

МИНИСТЕРСТВО НАУКИ ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН ✓

Некоммерческое акционерное общество Казахский «Национальный исследовательский
технический университет имени К. И. Сатпаева



**SATBAYEV
UNIVERSITY**

Институт автоматки и информационных технологий

Кафедра «Робототехники и технических средств автоматки»

Специальность 6В07111– Робототехника и мехатроника

Таласбаев Алихан Русланулы

Тема: Разработка программно-аппаратной части для портативного устройства с обширной
системой анализа данных

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к дипломному проекту

Специальность 6В07111– Робототехника и мехатроника

Алматы 2023



SATBAYEV
UNIVERSITY

Институт автоматки и информационных технологий

Кафедра «Робототехники и технических средств автоматки»



ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту

На тему: «Разработка программно-аппаратной части для портативного устройства с
обширной системой анализа данных»

по специальности 6В07111– Робототехника и мехатроника

Выполнил

Таласбаев Алихан Русланулы

Рецензент

РнД, ассоциированный профессор
(ученая степень, звание)

Балбаев Г. К.

подпись

Ф. И. О.

«26» май 2023 г.

Научный руководитель

канд. тех. наук, ассоц. профессор

Тулешов Е. А.

«27» май 2023 г.

МИНИСТЕРСТВО НАУКИ ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество Казахский «Национальный исследовательский
технический университет имени К. И. Сатпаева»



SATBAYEV
UNIVERSITY

Институт автоматки и информационных технологий

Кафедра «Робототехники и технических средств автоматки»



ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту

На тему: «Разработка программно-аппаратной части для портативного устройства с
обширной системой анализа данных»

по специальности 6В07111– Робототехника и мехатроника

Выполнил

Таласбаев Алихан Русланулы

Рецензент

PhD, ассоциированный профессор
(ученая степень, звание)

Балбаев Г. К.

подпись

Ф. И. О.

«26» май 2023 г.

Научный руководитель

канд. тех. наук, ассоц. профессор

Тулешов Е. А.

«27» май 2023 г.

Алматы 2023

АННОТАЦИЯ

Целью данного проекта является разработка и реализация системы искусственного интеллекта с голосовым управлением, используя язык программирования Python и API ChatGPT от OpenAI. Проект предоставляет пользователю возможность взаимодействовать с искусственным интеллектом, задавая голосовые команды.

Этот проект актуален в сфере искусственного интеллекта и голосового управления, так как позволяет пользователям взаимодействовать с искусственным интеллектом через голосовые команды, делая процесс более естественным и удобным. Использование языка программирования Python и API ChatGPT открывает возможности для гибкой настройки и расширения функциональности системы.

В результате разработки этого проекта была создана система искусственного интеллекта с голосовым управлением, которая может быть использована в различных практических сферах, включая помощников, чат-ботов и другие приложения, требующие голосового взаимодействия с искусственным интеллектом.

АҢДАТПА

Бұл жобаның мақсаты-Python бағдарламалау тілі мен OpenAI CHATGPT API көмегімен дауыспен басқарылатын жасанды интеллект жүйесін әзірлеу және енгізу. Жоба пайдаланушыға дауыстық командаларды орнату арқылы жасанды интеллектпен өзара әрекеттесуге мүмкіндік береді.

Бұл жоба жасанды интеллект пен дауысты басқару саласында өзекті болып табылады, өйткені ол пайдаланушыларға дауыстық командалар арқылы жасанды интеллектпен өзара әрекеттесуге мүмкіндік береді, бұл процесті табиғи және ыңғайлы етеді. Python бағдарламалау тілін және CHATGPT API-ді қолдану жүйенің функционалдығын икемді конфигурациялау және кеңейту мүмкіндіктерін ашады.

Бұл жобаны әзірлеу нәтижесінде әртүрлі практикалық салаларда, соның ішінде көмекшілерде, чатботтарда және жасанды интеллектпен дауыстық өзара әрекеттесуді қажет ететін басқа қолданбаларда пайдалануға болатын дауыспен басқарылатын жасанды интеллект жүйесі құрылды.

ANNOTATION

The aim of this project is to develop and implement an artificial intelligence system with voice control, using the Python programming language and the ChatGPT API from OpenAI. The project provides users with the ability to interact with the artificial intelligence system by issuing voice commands.

This project is relevant in the field of artificial intelligence and voice control as it allows users to interact with artificial intelligence through voice commands, making the process more natural and convenient. The use of the Python programming language and the ChatGPT API provides opportunities for flexible customization and extension of the system's functionality.

As a result of developing this project, an artificial intelligence system with voice control has been created, which can be utilized in various practical domains, including personal assistants, chatbots, and other applications that require voice interaction with artificial intelligence.

СОДЕРЖАНИЕ

	Введение	
1	Теоретические сведения	7
1.1	Искусственный интеллект (ИИ)	7
1.1.1	Определение и понятие ИИ	7
1.1.2	Основные принципы и подходы к разработке ИИ	7
1.1.3	Области применения ИИ и его роль в современном обществе	8
1.2	Голосовое управление	9
1.2.1	Определение и концепция голосового управления	9
1.2.2	Преимущества и недостатки	10
1.2.3	Технологии и алгоритмы	11
1.3	Архитектура системы искусственного интеллекта	12
1.3.1	Компоненты и модули системы	12
1.3.2	Взаимодействие между компонентами	12
1.3.3	Роль голосового интерфейса в системе	13
1.4	Обзор существующих решений	14
1.4.1	Изучение и анализ существующих систем голосового управления	14
1.4.2	Рассмотрение популярных платформ	15
2	Разработка	17
2.1	Описание проекта	17
2.2	Цель и актуальность проекта	17
2.3	Поэтапное описание создания программы голосового ассистента	17
	Заключение	24
	Список использованной литературы	
	Приложение А	
	Приложение Б	

ВВЕДЕНИЕ

В современном обществе искусственный интеллект становится неотъемлемой частью нашей жизни. Он активно применяется во множестве сфер, от повседневных задач до сложных промышленных процессов. Благодаря своей способности обрабатывать и анализировать большие объемы данных, искусственный интеллект помогает нам принимать более обоснованные и эффективные решения[11].

