....	35
5.2. Анализ данных предоставленных в результате тестирования.....	35
5.3. Список предложенных доработок системы.....	37
Результаты и выводы по пятой главе.....	38
6. ПРИМЕНЕНИЕ.....	39
6.1. Инструкция по применению прототипа системы.....	39
ЗАКЛЮЧЕНИЕ.....	41
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	42
ПРИЛОЖЕНИЕ.....	44

ВВЕДЕНИЕ

Актуальность проблемы и темы исследования обоснована тем, что с каждым годом в области преподавания иностранного языка появляется все больше разновидностей методов и форм дистанционного обучения с применением новых информационных технологий. Они представляют собой не только отдельные технические средства или системы обмена и передачи информации, с помощью которых осуществляется учебный процесс, но и целостную систему методов обучения, направленных на развитие коммуникативных компетенций обучающихся, развитие навыков речи.

Применение средств распознавания речи может стать неотъемлемой частью дистанционного обучения иностранному языку в будущем и снизить нагрузку на преподавателей иностранных языков

Грамотное применение современных информационных технологий в обучении иностранному языку позволит повысить продуктивность образования, увеличить качество получения информации и упростить ее усвоение, преподаватель в профессиональной деятельности сталкивается с проблемой внедрения современных эффективных технологий, поэтому возникает необходимость нахождения оптимальных способов использования данных технологий в практике преподавания.

В настоящее время не существует универсального инструмента для преподавателя, в котором бы использовались технологии распознавания речи и который мог бы использоваться в учебных заведениях наряду с привычными методами обучения.

Проблема исследования – изучение возможностей применения компьютерных технологий распознавания речи в обучении иностранному языку.

Цель исследования – изучить особенности применения технологий распознавания речи и углубиться в их применение в обучении иностранному языку

В соответствии с целью исследования в ходе исследования решались следующие задачи:

- 1) Раскрыть сущность понятий “распознавание образов” и “распознавание речи”;
- 2) Рассмотреть практики применения технологий распознавания речи в обучении иностранному языку;
- 3) Спроектировать систему, использующую распознавание речи в обучении иностранному языку;
- 4) Разработать рабочий прототип спроектированной системы;
- 5) Оценить и интерпретировать результаты опытно-экспериментальной работы.

Теоретико-методологической основой исследования являются:

педагогические и методические разработки таких авторов как: (В.П. Беспалько, Е.С. Полат, Н.Д. Гальсковой, Е.И. Машбец., Назарова Н.Б., Мохова О.Л.); Повышение эффективности и качества обучения продуктивной и рецептивной лексике английского языка в средней школе через использование новых средств обучения в урочной и внеурочной деятельности (В.Г.Адмони, И.Л.Бим, Н.И.Гез, О.Д.Митрофанова, В.Г.Костомаров, А.К.Пепеляев, Р.Г.Рогова.)

Структура работы:

Работа состоит из введения, шести глав, заключения, списка использованной источников, приложений.

Во введении представлена актуальность темы исследования, формулируется проблема, цель и задачи исследования, определяются его объект и предмет, описывается практическая значимость исследования.

В первой главе исследования содержится анализ предметной области исследования, рассмотрены основные категории и понятия связанные с

распознаванием речи и практика применения технологий распознавания речи в обучении иностранному языку.

Во второй главе исследования сформулированы функциональные и нефункциональные требования к разработке прототипа обучающей системы, также описаны требования необходимые для применения системы.

В третьей главе описан процесс проектирования обучающей системы, ее функциональных компонентов, а также алгоритма анализа и сравнения текста.

В четвертой главе исследования описывается детализированный процесс разработки прототипа системы и ее модулей.

В пятой главе приводятся результаты тестирования системы.

Шестая глава представляет собой инструкцию по применению системы.

Список использованных источников включает в себя 10 источников. Работа содержит 1 приложение.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

В современном мире все большее количество процессов переходят в цифровой вид, электронный документооборот, цифровые паспорта и другие идентификационные документы, электронные очереди – все это быстрыми темпами входит в нашу жизнь и становится повседневной практикой. Институт образования не отстает, а во многом даже опережает другие сферы на пути к цифровизации. Последние два года показали преимущества онлайн обучения, также и выявили множество проблем, связанных с переходом образования в дистанционный формат, основной из которых – отсутствие целого набора инструментов, необходимых для проведения плодотворных онлайн занятий. Особенно это выделяется во время обучения иностранным языкам – существует небольшое количество систем, позволяющих использовать различные компьютерные средства для обучения иностранным языкам, но это закрытые системы с готовыми учебными программами, что не может служить полноценным инструментом для преподавателя. В данной работе будет конкретно рассматриваться применение автоматического распознавания речи или АРР в обучении английскому языку.

1.1 Распознавание образов

Распознавание образов – это научная дисциплина, целью которой является классификация объектов по нескольким категориям или классам. Объекты называются образами [2]. За последние несколько десятилетий распознающие системы стали доступны только благодаря значительным достижениям в области компьютерной обработки. Однако многие из этих новых автоматизированных методов основаны на идеях, которые изначально были задуманы сотни и даже тысячи лет назад. Один из старейших и наиболее основных примеров

характеристики, которая использовалась для распознавания людьми, – это лицо. Данный способ идентификации известных (знакомых) личностей был удобен для малых групп населения, но с течением времени появлялись новые способы путешествий и количество людей (а, значит, и объектов для распознавания) стремительно увеличивалось благодаря таким миграциям. В связи с этим возникла необходимость в новых систематизированных способах распознавания, в том числе основанных на поведенческих характеристиках человека (например, речь или походка) [3]. На сегодняшний день в качестве объекта распознавания могут выступать: лицо, отпечаток пальца и другие биометрические параметры человека; звуковой сигнал (в том числе речевой); символы (например, в виде текста); различные изображения; другие объекты. Несмотря на то, что создание универсальных алгоритмов с точностью распознавания до 100 % является крайне трудоемким процессом и на данный момент представляется практически невозможным, уже существуют распознающие системы, показывающие высокий процент распознавания и активно используемые в разных сферах деятельности человека.



Рисунок 1 - Распознавание образов

Среди представленных выше примеров распознавание речевых сигналов является активно развивающейся областью науки. Речевая биометрия является одним из способов идентификации личности для обеспечения информационной безопасности (голосовая верификация) [3], используется для автоматического заполнения документов (например, в медицинских организациях при заполнении амбулаторных карт или в судах для автоматизации протоколирования), а также применяется правоохранительными органами и службой безопасности для детектирования опасных переговоров в целях предотвращения террористических актов [4]. В данной статье рассмотрен еще один из частных примеров применения этой технологии на практике, а именно при создании обучающего языкового тренажера.

