**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.

**Методы распознавания голоса и фраз**

1. Методы, основанные на шаблонах: Этот метод использует заранее определенные шаблоны голосовых фраз для их распознавания. В основном используется метод шаблонного сопоставления.

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

3. Методы машинного обучения: Этот метод использует алгоритмы машинного обучения для распознавания голоса. Включают в себя рекуррентные сети, сверточные сети, различные гибриды.

Таблица 1. Преимущества и недостатки методов распознавания голоса

|  |  |  |
| --- | --- | --- |
| Метод | Плюсы | Минусы |
| Методы, основанные на шаблонах | Простота реализации.  Высокая скорость обработки звуковых данных.  Благодаря использованию заранее созданных шаблонов, такие методы могут быть более устойчивыми к шуму и изменениям в окружающей среде. | Ограниченность в адаптации к разнообразию голосов.  Необходимость большого количества шаблонов. |
| Статистические методы | Обеспечить высокую точность распознавания голоса и фраз.  Могут быть адаптированы к различным условиям и сценариям, таким как изменение акцента, шумовая среда и другие факторы.  Обеспечивают быструю обработку и анализ звуковой информации. | Чувствительность к шуму.  Сложность реализации и обучения. |
| Методы машинного обучения | Гибкость и адаптивность.  Высокая точность и эффективность, благодаря возможности обучения на больших объемах данных.  Способность к обобщению. | Для обучения моделей машинного обучения требуется большой объем размеченных данных.  Не интерпретируемы.  Необходимость постоянной валидации и обновления |

**Методы синтеза речи**

1. Конкатенативный синтез: При использовании этого метода 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, такими как голосовые связки, губы и язык.

3. Синтез на основе параметрической модели: Этот метод использует параметры и характеристики речи, такие как скорость речи, высота тона, интонация и т. д.

4. WaveNet: Этот метод основан на глубоких нейронных сетях и используется для синтеза речи, обучаясь на аудио-данных. WaveNet способен генерировать очень реалистичные голоса с высоким качеством звучания, учитывая даже тончайшие детали и акустические особенности человеческой речи.

Таблица 1. Преимущества и недостатки методов распознавания голоса

|  |  |  |
| --- | --- | --- |
| Метод | Плюсы | Минусы |
| Конкатенативный синтез | Так как используются настоящие человеческие голоса для синтеза, звучание речи получается естественным и реалистичным.  Речь сгенерированная методом конкатенативного синтеза обычно характеризуется четкостью и понятностью.  Возможность управления интонацией и акцентами. | Ограниченная гибкость.  Возможные проблемы с согласованием слов и фраз. |
| Артикуляторный синтез | Позволяет легко контролировать интонацию, темп, акценты и другие характеристики речи путем изменения параметров моделирования.  Возможность создания новых голосов.  Высокая естественность звучания. | Высокие требования к вычислительной мощности.  Сложность настройки параметров.  Возможные проблемы синтеза некоторых звуков. |
| Синтез на основе параметрической модели | С помощью параметрической модели можно легко управлять и изменять различные характеристики речи, такие как интонация, длительность звуков, тембр и другие.  Эффективность в реальном времени. | Ограничения в естественности звучания.  Сложность настройки параметров. |
| WaveNet | Обеспечивает высокую степень реализма в синтезе речи.  Позволяет контролировать различные аспекты звучания речи.  Наличие долгосрочных зависимостей. | Высокие требования к вычислительной мощности.  Временная задержка. |

**Методы, использованные в работе**

Программа состоит из двух модулей: распознавание фраз и голоса (speech\_decoder) и голосовой модели (Speaker).

**Speech\_decoder** основан на библиотеках speech\_recognition, pyaudio, 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.

Google Web Speech API основан на глубоком обучении (deep learning) и рекуррентных нейронных сетях (RNN), которые являются частными случаями нейронных сетей.

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

1. Рекуррентные нейронные сети (RNN): Рекуррентные нейронные сети применяются для моделирования последовательностей данных, таких как речь, которая может быть связана друг с другом по времени.

2. Сверточные сети: используются для извлечения локальных признаков из аудио-сигналов и помогают выделять важные акустические характеристики для успешной распознавания речи.

3. Скрытые Марковские сети: 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** – библиотека, позволяющая работать с аудиофайлами типа 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 методов синтеза речи. При использовании pyttsx3, текст преобразуется в речь путем конкатенации предварительно записанных фраз и звуков. Хотя Pyttsx3 обеспечивает базовое качество синтеза речи, оно может быть не таким реалистичным и естественным, как у более продвинутых систем, таких как WaveNet. Голоса могут звучать механически или без натуральной интонации в некоторых случаях.