Одной из самых важных и прогрессивных областей применения искусственного интеллекта является голосовое управление[12]. Системы голосового распознавания и управления позволяют нам взаимодействовать с компьютерами, мобильными устройствами и домашними ассистентами с помощью голосовых команд. Это открывает новые возможности для более естественного и удобного взаимодействия с технологией.

Моя цель в данном проекте состояла в разработке и реализации системы искусственного интеллекта с голосовым управлением, которая была бы доступна для практического использования. Я стремился создать интуитивно понятный и легко настраиваемый интерфейс, который позволит пользователям максимально эффективно взаимодействовать с системой и выполнять различные задачи.

В процессе разработки я уделял особое внимание алгоритмам голосового распознавания и обработки естественного языка[10]. Я использовал передовые методы и библиотеки, чтобы обеспечить высокую точность распознавания и понимания голосовых команд. Помимо этого, я также интегрировал API искусственного интеллекта, чтобы обогатить систему способностью к общению и предоставлению контекстно-зависимых ответов.

Результатом моей работы стала функциональная система искусственного интеллекта с голосовым управлением, готовая к практическому использованию. Я надеюсь, что эта система станет полезным инструментом для повышения эффективности и комфорта в повседневной жизни пользователей, предоставляя им возможность управлять устройствами и выполнять задачи с помощью голосовых команд.

1 Теоретические сведения

1.1 Искусственный интеллект (ИИ)

1.1.1 Определение и понятие ИИ

Искусственный интеллект (Artificial Intelligence, AI) - это область компьютерных наук, которая занимается созданием интеллектуальных систем, способных воспринимать, анализировать, обрабатывать информацию и принимать решения, аналогичные или близкие решениям, которые принимают люди[**Ошибка! Источник ссылки не найден.**].

Основная цель искусственного интеллекта - создание компьютерных программ и систем, которые могут имитировать и выполнять задачи, требующие интеллектуальных способностей человека. Это включает в себя способность обучаться на основе опыта, адаптироваться к изменяющимся условиям, делать выводы и принимать решения на основе имеющейся информации.

Одной из ключевых задач искусственного интеллекта является разработка алгоритмов и методов машинного обучения, которые позволяют компьютерам "учиться" на основе больших объемов данных и опыта. Это включает в себя использование статистических моделей, нейронных сетей, генетических алгоритмов и других методов, которые позволяют системам самостоятельно находить закономерности и делать предсказания.[12]

Искусственный интеллект находит широкое применение во множестве областей, таких как медицина, финансы, производство, транспорт, наука и др. Он способен автоматизировать задачи, улучшить процессы принятия решений, оптимизировать работу систем и повысить эффективность бизнеса. Все большее развитие и применение искусственного интеллекта открывает новые возможности для инноваций и улучшения качества жизни людей.[14]

1.1.2 Основные принципы и подходы к разработке ИИ

Разработка искусственного интеллекта (ИИ) основана на ряде принципов и подходов, которые помогают создать интеллектуальные системы. Вот некоторые из основных принципов и подходов, используемых в разработке ИИ[10]:

– Машинное обучение: Это подход, при котором компьютерные системы обучаются на основе больших объемов данных и опыта. Методы машинного обучения, такие как нейронные сети, решают задачи, опираясь на статистические закономерности и образцы в данных[11];

– Обработка естественного языка: Этот подход связан с разработкой методов и алгоритмов, которые позволяют компьютерам понимать и обрабатывать естественный язык, используемый людьми. Это включает в себя распознавание речи, генерацию текста, анализ смысла и другие техники;

– Компьютерное зрение: Этот подход связан с разработкой систем, способных анализировать и понимать изображения и видео. Компьютерное зрение используется для распознавания объектов, лиц, анализа сцен и других задач.

– Экспертные системы: Этот подход основан на создании компьютерных программ, которые моделируют знания и экспертное мнение специалистов в определенной области. Экспертные системы используют правила и логику для принятия решений и решения сложных задач;

– Генетические алгоритмы: Этот подход основан на принципах естественного отбора и эволюции. Генетические алгоритмы используются для решения оптимизационных задач и поиска оптимальных решений на основе популяции решений[12];

– Автоматическое планирование и принятие решений: Этот подход связан с разработкой алгоритмов и методов, которые позволяют системам планировать действия и принимать решения на основе определенных целей и ограничений;

– Робототехника: Этот подход сочетает различные методы и подходы для создания физических роботов с искусственным интеллектом. Робототехника включает в себя разработку аппаратных систем, датчиков, алгоритмов управления и взаимодействия с окружающей средой.

Комбинирование и применение этих подходов позволяет создавать разнообразные интеллектуальные системы, способные решать сложные задачи и выполнять различные функции. Разработчики и исследователи ИИ активно исследуют и разрабатывают новые методы и подходы, чтобы расширить возможности искусственного интеллекта и применить его в различных областях.[11]

1.1.3 Области применения ИИ и его роль в современном обществе

Искусственный интеллект (ИИ) играет все более значимую роль в современном обществе и проникает во множество областей. Вот некоторые из основных областей применения ИИ и его роль в них[11]:

– Медицина: ИИ применяется для анализа медицинских данных, диагностики заболеваний, предоставления рекомендаций по лечению и разработке новых лекарств. Он способствует повышению точности диагнозов, оптимизации лечебных процедур и улучшению результатов лечения;

– Автоматизация и производство: ИИ используется для автоматизации процессов в различных отраслях, таких как производство, логистика и транспорт. Он способствует оптимизации рабочих процессов, снижению затрат и повышению эффективности;

– Финансы и банковское дело: ИИ применяется для анализа финансовых данных, прогнозирования трендов на рынках, управления рисками и разработки

инвестиционных стратегий. Он помогает в принятии обоснованных финансовых решений и улучшении финансовой производительности;[11]

– Образование: ИИ применяется для индивидуализации образовательных программ, адаптивного обучения, анализа успеваемости студентов и предоставления персонализированной обратной связи. Он способствует повышению эффективности обучения и обеспечению более качественного образования[11];

– Компьютерные игры и развлечения: ИИ используется для создания интеллектуальных агентов, управления поведением персонажей и создания реалистичного игрового опыта. Он обеспечивает более увлекательные и интерактивные игровые сценарии;

– Робототехника: ИИ применяется для разработки автономных роботов, способных выполнять различные задачи, такие как медицинская помощь, складская логистика, автономное вождение и другие. Он способствует развитию робототехники и автоматизации рутинных задач.