1.2 Постановка задачи распознавания речи

Существует множество различных информационных технологий, позволяющих усовершенствовать навыки аудирования, чтения и письма при изучении иностранных языков, но направлению улучшения навыков разговорной речи уделяется недостаточно внимания. Но в то же время для обучающегося важно научиться правильно произносить звуки, буквы и целые слова, так как это является основой для коммуникации с носителем изучаемого языка. На сегодняшний день наиболее перспективным направлением при решении данной проблемы является внедрение технологий распознавания речи, в основе которых лежат особенности голоса обучающегося [4, 9]. Каждый человек обладает уникальным голосом, но с фонетической точки зрения речь состоит из множества различных звуков, имеющих артикуляционные различия. Эти звуки в общем случае принято называть фонемами. Но в разных словах одни и те же фонемы

могут видоизменяться, поэтому в практику также вводятся аллофоны – варианты фонем [5]

Для успешного распознавания речи обычно рассматривают участки звукового сигнала в несколько десятков миллисекунд, называемые фреймами [7]. Выполнить эту задачу довольно сложно ввиду того, что некоторые фонемы достаточно похожи друг на друга, но можно решить ее в терминах «вероятностей». Одни фонемы более вероятны для данного сигнала, другие – менее. Строится акустическая модель, которая является функцией, принимающей на вход участок небольшого звукового сигнала (фрейм) и выдающей распределение вероятностей различных фонем на этом фрейме. На основе акустической модели можно с определенной степенью уверенности восстановить то, что было произнесено [6].

1.3 Обзор аналогов

На сегодняшний день существует достаточно большое количество сервисов, предоставляющих возможность изучения иностранных языков с использованием ПК или мобильных устройств, использующие те или иные методы распознавания и анализа речи, также эти сервисы отличаются методикой применения своих алгоритмов в обучении. Рассмотрим несколько самых популярных примеров: Одним из первых подобных решений на рынке было приложение Rosetta Stone, выпущенное в 1992 году и одним из первых предоставившее возможность использования распознавание речи в обучении иностранному языку, за 30 лет существования, система обрела возможность обучения более чем 30 языкам, были разработаны полноценные учебные программы, разделенные по уровням языка, также не так давно стало возможным использовать сервис на мобильных устройствах. На ПК полноценные курсы присутствуют только в платном варианте, на мобильных устройствах базовые упражнения бесплатны,

упражнения более высоких уровней доступны по платной подписке. Отличительная черта Rosetta Stone в использовании комбинаций текста, изображений и звука для того, чтобы обучающийся запоминал слова и грамматику интуитивно, без использования словарей. По мере развития ученика сложность повышается. В рекламе продукта говорится, что это такой же способ обучения, которым люди изучают свой первый язык. Например, программа демонстрирует обучающемуся четыре фотографии. Далее даётся звуковое и текстовое пояснение, которое описывает одну из фотографий. Обучающийся должен выбрать фотографию, которая наиболее соответствует описанию. В другом варианте обучающийся должен завершить описание фотографии. Программное обеспечение может оценить произношение слов. Можно настроить тонкость восприятия произношения, что позволяет тренировать правильную речь.

Следующий аналог – приложение Busuu. Busuu — это приложение для изучения языков. Оно доступно в Интернете через iOS и Android. Курсы доступны на двенадцати языках: английском, испанском, французском, немецком, итальянском, бразильском португальском, русском, польском, турецком, арабском, японском и китайском. Учащиеся выбирают один или несколько из этих языков и работают с помощью самостоятельных уроков, изучая лексику и грамматику, и затем они могут практиковать свои знания в разговорных или письменных беседах с носителями языка в сообществе Busuu. По состоянию на декабрь 2017 года сайт Busuu насчитывает более 80 миллионов зарегистрированных пользователей в своём сообществе.

Еще одним очень распространенным аналогом является сервис Дуолинго Duolingo — бесплатная платформа для изучения языка и краудсорсинговых переводов. Сервис разработан так, что по мере прохождения уроков пользователи параллельно помогают переводить веб-сайты, статьи и другие документы. По состоянию на январь 2022 года пользователи, знающие русский язык, могут

изучать английский, немецкий, французский и испанский языки; в процессе подготовки — шведский и украинский. Эффективность подхода Дуолинго, основанного на анализе статистики, была проверена сторонним исследованием по заказу компании. Исследование, проведённое профессорами Городского университета Нью-Йорка и Университета Южной Каролины, показало, что 34 часа на Дуолинго дают столько же навыков чтения и письма, сколько даёт начальный семестровый курс в американском высшем учебном заведении, занимающий около 130 часов. В процессе исследования 108 из 196 человек, изучающих язык на Дуолинго, бросили занятия менее чем через два часа. То же исследование показало, что пользователям Rosetta Stone необходимо от 55 до 60 часов занятий, чтобы выучить такой же объём знаний.

Дуолинго предлагает многочисленные письменные уроки и диктанты, однако разговорным навыкам уделяется меньше внимания. В Дуолинго есть игровое дерево навыков, по которому продвигаются пользователи, и словарный раздел, где можно практиковать уже изученные слова. Пользователи получают «очки опыта» (монеты, баллы) по мере изучения языка, например, после прохождения урока. Навыки считаются изученными, когда пользователи выполняют все связанные с ними уроки. За один урок можно заработать 10 очков. В Дуолинго также есть функция тренировки на время, когда пользователям даётся 30 секунд и двадцать вопросов. За каждый правильный ответ даётся одно очко опыта и семь или десять дополнительных секунд (время зависит от длины вопроса). За один курс пользователь может изучить до 2000 слов. Дуолинго использует для обучения подход, основанный на анализе большого количества статистических данных. На каждом этапе система запоминает, какие вопросы вызвали у пользователей трудности и какие ошибки были совершены. Затем она агрегирует эти данные и использует для машинного обучения. Таким образом формируются индивидуальные уроки.

Результаты и выводы по первой главе

Проведя обзор аналогов, мы можем сделать вывод о популярности данного типа сервисов и потенциале развития нашей системы. Также можем отметить и основные отличия нашей системы и существующих аналогов, все вышеперечисленные сервисы уже имеют встроенную в себя программу обучения, заранее разработанную, система же, описываемая в работе, предполагает создание собственной программы обучения учебным заведением или конкретным преподавателем, что является новым предложением на рынке и не имеет прямых аналогов.

2. ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ

Для реализации поставленных задач проекта требуются следующие компоненты:

- 1) Модуль прослушивания микрофона
- 2) Модуль распознавания речи
- 3) Модуль перевода речи в текст
- 4) Модуль перевода текста в речь
- 5) Модуль анализа полученного текста
- 6) База данных для хранения текстов и результатов анализа

2.1 Функциональные требования

В системе предусматривается следующий перечень участников: клиентская часть, серверная часть, а также группа лиц, взаимодействующих с сервисом, то есть пользователи

1.1.1 Реализуемые задачи

Перечень задач разрабатываемой системы, следующий:

1. Установка эталонного текста.
2. Установка словарей.
3. Прослушивание микрофона пользователя.
4. Распознавание речи пользователя.
5. Анализ и сопоставление текста с эталоном и словарями.
6. Перевод текста в речь.
7. Сохранение статистики

2.1.2 Требования к ПО

Программное обеспечение подразделяется на клиент и сервер. Клиентов может быть неограниченное количество, т.к. основные вычисления производятся на устройстве пользователя, что позволяет не нагружать сервер алгоритмическими вычислениями и выполнять только задачи по манипуляции с данными и авторизацией.

Теоретически предполагаемые функциональные возможности сервера по отношению к клиентам:

- 1) Создание, хранение и передача данных.
- 2) Проверка токенов аутентификации.
- 3) Переадресация HTTP запросов.

2.2 Нефункциональные требования

2.2.1 Условия эксплуатации

Система предполагает как использование в учебных заведениях, так и домашнее использование, в случае дистанционного обучения, что в последние годы набирает все большую популярность и является неотъемлемым требованием к современным информационным системам, используемым в обучении.

