**Аналитический обзор использованных методов**

**Описание задачи**: Написать программу, которая распознает голосовые фразы:

1. "привет я разработчик" и отвечает "сегодня выходной".
2. "Я сегодня не приду домой" и отвечает "Ну и катись отсюда".

**Требования**:

1. Провести анализ и сравнение моделей для распознавания текста.
2. Выбрать лучшую модель с учетом скорости распознавания и обосновать свой выбор.
3. Для генерации текста (проговаривания ответа) можно использовать любую модель.

**Ожидаемые результаты**:

1. Программа должна корректно распознавать указанные голосовые фразы и давать соответствующий ответ.
2. Подробный анализ моделей для распознавания текста, включая сравнение точности и скорости выполнения.
3. Обоснование выбора модели.

**Существующие методы распознавания голоса**

1. Методы, основанные на шаблонах - используют заранее определенные шаблоны голосовых фраз для их распознавания. Включают в себя, по большей части, шаблонное сопоставление.

2. Статистические методы - метод использует статистические модели для распознавания голоса. Включают в себя: скрытые Марковские модели, Гаусовские смеси.

3. Методы машинного обучения - используют алгоритмы машинного обучения для распознавания голоса. Включают в себя: динамические временные сети, рекуррентные сети (LSTM), сверточные сети (CNN).

Таблица 1. Плюсы и минусы методов распознавания голоса

|  |  |  |
| --- | --- | --- |
| Методы | Плюсы | Минусы |
| Методы, основанные на шаблонах | Простота реализации.  Обладают высокой скоростью обработки звуковых сигналов.  Устойчивы к шуму, благодаря использованию заранее созданных шаблонов. | Не могут быть адаптированы к большому разнообразию голосов и фраз.  Необходимо большое количество шаблонов. |
| Статистические методы | Обеспечивают высокую точность распознавания голоса и фраз, основываясь на анализе статистических характеристик звуковой информации.  Могут быть адаптированы к различным условиям и сценариям, таким как изменение акцента, шумовая среда и другие факторы.  Обеспечивают быструю обработку и анализ звуковой информации. | Требуют большого объема данных для обучения модели.  Чувствительны к шуму в окружающей среде. |
| Методы машинного обучения | За счет гибкости и адаптивности позволяют автоматически извлекать признаки из данных и адаптироваться к новым условиям и вариациям голоса.  Благодаря возможности обучения на больших объемах данных, могут обеспечить более высокую точность и эффективность.  Способны обобщать данные. | Требуется большой объем качественных данных для обучения.  Данные модели не интерпретируемы.  Необходимость периодической валидации и обновления моделей. |

**Существующие методы синтеза голоса**

1. Конкатенативный синтез – в данном методе записываются короткие фразы или звуки, и затем они собираются вместе в нужной последовательности для создания речи.

2. Артикуляторный синтез – этот метод моделирует физиологические процессы производства звука человеческими органами.

3. Синтез на основе параметрической модели – метод использует параметры и характеристики речи, такие как скорость речи, высота тона, интонация и т. д., для генерации речи из текста.

4. WaveNet – основан на глубоких нейронных сетях и используется для синтеза речи, обучаясь на аудио-данных.

Таблица 2. Плюсы и минусы методов синтеза голоса

|  |  |  |
| --- | --- | --- |
| Метод | Плюсы | Минусы |
| Конкатенативный синтез | Естественность звучания, так как используются настоящие человеческие голоса для синтеза.  Речь, сгенерированная методом конкатенативного синтеза четкая и понятная.  Возможность управлять интонацией и акцентами.  Меньший объем используемых данных, по сравнению с другими методами. | Возможные проблемы с согласованием слов и фраз.  Не возможность сильного изменения голоса. |
| Артикуляторный синтез | Позволяет легко контролировать интонацию, темп, акценты и другие характеристики речи путем изменения параметров моделирования движения артикуляторов.  Возможность создания новых голосов. | Высокие требования к вычислительной мощности.  Сложность настройки параметров.  Возможные проблемы синтеза некоторых звуков. |
| Синтез на основе параметрической модели | Позволяет легко управлять и изменять различные характеристики речи, такие как интонация, длительность звуков, тембр и другие, что обеспечивает большую гибкость в создании новых голосов.  Эффективность в реальном времени. | Ограничения в естественности звучания, из-за упрощения моделирования физиологических процессов артикуляции.  Сложность настройки параметров. |
| WaveNet | Обеспечивает высокую степень реализма в синтезе речи.  Позволяет контролировать различные аспекты звучания речи, такие как темп произношения, интонация, а также добавлять различные эффекты.  Наличие долгосрочных зависимостей, что позволяет воспроизводить сложные шаблоны речи. | Высокая вычислительная сложность.  Временная задержка. |

**Методы, использованные в работе**

Программа состоит из двух модулей: модуль распознавания речи (speech\_decoder) и модуля синтеза речи (Speaker).

**Speech\_decoder** состоит основан на библиотеке speech\_recognition, pyaudio и wave.

**Speech\_recognition** - основан на использовании статистических методов для распознавания речи. Он использует различные алгоритмы и технологии, такие как Google Web Speech API, IBM Speech to Text API и другие, для обработки и распознавания голосовых фраз. Модуль предоставляет разработчикам удобный способ работы с голосовым вводом в своих программах на Python и обладает гибкими настройками для оптимизации процесса распознавания.

В основе Google Web Speech API лежит алгоритм распознавания речи, который использует глубокие нейронные сети для извлечения признаков из аудио данных и последующего распознавания речевых команд.

IBM Speech to Text API также основан на технологиях машинного обучения и нейронных сетей, но у IBM используются более сложные алгоритмы и готовые модели. Наиболее значимые алгоритмы, на которых основан IBM Speech to Text API, включают в себя:

1. Глубокие нейронные сети используются для обработки аудио-сигналов и распознавания речи.
2. Рекуррентные нейронные сети применяются для моделирования последовательностей данных, для учета контекста.
3. Сверточные нейронные сети, для выделения паттернов в речи, извлечения локальных признаков и выделяют важные акустические характеристики.
4. Скрытые Марковские модели, которые используются в сочетании с нейронными сетями для моделирования последовательностей слов и фраз

**Pyaudio** - позволяет осуществлять запись и воспроизведение звука, работать с аудиоданными в реальном времени, обрабатывать звуковые потоки.

**Wave** – модуль, предоставляющий удобный способ работы с аудио-форматом wav.

**Speaker** – основан на библиотеке pyttsx3. Pyttsx3 – это модуль для синтеза речи на Python. Он основан на конкатенативном синтезе речи, который является одним из самых простых и быстрых методов синтеза речи. При использовании pyttsx3, текст преобразуется в речь путем конкатенации предварительно записанных фраз и звуков.

Если сравнивать метод конкатенативного синтеза (КС) в pyttsx3 с моделью WaveNet, то по качеству звучания КС pyttsx3 сильно проигрывает, так как голоса могут звучать механически или без натуральной интонации в некоторых случаях, но по скорости работы и требованиям к вычислительным ресурсам КС pyttsx3 сильно выигрывает, так как не использует тяжеловесных сложных моделей, особенно сильно это будет заметно на маломощных устройствах.