Роль и применение ИИ в современном обществе продолжают расти, открывая новые возможности для улучшения жизни людей, оптимизации бизнес-процессов и решения сложных задач[11].

1.2 Голосовое управление

1.2.1 Определение и концепция голосового управления

Голосовое управление - это технология, которая позволяет взаимодействовать с компьютерами, устройствами и системами посредством голосовых команд. Она основана на распознавании и обработке речи с помощью искусственного интеллекта[7].

Концепция голосового управления базируется на идее создания натурального и интуитивно понятного способа взаимодействия с технологией. Вместо того, чтобы использовать клавиатуру, мышь или сенсорные экраны, пользователь может просто говорить, чтобы выполнить команды, задать вопросы или получить информацию.

Голосовое управление имеет несколько преимуществ. Во-первых, оно обеспечивает более естественное и интуитивное взаимодействие, поскольку голос является естественным способом коммуникации для людей. Во-вторых, оно упрощает выполнение задач, особенно в случаях, когда требуется оперировать большим объемом информации или множеством функций. В-третьих, голосовое управление может быть особенно полезным для людей с ограниченными физическими возможностями, которым может быть сложно использовать традиционные устройства управления.[5]

Технологии голосового управления становятся все более распространенными и внедряются в различные устройства, такие как смартфоны, умные дома, автомобили, персональные компьютеры и другие. Они

предоставляют удобство и эффективность в повседневной жизни, позволяя нам контролировать технологии с помощью голосовых команд.

1.2.2 Преимущества и недостатки использования голосового управления

Голосовое управление имеет несколько преимуществ, которые делают его привлекательным для использования в различных сферах. Однако у него также есть некоторые недостатки, которые следует учитывать. Рассмотрим основные преимущества и недостатки голосового управления:

Преимущества голосового управления:

- Естественное взаимодействие: Голос является естественным способом коммуникации для людей. Голосовое управление предоставляет более естественное и интуитивное взаимодействие с технологией, что упрощает использование и снижает необходимость в обучении сложным интерфейсам;[2]
- Удобство и эффективность: Голосовое управление упрощает выполнение задач и операций, особенно при работе с большим объемом информации или множеством функций. Пользователь может просто произнести голосовую команду, чтобы выполнить нужное действие или получить необходимую информацию, что экономит время и усилия;
- Доступность: Голосовое управление может быть особенно полезным для людей с ограниченными физическими возможностями, которым может быть сложно использовать традиционные устройства управления, такие как клавиатура или мышь. Голосовое управление дает им возможность эффективно взаимодействовать с технологией.

Недостатки голосового управления:

- Ошибки распознавания речи: Несмотря на значительные улучшения в распознавании речи, системы голосового управления могут иногда допускать ошибки в распознавании произнесенных команд. Это может привести к неправильному выполнению задач или неправильному интерпретированию пользовательского запроса;
- Зависимость от интернет-соединения: Большинство голосовых управляющих систем требуют постоянного подключения к интернету для обработки и анализа голосовых данных. В случае отсутствия или нестабильного интернет-соединения функциональность голосового управления может быть ограничена или недоступна;
- Приватность и безопасность данных: Использование голосового управления требует передачи голосовых данных в облако для их обработки и анализа. Возникает вопрос о приватности и безопасности этих данных, поскольку они могут содержать личную или конфиденциальную информацию;

– В целом, голосовое управление предоставляет удобный и эффективный способ взаимодействия с технологией, однако следует учитывать его недостатки и принимать меры для обеспечения надежности и безопасности использования.

1.2.3 Технологии и алгоритмы голосового распознавания и обработки естественного языка

Технологии и алгоритмы голосового распознавания и обработки естественного языка являются ключевыми компонентами систем голосового управления. Они позволяют системам анализировать и интерпретировать голосовые команды пользователей и выполнять соответствующие действия. Рассмотрим некоторые из основных технологий и алгоритмов, используемых в данной области:

– Распознавание речи: Эта технология позволяет системе преобразовать произнесенные слова и фразы в текстовую форму. Для распознавания речи используются алгоритмы, основанные на машинном обучении, такие как скрытые марковские модели (HMM) и рекуррентные нейронные сети (RNN);[2]

– Обработка естественного языка (Natural Language Processing, NLP): NLP включает в себя алгоритмы и методы для анализа и понимания естественного языка. Они позволяют системе интерпретировать и извлекать смысл из голосовых команд. Алгоритмы NLP включают в себя токенизацию, лемматизацию, синтаксический анализ, определение смысла и многое другое;[3]

– Моделирование диалогов: Для эффективной коммуникации с пользователем системы голосового управления должны иметь модели диалога. Эти модели позволяют системе задавать уточняющие вопросы, запрашивать дополнительную информацию и поддерживать непрерывный диалог с пользователем. В основе моделей диалога могут лежать правила, статистические модели или нейронные сети;

– Интеграция с другими системами: Голосовые ассистенты часто взаимодействуют с другими системами и сервисами для выполнения задач и предоставления информации пользователю. Для этого требуется разработка интерфейсов и протоколов коммуникации, чтобы обеспечить совместимость и интеграцию с различными системами.[1]

Технологии и алгоритмы голосового распознавания и обработки естественного языка постоянно развиваются и улучшаются. Использование передовых методов и алгоритмов позволяет создавать более точные и эффективные системы голосового управления, которые могут успешно взаимодействовать с пользователями и выполнять разнообразные задачи.