2.2.2 Пользователи

Требования к пользователям включают в себя правила пользования ПК, базовую компьютерную грамотность. Также система не имеет возрастных ограничений.

2.2.3 Характеристики сервера

Минимальными системными требованиями для сервера являются:

- 1) ОС Windows не старше Server 2008 или Linux, поддерживающие Node.js версии 16.15.1
- 2) 2 Гб оперативной памяти
- 3) MongoDB версии 4.4
- 4) Хранилище из примерного расчета 5-10 Мбайт на каждого предполагаемого пользователя, для хранения статистики в течении необходимого периода времени и 1-2 Гбайт на каждого преподавателя, пользующегося системой, для хранения всех текстов, словарей и других методических материалов

2.2.4 Характеристики клиента

Для работы системы на стороне клиента минимальные требования устройству, следующие:

- 1) ОС Windows, Linux, MacOS, Android, IOS, с поддержкой браузеров из пункта 2 (состояние на 2022г.)
- 2) Браузеры указанные на рисунке 2
- 3) 4 Гб оперативной памяти
- 4) Микрофон, гарнитура или веб-камера с микрофоном

PC						Mobile					
Chrome	Edge	Firefox	Internet Explorer	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android
✓	✓	✗	✗	✗	✓	✓	✗	✗	✓	✓	✓
33	79	No	No	No	14.1	33	No	No	14.5	2.0	4.4.3
-x*	-x*				-x	-x*			-x	-x*	-x*

Рисунок 2 - совместимость клиента с браузерами

Результаты и выводы по второй главе

Во второй главе были определены и описаны функциональные и нефункциональные требования, описаны необходимые компоненты, перечислены реализуемые задачи, описаны требования к ПО, а также требования к характеристикам клиентов и сервера. Также были описаны условия эксплуатации и требования к пользователям

3. ПРОЕКТИРОВАНИЕ

В этой главе приводится работа над проектированием прототипа системы. Сначала рассматриваются варианты использования системы в различных ситуациях. Затем приводится архитектура компонентов прототипа. После этого проектируются алгоритмы и вводится описание данных

3.1 Варианты использования системы

Для решения задач обучения иностранному языку предлагается ряд вариантов использования:

- 1) Чтение текста и сравнение результата с эталоном
- 2) Произношение текста и проверка на соответствие тематике и содержание слов из predetermined словарей
- 3) Чтение текста и замер скорости чтения
- 4) Произношение текста и проверка на полноту речи
- 5) Прослушивание текста, читаемого системой

Таким образом, система предлагает помощь в развитии таких навыков как: чтение, говорение и аудирование, которые являются необходимыми для полноценного освоения любого языка. Подробнее рассмотреть типичный вариант использования можно на рисунке 3.

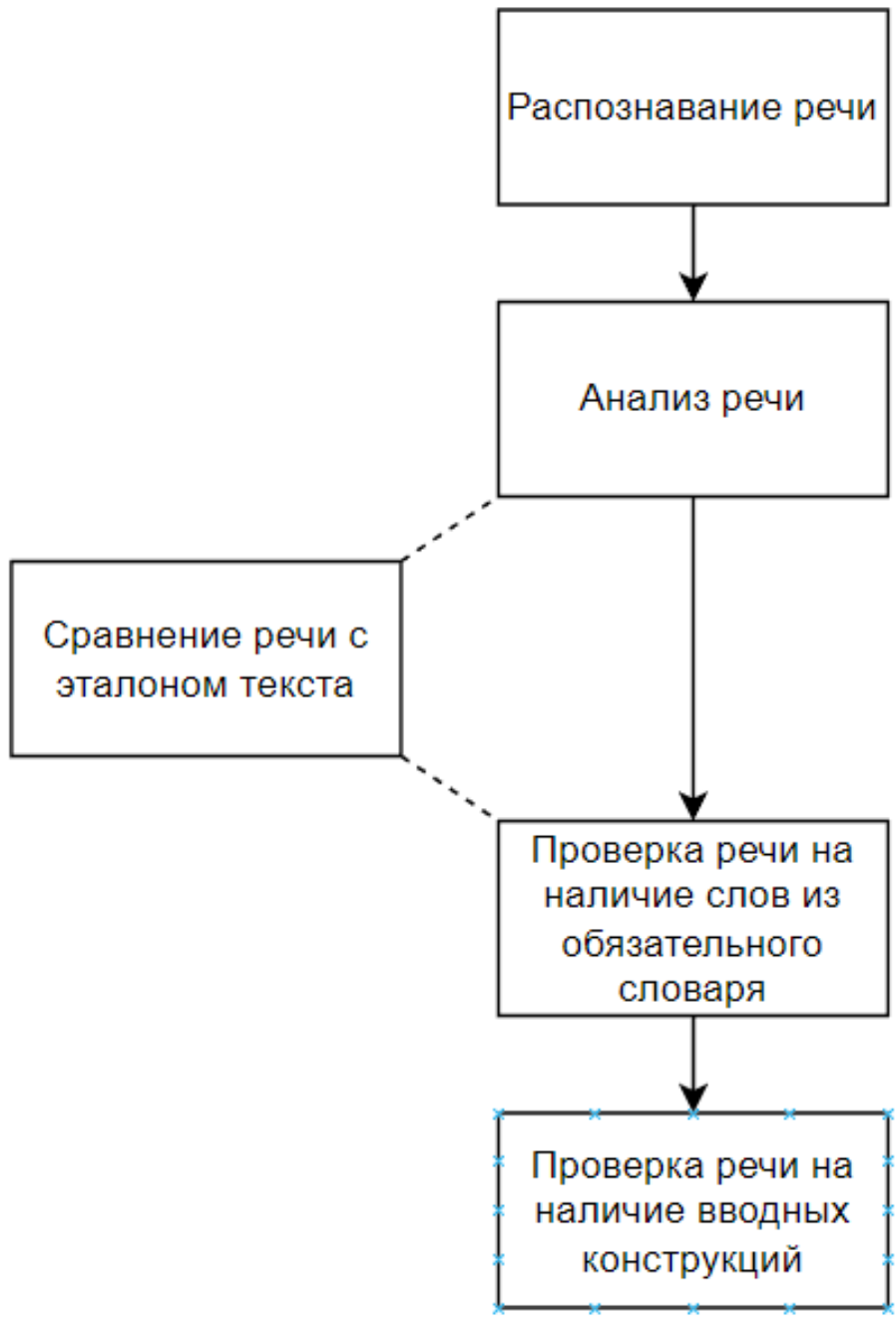


Рисунок 3 - типичный процесс использования системы

3.2 Архитектура компонентов

Для функционирования системы понадобится следующий набор компонентов:

- 1) Основной компонент обращения к ОС для прослушивания микрофона и обращения к Web Speech Api для распознавания текста,
- 2) Компонент присвоения эталонного текста
- 3) Компонент присвоения словаря
- 4) Компонент сравнения распознанного текста и эталона
- 5) Компонент замера скорости чтения
- 6) Компонент сравнения распознанного текста и словаря
- 7) Компонент перевода любого текста в речь

Такой метод разбивки системы на функциональные компоненты позволяет обеспечить чистоту кода и рабочего пространства, также упрощает дальнейшее масштабирование системы и расширение функционала.

3.3 Проектирование алгоритма сравнения и анализа текста

Было решено разрабатывать алгоритм сравнения и анализа текста самостоятельно средствами языка без использования сторонних библиотек и заимствования чужого программного кода.

Входными данными для алгоритма был выбран такой примитив языка JavaScript, как строка, так как это наиболее удобный вариант для пользователя, при использовании строк пользователю не придется самостоятельно решать вопросы с форматированием, знаками препинания и другими тонкостями работы с текстом в языках программирования.

В первую очередь строка должна приводиться к виду удобному для сравнения с другими строками (эталонным текстом, словарями и др.), для соблюдения этого

условия необходимо привести все морфемы содержащиеся в строке к единообразию, для чего сначала все буквы приводятся к строчному виду, затем все знаки препинания заменяются на пробелы, после чего строка переводится в массив с разделителем “пробел” в итоге мы получаем массив, со словами содержащими только строчные буквы и с отсутствием знаков препинания. Знаки препинания было решено исключить из проверки т.к. на данный момент Web Speech Api не обладает функционалом расставления знаков препинания по интонации говорящего. Такие же операции мы выполняем над эталонным текстом и словарями. Для сравнения всегда получаем два массива строк, каждая из которых содержит по одному слову, что позволяет нам сравнивать массивы перебором. Также во время создания массива вычисляем общую длину строки в символах и вес в процентах каждой морфемы, для дальнейшего подсчета совпадений.

Вторым этапом алгоритма будет являться сравнение двух массивов на совпадения, что оптимально выполняется средствами языка JavaScript. Во время сравнения создаем два массива в один, из которых помещаем слова, попавшие в пересечения, во второй те, что не попали. Так как вес каждого слова нам известен посчитать процент совпадений не составит труда. Отображаем пользователю процент совпадения и слова не попавшие в пересечение.

Сравнения других видов, например на попадание слов из словаря в распознанную речь будут основаны на модифицированных версиях вышеописанного алгоритма, так как сам по себе алгоритм является универсальным и может применяться для разного вида сравнения и анализа текста.

Результаты и выводы по третьей главе

Были сформулированы варианты использования прототипа системы, описана архитектура функциональных компонентов, и спроектирован универсальный алгоритм сравнения и анализа текста.

4. РЕАЛИЗАЦИЯ

В этой главе рассматриваются способы реализации спроектированных компонентов и объединение их в рабочую систему.

Для моделирования работы системы и взаимодействия ее ключевых элементов реализован прототип клиента, разработанный с помощью JavaScript и фреймворка Vue.JS и прототип сервера, разработанный на Node.JS.

4.1 Общее описание принципов реализации

Для реализации клиентского модуля использовался паттерн программирования “MVC”, предполагающий разделение интерфейса, бизнес-логики и моделей данных по разным классам (в случае с Vue.JS по разным файлам). Данный паттерн был выбран из-за удобства разработки, которое он предоставляет, также был использован компонентный подход, позволяющий сократить кодовую базу за счет переиспользования однотипных компонентов, также позволяющий придерживаться принципа “DRY” – Don’t Repeat Yourself, что считается хорошей практикой в среде разработчиков.

Таким образом мы получаем отдельный компонент на каждый функциональный модуль, например компонент распознавания речи, компонент проверки совпадения текстов, компонент проверки совпадения текста и словаря и т.д.

Изначально, в разработке модулей прослушивания микрофона и распознавания речи использовался сервис от компании Google, “Speech To Text”, но в процессе разработки правила использования сервиса изменились и стали неподходящими для реализации описываемой системы. В связи с чем текущая реализация модулей прослушивания микрофона и распознавания речи выполнена с помощью встроенных языковых функций JavaScript, называемых Web Speech Api, без использования сторонних библиотек или ограничений от

каких-либо корпораций, что является преимуществом, как для дальнейшей разработки и поддержки, так и для использования системы в различных целях.

4.2 Реализация алгоритма анализа текста

Алгоритм анализа текста самописный и так же реализован исключительно средствами языка JavaScript. Алгоритм представлен в листинге 1.

```
//Разделяем полученную строку на отдельные слова
const testStrings = (...args) => {
  const [wArr, test] = Array.from(args).map((s) =>
    s
    .toLowerCase()
    .replace(/^[^a-z\s]/g, "")
    .split(" ")
    .filter(Boolean)
  );
  //Создаем массив строк, содержащих одно слово, замеряем вес каждого слова в
  //процентном соотношении от всего текста
  const sumWeight = wArr.join("").length;
  const wMap = wArr.reduce((acc, word) => {
    if (!(word in acc)) {
      acc[word] = {
        count: 0,
        seen: false,
        weight: (100 / sumWeight) * word.length,
      };
    }
    acc[word].count += 1;
    return acc;
  }, {});
  //Сравниваем слова из получившегося массива с эталонным текстом, совпадения и
  //несовпадения разносим в разные массивы
  const diff = [];
  return {
    match: test.reduce((acc, word) => {
      if (wMap[word]?.count) {
        wMap[word].count--;
        wMap[word].seen = true;
        acc += wMap[word].weight;
      } else {
        diff.push(word);
      }
    }, 0),
    diff: diff.concat(
      Object.keys(wMap).filter((word) => {
```

```
        return !wMap[word].seen;
    })
),
};
};
```

Листинг 1 – Алгоритм сравнения текстов.

Алгоритм позволяет посчитать количество слов в тексте, вес каждого отдельного слова в процентах, в зависимости от количества символов, по морфемно сравнить распознанный текст и любой другой. Алгоритм создавался таким образом, чтобы его можно было переиспользовать в ряде подобных задач, чего удалось добиться и этот же алгоритм применяется, как в сравнении текста с эталоном, так и со словарем необходимых слов, с небольшими доработками, где сравнивается не процент совпавших слов, а процент присутствующих слов из словаря.

4.4 Реализация клиента и сервера

Работа системы выстроена таким образом, чтобы минимизировать обращения к серверу. Рассмотрим схемы процессов, выполняемых клиентом и сервером, отображенных на рисунке 3.