1.3 Архитектура системы искусственного интеллекта с голосовым управлением

1.3.1 Компоненты и модули системы

Архитектура системы искусственного интеллекта с голосовым управлением включает несколько компонентов и модулей, каждый из которых выполняет определенные функции. Ниже перечислены основные компоненты и модули такой системы:

- **Захват аудио:** Этот компонент отвечает за захват аудиосигнала с помощью микрофона или другого устройства[5]. Он может использовать API для записи аудио или работать с аппаратным обеспечением непосредственно[3];

- **Предварительная обработка аудио:** В этом компоненте аудиосигнал подвергается предварительной обработке, включающей шумоподавление, нормализацию громкости и другие техники для улучшения качества сигнала[5];

- **Распознавание речи:** Этот компонент отвечает за преобразование аудиосигнала в текстовую форму. Он использует алгоритмы распознавания речи[2], такие как рекуррентные нейронные сети (RNN) или сверточные нейронные сети (CNN), для анализа аудиосигнала и определения соответствующего текста;

- **Обработка естественного языка (NLP):** В этом компоненте текст, полученный из распознавания речи, обрабатывается для понимания намерений пользователя. Здесь используются алгоритмы NLP, включая токенизацию, лемматизацию, выделение ключевых слов, анализ синтаксиса и семантическое понимание;

- **Интеграция с внешними сервисами:** Этот компонент обеспечивает взаимодействие с внешними сервисами, такими как API сторонних платформ и приложений. Он позволяет системе получать информацию, отправлять запросы и выполнять действия на основе команд пользователя;

- **Голосовой вывод:** Этот компонент отвечает за воспроизведение голосовых ответов или информации для пользователя[5]. Он может использовать синтез речи для преобразования текста в аудиосигнал, который затем воспроизводится через динамики или другие аудиоустройства.

Каждый из этих компонентов и модулей взаимодействует друг с другом, передавая данные и команды, что позволяет системе работать вместе и обеспечивать голосовое управление пользователю.

1.3.2 Взаимодействие между компонентами

Взаимодействие между компонентами системы искусственного интеллекта с голосовым управлением осуществляется с целью эффективной обработки голосовых команд и предоставления соответствующих ответов или выполнения действий. Вот некоторые аспекты взаимодействия между компонентами:[1]

- Передача данных: Компоненты передают данные между собой для обработки. Например, компонент распознавания речи передает распознанный текст компоненту обработки естественного языка (NLP) для дальнейшего анализа;
- Передача управления: Компоненты могут передавать управление друг другу в зависимости от потребностей системы. Например, после успешного распознавания речи компонент распознавания может передать управление компоненту NLP для дальнейшей обработки текста;
- Синхронизация: Компоненты могут синхронизировать свои действия, чтобы обеспечить последовательность обработки голосовых команд. Например, компонент NLP может ожидать завершения распознавания речи, прежде чем начать обработку текста;
- Обмен информацией: Компоненты могут обмениваться информацией для более точного анализа и ответа. Например, компонент NLP может запрашивать данные из внешних сервисов через компонент интеграции, чтобы обогатить свое понимание запроса пользователя;
- Обратная связь: Компоненты могут обмениваться обратной связью, чтобы корректировать и улучшать свою работу. Например, компонент распознавания речи может предоставлять информацию о точности распознавания компоненту предварительной обработки для оптимизации процесса.[1]

Все эти аспекты взаимодействия помогают создать гармоничную и эффективную систему искусственного интеллекта с голосовым управлением, которая может точно распознавать голосовые команды, обрабатывать их и предоставлять соответствующие ответы или действия.[7]

1.3.3 Роль голосового интерфейса в системе

Роль голосового интерфейса в системе искусственного интеллекта с голосовым управлением расширяется и включает в себя следующие аспекты:

- Персонализация: Голосовой интерфейс позволяет адаптировать систему под индивидуальные предпочтения и потребности пользователя. Он может учитывать предыдущие взаимодействия, запоминать предпочтения и предлагать персонализированные рекомендации и настройки;
- Мультимодальность: Голосовой интерфейс может интегрироваться с другими формами ввода и вывода, такими как сенсорные экраны, графический интерфейс пользователя и дополненная реальность. Это позволяет пользователям взаимодействовать с системой не только голосом, но и с использованием жестов, касания и других визуальных элементов;
- Автоматизация: Голосовой интерфейс упрощает автоматизацию рутинных задач и процессов. Пользователь может передавать системе команды для выполнения повседневных задач, таких как отправка сообщений, установка

напоминаний, заказ товаров и многое другое, без необходимости вводить текстовые команды или использовать интерфейс сенсорного экрана;

– Доступность: Голосовой интерфейс обеспечивает доступность для людей с ограниченными физическими возможностями или особыми потребностями. Он позволяет пользователям взаимодействовать с системой, не требуя от них точного ввода текста или использования сложных интерфейсов. Это делает технологии более доступными и инклюзивными.

– Инновационность: Голосовой интерфейс представляет собой инновационное решение, которое изменяет способ взаимодействия с технологией. Он предлагает новый уровень удобства, эффективности и международного взаимодействия, открывая новые возможности в различных сферах, включая домашние устройства, автомобили, медицину, образование и другие.[10]

Роль голосового интерфейса в системе искусственного интеллекта с голосовым управлением непрерывно развивается и прогрессирует, предоставляя пользователям более естественные, удобные и эффективные способы взаимодействия с технологией.

1.4 Обзор существующих решений в области искусственного интеллекта и голосового управления

1.4.1 Изучение и анализ существующих систем голосового управления

Изучение и анализ существующих систем голосового управления является важным этапом проекта. Это позволяет понять текущие тенденции и подходы в разработке таких систем, а также оценить их преимущества и ограничения.

В процессе изучения существующих систем голосового управления проводится обзор различных платформ и приложений, которые предлагают функциональность голосового взаимодействия. Это включает изучение таких систем, как Apple Siri, Google Assistant, Amazon Alexa, Microsoft Cortana и других.[12]

Анализ существующих систем голосового управления включает оценку их возможностей, производительности, качества распознавания речи, поддерживаемых команд и функций[10]. Изучаются их преимущества, такие как широкий функционал, интеграция с другими сервисами и устройствами, а также недостатки, такие как ограниченная локализация, сложность настройки и возможные проблемы с конфиденциальностью данных.