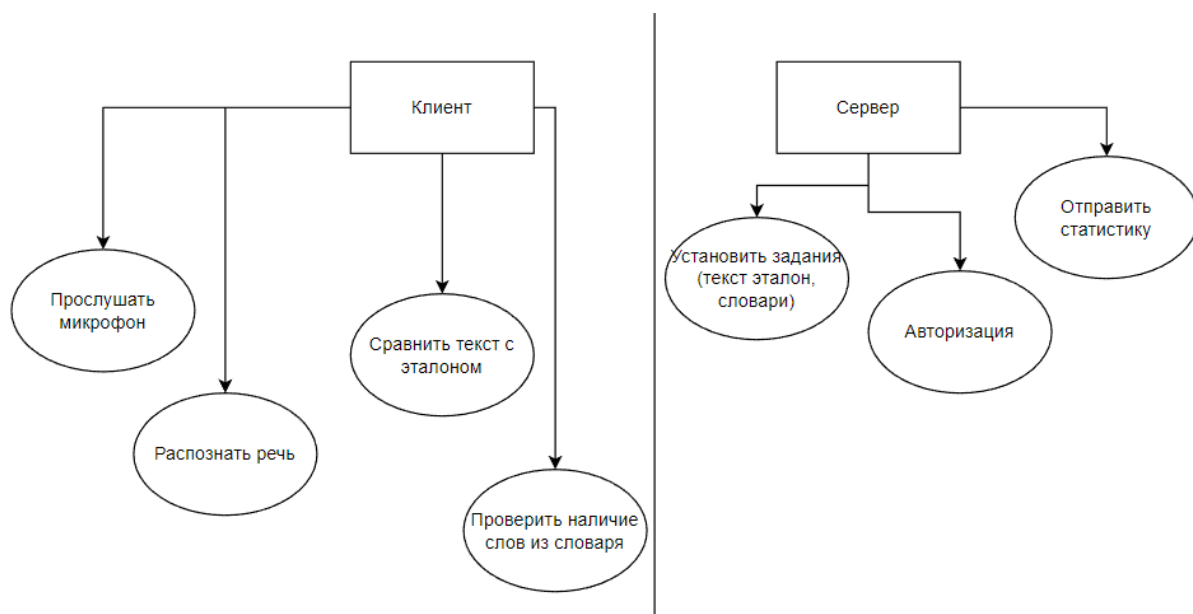


Рисунок 4 - схема процессов, выполняемых клиентом и сервером

Здесь мы видим, что сервер выполняет только одноразовые задания, необходимые пользователям. Так, например преподаватель только единожды устанавливает эталонный текст для каждой темы, процессы авторизации и отправки статистики также выполняются один раз, при входе в систему и при выходе из нее – соответственно.

Все ресурсо-затратные операции производятся мощностями ПК пользователя, что позволит в будущем при распространении системы держать сервер практически на любом школьном персональном компьютере, что несомненно является большим преимуществом перед системами, требующими больших вычислительных мощностей. При этом в связи с постоянной работой разработчиков над браузерами и языком JavaScript, требования к ПК пользователей, на которые ложатся основные вычислительные задачи, относительно не высокие, единственные требования – это поддержка современных браузеров и объем оперативной памяти достаточный для стабильной работы браузера. Таким образом внедрение системы в образовательное учреждение не требует увеличения мощностей ни со стороны организации, ни со стороны пользователей.

4.5 Реализация модулей системы

Для упрощения демонстрации интерфейсы студента и преподавателя были объединены на одной веб-странице, в версии готовой к использованию, доступ к интерфейсам будет реализован на основании групп пользователей и после авторизации, таким образом студент будет видеть только ту часть, с которой ему необходимо работать, а преподаватель сможет редактировать учебную программу для групп студентов или при необходимости составлять персональную программу обучения.

На рисунке 4 изображен модуль распознавания речи и счетчик скорости чтения, при первом нажатии на зеленую кнопку старт, браузер обратится к пользователю за получением разрешения на использование микрофона, в дальнейшем это не потребуется, если в настройках браузера не выставлено обратное. По окончании чтения текста или монолога необходимо нажать синюю кнопку “Stop and Confirm text”. По нажатии запись окончится и распознанный текст сохранится в состоянии приложения, что необходимо для его дальнейшего анализа. Над счетчиком скорости чтения расположено текстовое окно, которое в реальном времени отображает распознанный на данный момент текст, что удобно для проверки правильности распознавания, что необходимо, так как в распознавании могут быть погрешности, связанные с произношением, акцентом и качеством звукозаписывающего устройства. Также в случае каких-либо ошибок над счетчиком скорости чтения отобразится красная рамка с указанием вида ошибки, например: ошибка сети, ошибка микрофона и т.д.

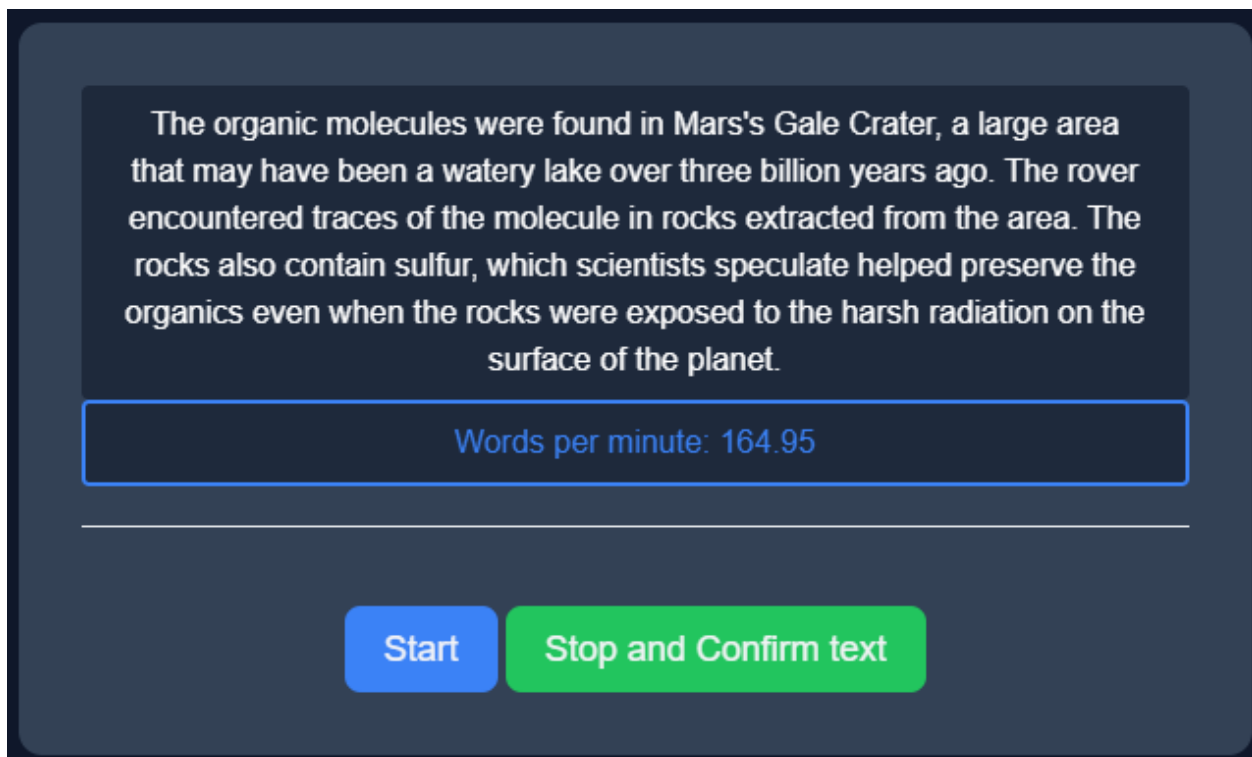


Рисунок 5 - Модуль распознавания речи

На рисунке 5 изображен модуль установки эталонного текста, доступный преподавателям, в модуле присутствует текстовое поле и кнопка “Save”, чтобы установить эталонный текст, с которым будет сравниваться распознанная речь студента необходимо вставить или ввести текст в текстовое поле и нажать кнопку “Save”.