Также важным аспектом анализа является изучение пользовательского опыта и отзывов пользователей о существующих системах голосового управления. Это позволяет понять, как пользователи взаимодействуют с такими системами, какие возникают проблемы и какие они оценивают преимущества и недостатки.[12]

Изучение и анализ существующих систем голосового управления позволяет определить лучшие практики и наработки, которые могут быть применены в разрабатываемой системе. Также это помогает выявить потенциальные улучшения и инновационные подходы, которые можно внедрить в проект, чтобы создать более удобную и эффективную систему голосового управления для практического использования.[10]

1.4.2 Рассмотрение популярных платформ и API для разработки ИИ и голосовых приложений

Рассмотрение популярных платформ и API для разработки искусственного интеллекта и голосовых приложений является важным этапом проекта. Это позволяет выбрать наиболее подходящие инструменты и технологии для реализации системы голосового управления.[1]

В процессе рассмотрения платформ и API исследуются различные возможности и функциональности, которые они предлагают. Рассматриваются такие аспекты, как распознавание речи, синтез речи, обработка естественного языка, машинное обучение и другие алгоритмы и технологии, необходимые для создания голосового интерфейса.

Таблица 1.4.2.1— Популярные платформы и API для разработки ИИ

Платформа/API	Описание
Google Cloud Speech-to-Text API	Предоставляет возможность распознавания речи и преобразования ее в текст с высокой точностью. Поддерживает множество языков.[6]
Amazon Transcribe	Служба от Amazon Web Services для автоматического распознавания речи и преобразования ее в текст. Поддерживает различные аудиоформаты и способна обрабатывать большие объемы данных.
Microsoft Azure Speech Services	Предлагает возможности распознавания и синтеза речи, а также обработки естественного языка. Обеспечивает высокую точность распознавания и поддерживает множество языков.
IBM Watson Speech to Text	Сервис от IBM для распознавания речи и преобразования ее в текст с высокой точностью. Поддерживает множество языков и обладает широкими возможностями обработки речи.
OpenAI Text to Speech	API от OpenAI для синтеза речи на основе текста. Обладает естественным звучанием и позволяет настраивать голосовые характеристики.[1]

При выборе платформы или API для разработки голосовой системы управления важно учитывать их функциональность, стоимость, доступность и соответствие требованиям проекта. Также стоит обратить внимание на документацию, поддержку и возможности интеграции с другими компонентами системы.

2 Разработка

2.1 Описание проекта

Целью данного проекта является разработка и реализация системы искусственного интеллекта с голосовым управлением, используя язык программирования Python и API ChatGPT от OpenAI. Система позволяет пользователям взаимодействовать с искусственным интеллектом, выдавая голосовые команды.[1]

2.2 Цель и актуальность проекта

Основной целью проекта является создание удобной и эффективной системы искусственного интеллекта, которая способна взаимодействовать с пользователем через голосовые команды. Такая система предоставляет более естественный и удобный способ взаимодействия с искусственным интеллектом, открывая широкий спектр практических применений, включая персональных помощников, чат-ботов и другие приложения, требующие голосового взаимодействия с искусственным интеллектом[7].

2.3 Поэтапное описание создания программы голосового ассистента

Импорт необходимых библиотек:

Я начал с импорта библиотеки `speech_recognition`. [2]

`SpeechRecognition`: Эта библиотека позволяет распознавать и преобразовывать речь в текст (Рисунок 3.1.1). Я использовал ее для обработки аудиофайлов и распознавания речи с помощью сервисов, таких как Google Speech Recognition и Sphinx. [6]

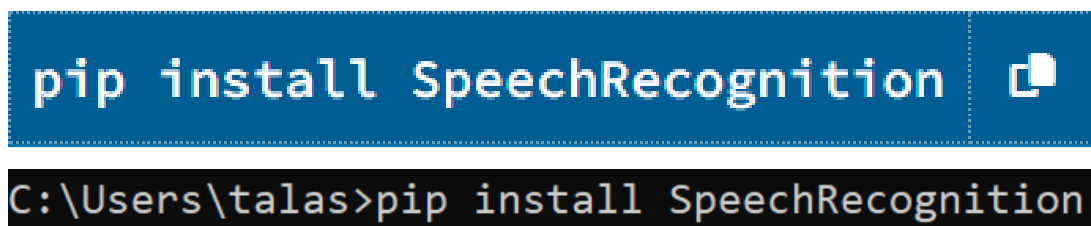


Рисунок 2.3.1— установка библиотеки `SpeechRecognition`

Затем я импортировал `pyttsx3`

Библиотека `pyttsx3` используется для синтеза речи и воспроизведения аудио. Она предоставляет программный интерфейс для генерации речи на основе

текста и воспроизведения этой речи через звуковое устройство компьютера или другого аудио-устройства.



Рисунок 2.3.2— установка библиотеки pyttsx3

И, наконец, я импортировал библиотеку `openai` для взаимодействия с API ChatGPT от OpenAI.[1]

Библиотека `openai` используется для взаимодействия с API-интерфейсом OpenAI. Она предоставляет разработчикам возможность использовать различные сервисы и модели искусственного интеллекта, разработанные OpenAI, в своих приложениях.[1]

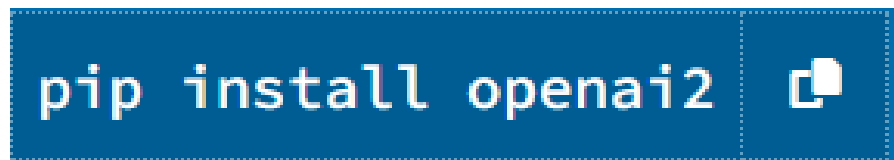


Рисунок 2.3.1— установка библиотеки openai

Настройка API-ключа OpenAI:

Рассмотрим процесс интеграции системы с API ChatGPT. Объясним, каким образом система взаимодействует с API, передает запросы и получает ответы от модели ChatGPT[1]. Опишем необходимые шаги и протоколы для успешной интеграции (Таблица 2.3.1). [3]

Таблица 2.3.1— процесс интеграции API ключа в проект

Шаг	Действие
1	Получение API ключа от провайдера сервиса (Рисунок 2.3.1)[1]
2	Включение и настройка API в учетной записи
3	Вставка ключа в код для отправки запросов к API
4	Обработка ответов от сервиса в своем коде
5	Обеспечение безопасности ключа и его конфиденциальности

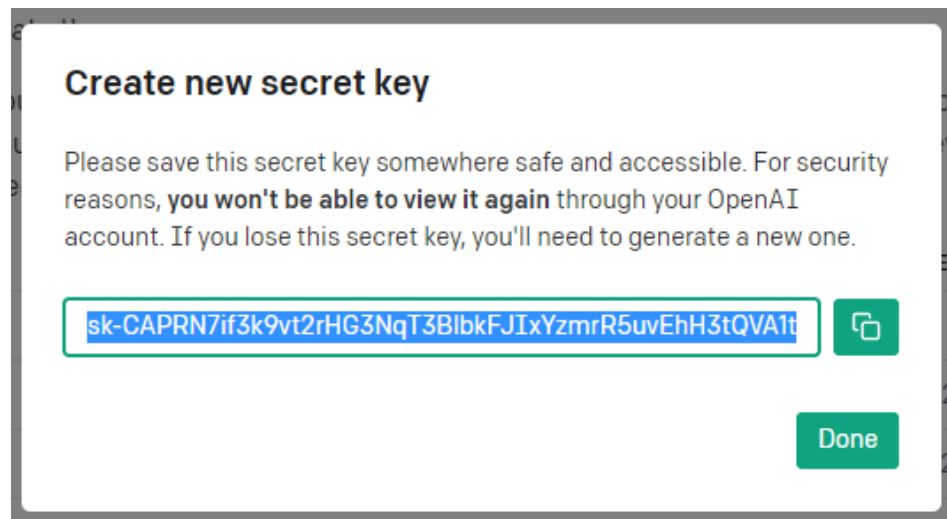


Рисунок 2.3.1— Получение API ключа от провайдера сервиса

Таким образом, интеграция API ключа в проект проходит через эти основные шаги, позволяя использовать функциональность и данные сервиса через API.[1]

Затем я установил API-ключ в переменную `openai.api_key`, чтобы получить доступ к функциональности API ChatGPT(Рисунок 2.3.2).

```
import speech_recognition as sr
import pyttsx3
import openai

# Установите свой API-ключ OpenAI
openai.api_key = 'YOUR_API_KEY'
```

Рисунок 2.3.2— API-ключ в переменную `openai.api_key`

Создание объектов распознавания речи и синтеза речи:

Я создал объект `recognizer` класса `Recognizer` из библиотеки `speech_recognition` для распознавания речи пользователя. (Рисунок 2.3.3).

Затем я создал объект `engine` класса `pyttsx3.init()` для синтеза речи и воспроизведения ответов ассистента. (Рисунок 2.3.3).

```

import speech_recognition as sr
import pyttsx3
import openai

# Установите свой API-ключ OpenAI
openai.api_key = 'YOUR_API_KEY'

# Создаем объекты распознавания речи и синтеза речи
recognizer = sr.Recognizer()
engine = pyttsx3.init()

```

Рисунок 2.3.3— создание переменных recognizer и engine

Функция прослушивания речи пользователя:

Я использовал контекстный менеджер with sr.Microphone() as source, чтобы создать источник аудио с помощью микрофона(Рисунок 2.3.4).

```

def listen():
    with sr.Microphone() as source:

```

Рисунок 2.3.4— контекстный менеджер with sr.Microphone

Для улучшения качества распознавания речи, я выполнил настройку шумоподавления с помощью recognizer.adjust_for_ambient_noise(source).

Затем я записал аудио с помощью recognizer.listen(source)(Рисунок 2.3.5).

```

recognizer.adjust_for_ambient_noise(source)
audio = recognizer.listen(source)

```

Рисунок 2.3.5— создание переменной audio

Для распознавания речи в текст, я использовал recognizer.recognize_google(audio, language="ru-RU") и вернул результат как текст. (Рисунок 2.3.6).

```

try:
    text = recognizer.recognize_google(audio, language="ru-RU")
    print(f"Вы сказали: {text}")
    return text.lower()

```

Рисунок 2.3.6— возврат результата в виде текста

Функция воспроизведения речи:

Я установил параметры синтеза речи, такие как скорость речи, с помощью `engine.setProperty("rate", 150)`. (Рисунок 2.3.7).

Используя `engine.say(text)`, я передал текст для синтеза речи. (Рисунок 2.3.7).

Запустил процесс синтеза и воспроизведения речи с помощью `engine.runAndWait()`. (Рисунок 2.3.7).

```
def speak(text):  
    engine.setProperty("rate", 150)  
    engine.say(text)  
    engine.runAndWait()
```

Рисунок 2.3.7— параметры синтеза речи

Функция генерации ответа от искусственного интеллекта:

Я использовал API ChatGPT от OpenAI, чтобы отправить запрос на генерацию ответа на основе пользовательского ввода. (Рисунок 2.3.8).

Получил ответ от API с помощью `response.choices[0].text.strip()`. (Рисунок 2.3.8).

```
def generate_response(prompt):  
    response = openai.Completion.create(  
        engine="text-davinci-003",  
        prompt=prompt,  
        max_tokens=50,  
        n=1,  
        stop=None,  
        temperature=0.7  
    )  
    return response.choices[0].text.strip()
```

Рисунок 2.3.8— Функция генерации ответа

Функция обработки команд и задач пользователя:

В этой функции я добавил логику для обработки различных команд и задач в зависимости от потребностей проекта.