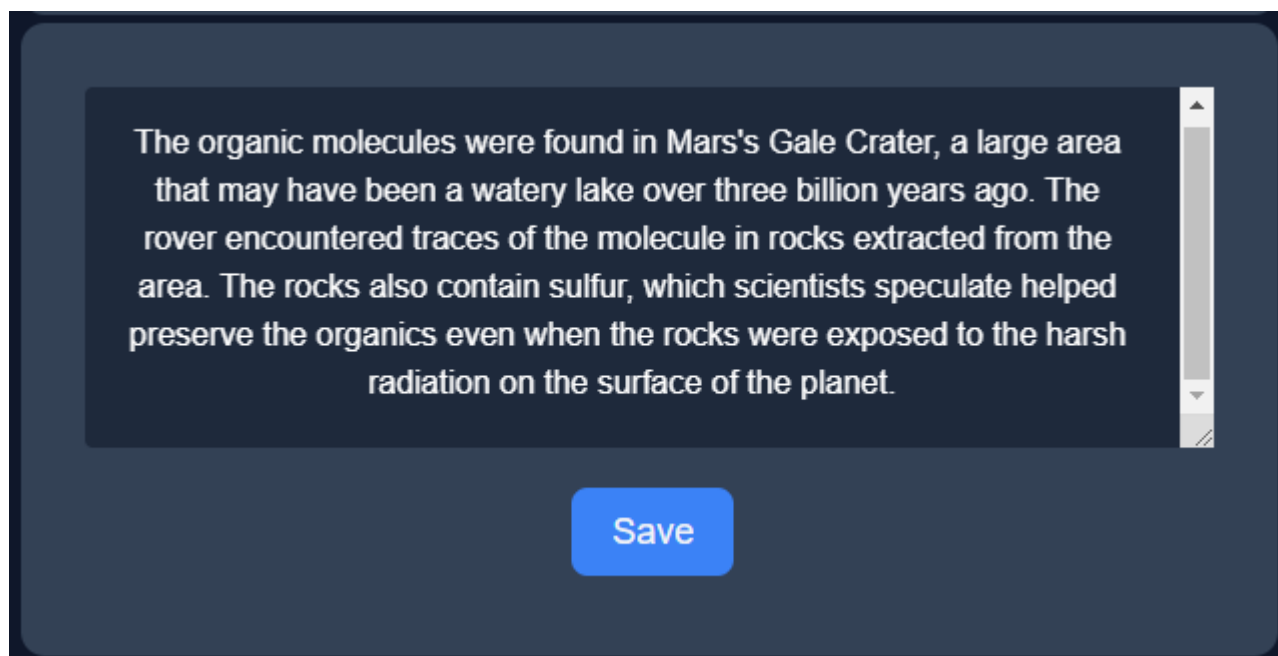


Рисунок 6 - модуль установки эталонного текста

Студенту также будет доступен модуль сравнения распознанной речи с эталонным текстом – он изображен на рисунке 6. Модуль содержит в себе кнопку “Compare”, при нажатии на которую будет приведен в действие алгоритм анализа и сравнения текста и в случае успешного выполнения отобразятся два поля, содержащие процент совпадения речи с эталоном и перечисленные через запятую слова, которые были пропущены или в которых была допущена ошибка, в случае стопроцентного совпадения второй блок отображен не будет. Также в случае ошибки выполнения алгоритма будет отображена ошибка.

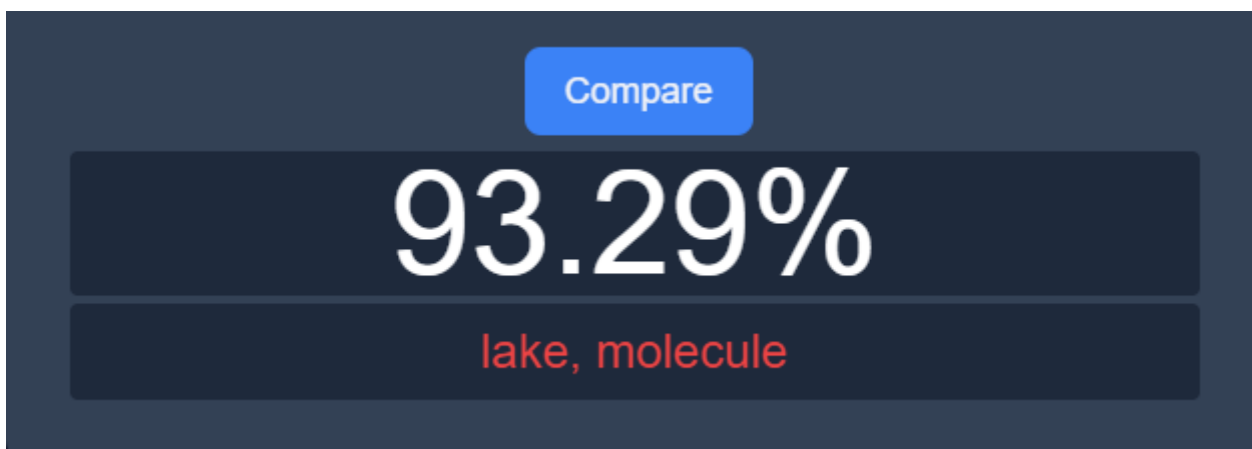


Рисунок 7 - модуль сравнения распознанной речи и эталонного текста

Еще один модуль, доступный студенту – модуль проверки наличия слов из необходимого словаря в распознанной речи. Модуль содержит в себе кнопку “Check vocabulary”, по нажатии на которую происходит запуск алгоритма сравнения и при успешном выполнении отображаются три блока, в зависимости от наличия в них наполнения:

- 1) Блок со счетчиком в процентах наличия слов из словаря в речи
- 2) Блок использованных слов, перечисленных через запятую
- 3) Блок неиспользованных слов