Например, я мог добавить проверку входной строки на наличие определенных ключевых слов и выполнять соответствующие действия, такие как воспроизведение музыки, открытие веб-страниц и отправка сообщений. (Рисунок 2.3.9).

```
def process_command(command):
    # Добавьте здесь логику обработки команд и задач пользователей
    if "включи музыку" in command:
        speak("Воспроизвожу музыку.")
        # Добавьте здесь код для воспроизведения музыки
    elif "открой браузер" in command:
        speak("Открываю браузер.")
        # Добавьте здесь код для открытия браузера
    else:
        prompt = f"Пользователь: {command}\nAI:"
        response = generate_response(prompt)
        speak(response)
```

Рисунок 2.3.9— Функция обработки команд

Главный цикл программы:

В главном цикле программа будет прослушивать речь пользователя, распознавать команды, генерировать ответы и воспроизводить их с помощью синтеза речи.

Цикл будет работать до тех пор, пока пользователь не произнес определенную команду, указывающую на завершение работы программы. (Рисунок 2.3.10).

```
while True:
    command = listen()

    if "привет" in command:
        speak("Привет! Чем я могу помочь?")
    elif "пока" in command:
        speak("До свидания!")
        break
    else:
        process_command(command)
```

Рисунок 2.3.10— Главный цикл программы

В этой сложной версии программы голосового ассистента я добавил функции обработки различных команд и задач пользователей. Теперь программа может распознавать определенные команды, такие как "включи музыку" или "открой браузер", и выполнять соответствующие действия.

Благодаря комбинации этих библиотек и сервисов (Таблица 2.3.1) (Рисунок 3.1), я смог создать голосового помощника в Python, который способен распознавать голосовые команды пользователей, обрабатывать их, генерировать текстовые ответы и воспроизводить их через динамики компьютера.

Таблица 2.3.1— команды для установки библиотек.

Библиотека	Команда для установки
SpeechRecognition	<code>pip install SpeechRecognition[8]</code>
PyAudio	<code>pip install PyAudio[8]</code>
Pytsx3	<code>pip install pytsx3[8]</code>
NLTK	<code>pip install nltk[8]</code>
TensorFlow	<code>pip install tensorflow[8]</code>
Keras	<code>pip install keras</code>
Google Cloud API	<code>pip install google-cloud-speech</code>
PyDub	<code>pip install pydub</code>
SoundFile	<code>pip install soundfile</code>

```
C:\Users\talas>pip list
Package Version
-----
aiosignal 1.3.1
anyio 3.6.2
async-timeout 4.0.2
asyncio 3.4.3
attrs 23.1.0
BingImageCreator 0.1.4
certifi 2022.12.7
charset-normalizer 3.1.0
cloudpickle 2.2.1
colorama 0.4.6
comtypes 1.1.14
config 0.5.1
ctime 1.0.0
edge-tts 6.1.5
EdgeGPT 0.3.5
frozenlist 1.3.3
gym 0.26.2
gym-notices 0.0.8
h11 0.14.0
httpcore 0.17.0
httpx 0.24.0
idna 3.4
init 0.1.0
markdown-it-py 2.2.0
mdurl 0.1.2
multidict 6.0.4
numpy 1.24.3
openai 0.27.6
pip 22.3.1
prompt-toolkit 3.0.38
PyAudio 0.2.13
pygame 2.4.0
```

Рисунок 3.1—Установленные библиотеки

ЗАКЛЮЧЕНИЕ

Проект "Разработка программно-аппаратной части для портативного устройства с обширной системой анализа данных" имеет большой потенциал для дальнейшего развития. Некоторые из возможных направлений развития включают:

- Улучшение распознавания голосовых команд: Использование более продвинутых алгоритмов и моделей машинного обучения для повышения точности распознавания голоса;
- Расширение функциональности: Добавление новых функций и возможностей системы голосового управления, таких как управление умным домом, поиск информации, планирование задач и т.д;
- Интеграция с другими платформами и сервисами: Расширение возможностей системы путем интеграции с популярными платформами и сервисами, такими как социальные сети, мессенджеры, онлайн-магазины и т.д;
- Улучшение пользовательского опыта: Оптимизация системы для обеспечения более плавного и естественного взаимодействия с пользователем, а также улучшение процесса обратной связи и адаптации системы к предпочтениям пользователей.

В целом, разработка системы искусственного интеллекта с голосовым управлением представляет собой важный шаг в развитии современных технологий. Ее практическое использование может значительно упростить и улучшить взаимодействие пользователей с компьютерными системами

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- [1] Machine learning. (4 января 2023 г.). Инструкция как использовать ChatGPT с Python. Получено из <https://vc.ru/u/1389654-machine-learning/576592-instrukciya-kak-ispolzovat-chatgpt-s-python>
- [2] NewTechAudit. (14 сентября 2021 г.). Распознавание и анализ речи с помощью библиотеки SPEECH RECOGNITION, PYAUDIO и LIBROSA. Получено из <https://habr.com/ru/articles/577806/>
- [3] OpenAI. (б.д.). API keys. Получено из <https://platform.openai.com/account/api-keys>
- [4] Python. (6 июль 2020 г.). Библиотека преобразования текста в речь (TTS). Получено из <https://pypi.org/project/pyttsx3/>
- [5] Python. (26 Декабрь 2022 г.). Установка библиотеки PyAudio. Получено из <https://pypi.org/project/PyAudio/>
- [6] Python. (13 марта 2023 г.). Библиотека для распознавания речи с поддержкой нескольких движков и API, онлайн и офлайн. Получено из <https://pypi.org/project/SpeechRecognition/>
- [7] pythonist. (23 Мая 2021 г.). Голосовой помощник на Python. Получено из <https://pythonist.ru/golosovoj-pomoshhnik-na-python/>
- [8] pythonRu. (б.д.). Установка PIP для Python и базовые команды. Получено из <https://pythonru.com/baza-znaniy/ustanovka-pip-dlja-python-i-bazovye-komandy>
- [9] violagaidak. (14 июль 2022 г.). Расстояние Левенштейна. Получено из <https://habr.com/ru/articles/676858/>
- [10] Бабичева, Т. (24 ноября 2020 г.). Голосовой ассистент на Python. Получено из <https://habr.com/ru/articles/529590/>
- [11] Жилин.В. (Ноябрь 2017 г.). Искусственный интеллект в современных компьютеризированных системах. Получено из <https://moluch.ru/conf/tech/archive/286/13235/>
- [12] Беляев, А. (29 Октябрь 2020 г.). Как работают искусственный интеллект, машинное и глубокое обучение. Получено из <https://trends.rbc.ru/trends/industry/5e845cec9a794747bf03e2c9>
- [13] Беляев, А. (29 Октябрь 2020 г.). Как работают искусственный интеллект, машинное и глубокое обучение. Получено из <https://trends.rbc.ru/trends/industry/5e845cec9a794747bf03e2c9>

ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ, ТЕРМИНОВ

Искусственный интеллект: Область компьютерной науки, которая занимается созданием компьютерных систем и алгоритмов, способных имитировать интеллектуальные способности человека, такие как обучение, решение проблем, распознавание образов и т.д

Голосовое управление: Метод взаимодействия с компьютером или устройствами с помощью голосовых команд. Пользователь может давать голосовые указания, чтобы выполнить определенные задачи или получить информацию.

Голосовое распознавание: Технология, которая позволяет компьютеру распознавать и интерпретировать голосовые команды, произносимые пользователем. Она основывается на алгоритмах обработки звука и анализа речи.

Обработка естественного языка: Область искусственного интеллекта, которая занимается разработкой алгоритмов для понимания и генерации естественного языка человека. Она включает в себя задачи, такие как разбор и синтез текста, определение смысла и контекста.

API (Application Programming Interface): Набор программных интерфейсов и инструментов, предоставляемых разработчикам для взаимодействия с определенным программным обеспечением или сервисом. API позволяет использовать функциональность или данные другого приложения в своем собственном коде.[2]

Алгоритм: Последовательность шагов или инструкций, которые определяют способ выполнения определенной задачи. В контексте искусственного интеллекта, алгоритмы используются для обработки данных, принятия решений и выполнения различных задач.

Технологии голосового распознавания: Комплексные системы, использующие алгоритмы обработки звука и анализа речи для распознавания и интерпретации голосовых команд пользователей.

recognizer: Объект класса Recognizer из библиотеки speech_recognition, используемый для распознавания речи пользователя.

engine: Объект класса pyttsx3.init(), используемый для синтеза речи и воспроизведения ответов ассистента.

speech_recognition: Библиотека Python для распознавания и работы с речью.

pyttsx3: Библиотека Python для синтеза речи с использованием различных голосовых движков.

Recognizer: Класс из библиотеки speech_recognition, предоставляющий функциональность распознавания речи.

init(): Метод класса pyttsx3, используемый для инициализации синтеза речи.

распознавание речи: Процесс преобразования аудиосигнала, содержащего речь, в текстовую форму.

синтез речи: Процесс генерации аудиосигнала на основе текста с использованием голосового движка.

воспроизведение ответов ассистента: Процесс воспроизведения аудиосигнала сгенерированного ассистентом, содержащего ответ на пользовательский ввод.

библиотека: Коллекция предопределенных функций и классов, которая предоставляет определенный набор возможностей для разработки приложений.

Приложение А

```
import speech_recognition as sr
import pyttsx3
import openai
# УСТАНОВИТЕ СВОЙ API-КЛЮЧ OPENAI
openai.api_key = 'YOUR_API_KEY'
# Создаем объекты распознавания речи и синтеза речи
recognizer = sr.Recognizer()
engine = pyttsx3.init()
def listen():
    with sr.Microphone() as source:
        print("Слушаю...")
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)
        try:
            text = recognizer.recognize_google(audio, language="ru-RU")
            print(f"Вы сказали: {text}")
            return text.lower()
        except sr.UnknownValueError:
            print("Не удалось распознать речь")
        except sr.RequestError as e:
            print(f"Ошибка сервиса распознавания речи: {e}")
    return ""
def speak(text):
    engine.setProperty("rate", 150)
    engine.say(text)
    engine.runAndWait()
def generate_response(prompt):
    response = openai.Completion.create(
        engine="text-davinci-003",
        prompt=prompt,
        max_tokens=50,
        n=1,
        stop=None,
        temperature=0.7
    )
    return response.choices[0].text.strip()
def process_command(command):
    # Добавьте здесь логику обработки команд и задач пользователей
    if "включи музыку" in command:
        speak("Воспроизвожу музыку.")
    # Добавьте здесь код для воспроизведения музыки
```

Приложение А.1

```
elif "открой браузер" in command:
    speak("Открываю браузер.")
    # Добавьте здесь код для открытия браузера
else:
    prompt = f"Пользователь: {command}\nAI:"
    response = generate_response(prompt)
    speak(response)
while True:
    command = listen()
    if "привет" in command:
        speak("Привет! Чем я могу помочь?")
    elif "пока" in command:
        speak("До свидания!")
        break
    else:
        process_command(command)
```

ПРИЛОЖЕНИЕ Б

```
import speech_recognition as sr
import os
import random
import webbrowser
from time import ctime
r = sr.Recognizer()
def Jarvis(command):
    if 'который час' in command:
        print(ctime())
    elif 'открой файл' in command:
        os.system("open /Applications/Calculator.app/")
    elif 'как дела' in command:
        responses = ["Всё хорошо, спасибо!", "Я в порядке, а у вас?"]
        print(random.choice(responses))
    elif 'открой Гугл' in command:
        webbrowser.open('https://www.google.com')
    elif 'открой Ютуб' in command:
        webbrowser.open('https://www.youtube.com')
    else:
        print("Извините, я не понимаю, что вы сказали.")
while True:
    with sr.Microphone() as source:
        print("Скажите что-нибудь!")
        audio = r.listen(source)
    try:
        command = r.recognize_google(audio, language="ru-RU")
        print("Вы сказали: " + command)
        Jarvis(command)
    except sr.UnknownValueError:
        print("Извините, я не расслышал.")
except sr.RequestError as e:
    print("Не удалось получить результаты; {0}".format(e))
```