Модуль изображен на рисунке 7

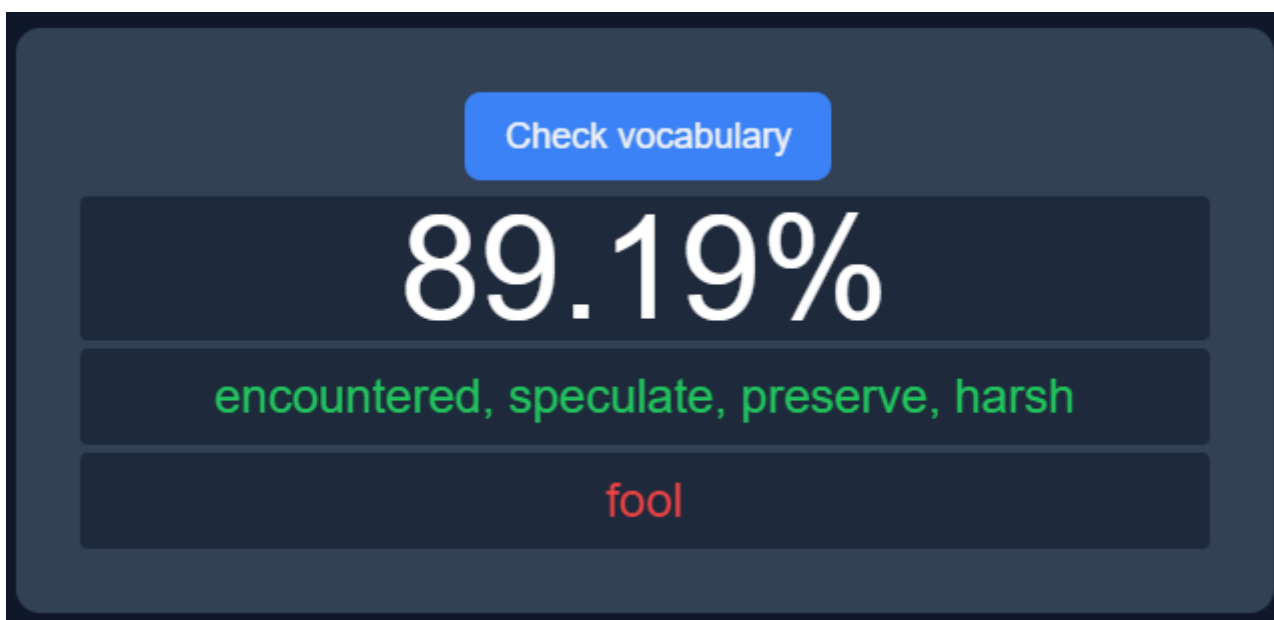


Рисунок 8 - модуль проверки наличия в речи слов из обязательного словаря

Таким образом можно заметить, что интерфейс приложения интуитивен, основные модули не перегружены функциями и содержат в себе только необходимые элементы. Также верстка выполнена с применением адаптивных технологий, что гарантирует правильное отображение интерфейса на дисплеях с различными расширениями.

Результаты и выводы по четвертой главе

Был реализован рабочий прототип системы, демонстрирующий основные функции. Для реализации были использованы современные и популярные технологии, которые позволяют упростить разработку, масштабирование и дальнейшую поддержку системы. В разработке был использован компонентный подход и модульный интерфейс. Интерфейс был создан максимально интуитивным и адаптивным. Все намеченные задачи были реализованы.

5. ТЕСТИРОВАНИЕ

5.1 Описание методов тестирования

В этой главе рассматриваются методы тестирования системы и предоставлены результаты его проведения. Для тестирования приложение было предоставлено выпускникам факультета преподавания иностранных языков, была собрана обратная связь и систематизирована для дальнейших выводов. Тестирующим давалось описание системы и инструкция по применению. После нескольких дней использования приложения тестирующие заполняли анонимную анкету, с которой и собирались данные. Статистка представлена на рисунках, содержащих диаграммы, составленные по ответам рецензентов на вопросы анкеты.

5.2 Анализ данных предоставленных в результате тестирования

Большинство респондентов высоко оценили точность распознавания речи системой, что говорит о высоком качестве распознавания и о потенциале метода распознавания в будущем т.к. Web Speech Api был добавлен в JavaScript недавно, и работа над улучшением качества ведется по сей день точные данные можно увидеть на рисунке 9.

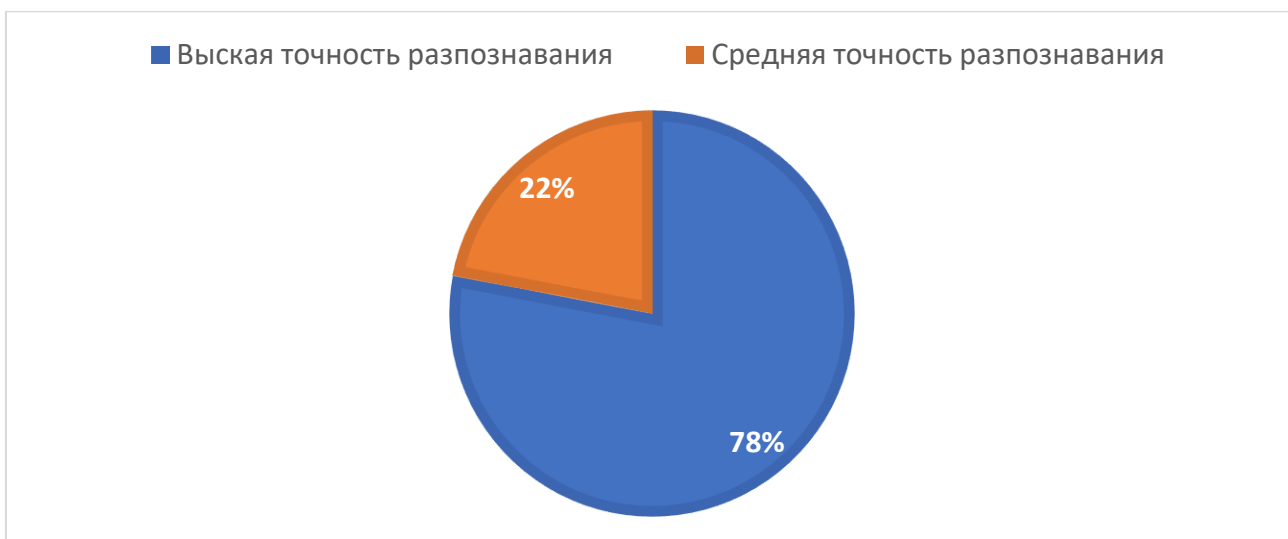


Рисунок 9 - оценка точности распознавания речи

Респонденты в подавляющем большинстве оценили удобство интерфейса как высокое или среднее, что говорит о правильной компоновке и надлежащем внешнем виде компонентов приложения, также высоко был оценен и пользовательский опыт. Лишь десяти процентам тестируемых интерфейс показался не удобным. Точные данные отображены на рисунке 10.



Рисунок 10 - Оценка уровня удобства интерфейса

Одним из важнейших вопросов в анкете являлся вопрос о готовности использовать систему в профессиональной деятельности. Большинство

опрашиваемых готовы использовать приложение в работе после доработок, которые были предложены самими опрашиваемыми, но также процент преподавателей не готовых использовать систему также высок, в большинстве случаев это связано с невозможностью по своему желанию внедрять в программу обучения дополнительные средства или с нежеланием менять устоявшиеся методики преподавания. Точные данные отражены на рисунке 11



Рисунок 11 - Оценка готовности использовать систему в обучении

5.3 Список предложенных доработок системы

Список нескольких из предложенных доработок:

- 1) Добавить возможность создавать учебную программу, разделенную на темы/разделы, с заранее заготовленными текстами, темами монологов и тематическими словарями;
- 2) Добавить возможность прослушивать речь, записанную студентом;
- 3) Добавить возможность записать речь преподавателя, для предоставления эталонов произношения;
- 4) Добавить возможность составления отчетов и просмотра статистики по каждому студенту;

5) Добавить возможность оставлять обратную связь по полученным ответам от студентов;

6) Добавить задания с изображениями и другими медиафайлами

Большинство из предложенных доработок планируются в релизной версии продукта, что подчеркивает правильность намеченного вектора развития продукта. Также некоторые доработки являются неплохим дополнением к будущему функционалу и будут рассмотрены к разработке.

Результаты и выводы по пятой главе

Тестирование показало, что прототип разработан качественно и выполняет поставленные перед ним задачи. Были получены данные, включающие мнение респондентов о текущем состоянии системы, которое в большинстве своем положительное, также тестировщиками были предложены доработки, некоторые из которых уже стоят в плане развития системы, что подчеркивает верно установленный вектор будущих разработок. Готовность преподавателей использовать приложение в работе неоднозначна. Для уточнения этих данных необходимо продолжить разработку и тестирование системы. Тестирование можно считать удачным, т.к. были получены необходимые данные для оценки работы системы и ее дальнейшего развития.

6. ПРИМЕНЕНИЕ

6.1 Инструкция по применению прототипа системы

В этой главе предоставлены рекомендации по применению системы.

Для получения положительного опыта использования системы и релевантных результатов был разработан алгоритм подготовки устройства пользователя и использования системы:

- 1) Скачать/Обновить интернет-браузер “Google Chrome” до последней версии;
- 2) Разрешить использование JavaScript в браузере (если было запрещено);
- 3) Подключить звукозаписывающее устройство (микрофон, гарнитура, веб-камера);
- 4) Разрешить браузеру использовать микрофон;
- 5) Для начала прослушивания нажать кнопку “Start”, прочитать текст в микрофон;
- 6) Нажать кнопку “Stop and Confirm text” для завершения прослушивания;
- 7) Ввести эталон текста, с которым необходимо сравнить прослушанную речь (в прототипе);
- 8) Нажать кнопку “Compare”, для сравнения;
- 9) После этих шагов вы получите результаты анализа вашей речи, такие как:
 - Техника чтения (количество слов в минуту);
 - Процентное соотношение совпадения речи и эталонного текста;
 - Список слов, которые не совпали (ошибки);
 - Процент использованных слов, из обязательного словаря (тематический или по уровню языка);
 - Список слов из обязательного словаря, которые не были использованы в речи.

Таким образом, совершив несколько простых шагов, мы произведем прослушивание речи, перевод речи в текст, анализ и сравнение речи по необходимым параметрам, получим результаты анализа, позволяющие оценить навыки говорения у студентов.

ЗАКЛЮЧЕНИЕ

При решении поставленных задач получены результаты, которые заключаются в следующем:

1. Выявлены специфические аспекты технологии распознавания речи и особенностей ее использования;
2. Проведен анализ существующих аналогов обучающей системы с распознаванием речи. Выделены их преимущества и недостатки;
3. Спроектирован и разработан алгоритм анализа и сравнения текстов;
4. Спроектирован и разработан прототип системы обучения иностранному языку, использующий технологии распознавания речи;
5. Прототип системы протестирован и намечено дальнейшее развитие системы;

На основании полученных результатов были сделаны следующие выводы:

1. Распознавание речи в обучении иностранному языку может успешно применяться;
2. Разработанный прототип показывает возможность создание полноценной рабочей системы обучения иностранному языку, использующей технологии распознавания речи;

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. M. Gales and S. Young The Application of Hidden Markov Models in Speech Recognition. Foundations and Trends in Signal Processing, 2007, vol. 1, No. 3, pp. 195- 304.
2. Верещагина Е. А., Фоминова Е. Р., Восприятие иностранного языка при помощи аудиального канала восприятия для обучения иностранному языку, 39 международная научно-методическая конференция «Актуальные вопросы качества образования», 8 декабря 2015 г.
3. Лебеденко, Ю. И. Биометрические системы безопасности: пособие / Ю. И. Лебеденко. – Тула: Издательство ТулГУ, 2012. – 159 с.
4. Материалы Региональной научно-практической конференции студентов, аспирантов и молодых учёных по естественным наукам, Владивосток, 15-30 апреля 2016 г. [Электронный ресурс]. – Электрон. дан. – Владивосток: Дальневост. федерал. ун-т, 2016. – URL: https://www.dvfu.ru/schools/school_of_natural_sciences/sciences/theconference/new-page.php.
5. Панина, Т. С. Современные способы активизации обучения: учебное пособие / Т. С. Панина, Л. Н. Вавилова; под ред. Т. С. Паниной. – 4-е изд., стер. – М.: Издательский центр «Академия», 2010. – 176 с.
6. Первушин Е. А. Обзор основных методов распознавания дикторов. Математические структуры и моделирование, 2011, № 24, с. 41-54.
7. Рабинер Л. Р., Шафер Р. В. Цифровая обработка речевых сигналов / Пер. с англ. М.: Радио и связь, 1981. 496 с.
8. Современные образовательные технологии: учебное пособие / под ред. Н. В. Бордовской. – 2-е изд., стер. – М.: КНОРУС, 2011. – 432 с
9. Сорокин В. Н., Вьюгин В. В., Тананыкин А. А. Распознавание личности по голосу: аналитический обзор. Информационные процессы, 2012, Том 12, № 1, с. 1-30.

10. Тампель И. Б. Автоматическое распознавание речи – основные этапы за 50 лет // Научно-технический вестник информационных технологий, механики и оптики. 2015. №6. URL: <http://cyberleninka.ru/article/n/avtomaticheskoe-raspoznavanie-rechiosnovnyye-etapy-za-50-let>.

ПРИЛОЖЕНИЕ

Листинг кода модуля распознавания речи

```
// обращаемся к Web Speech Api
var webkitSpeechRecognition = window.speechRecognition
|| window.webkitSpeechRecognition;
var recognition = new webkitSpeechRecognition();

recognition.continuous = true;
recognition.interimResults = true;
recognition.lang = "en-US";

export default function useSpeechRecognition() {

  const isListening = ref(false);
  const note = ref("The organic molecules were found
in Mars's Gale Crater, a large that may have been a watery
over three billion years ago. The rover encountered traces
of the in rocks extracted from the area. The also contain
sulfur, which scientists speculate helped preserve the
organics even when the rocks were exposed to the harsh
radiation on the surface of the planet.");
  const error = ref(null);

  // объявляем функции начала и окончания записи
  const handleListen = () => {
    if (isListening.value) {
      recognition.start();
    } else {
      recognition.stop();
    }
  };

  const startListening = () => {
    isListening.value = true
  }
  const stopListening = () => {
    isListening.value = false
  }
  const toggleListening = () => {
    isListening.value = !isListening.value;
  }
}
```

```

    };
    // возвращаем результат записи и распознавания
    recognition.onresult = (event) => {
        const transcript = Array.from(event.results)
            .map((result) => result[0])
            .map((result) => result.transcript)
            .join("");
        note.value = transcript;
    };
    //Обработка ошибок
    recognition.onerror = (event) => {
        error.value = event.error;
    };
    //Следим за состоянием
    watch(isListening, () => {
        handleListen();
    });
    // Экспорт объектов
    return {
        startListening,
        stopListening,
        toggleListening,
        note,
        error,
    };
}

```

Листинг кода модуля проверки на наличие в распознанной речи слов из словаря

```

const testStrings = (...args) => {

```

```

const [wArr, test] = Array.from(args).map((s) =>
  s
    .toLowerCase()
    .replace(/[^a-z\s]/g, "")
    .split(" ")
    .filter(Boolean)
);
const sumWeight = wArr.join("").length;
const wMap = wArr.reduce((acc, word) => {
  if (!(word in acc)) {
    acc[word] = {
      count: 0,
      seen: false,
      weight: (100 / sumWeight) * word.length,
    };
  }
  acc[word].count += 1;
  return acc;
}, {});
const diff = [];
const good = [];
return {
  match: test.reduce((acc, word) => {
    if (wMap[word]?.count) {
      wMap[word].count--;
      wMap[word].seen = true;
      acc += wMap[word].weight;
      console.log(wMap.seen);
    }
    return acc;
  }, 0),
  diff: diff.concat(
    Object.keys(wMap).filter((word) => {
      return !wMap[word].seen;
    })
  ),
  good: good.concat(
    Object.keys(wMap).filter((word) => {
      return wMap[word].seen;
    })
  ),
};
};

```