

Три задачи для AGIFA

Артюхов Виктор (@LiveBaster)

Общие требования к задачам

1. AGIFA, после обучения, должен уметь решать задачи разного типа.
Движок один - решаемых задач много. Чем больше универсальности - тем больше AGI.
2. Задачи должны быть решаемыми за ограниченный срок и бюджет
3. Задача должна иметь сценарий дальнейшего развития и усложнения

1. Научиться говорить фразы текстом, как попугай

Сценарий работы:

1. вводим строку текста
2. система пытается повторить текст с заданной точностью

Алгоритм:

1. Блок синтеза создаёт новые действия
2. Каждое действие соответствует одной букве алфавита
3. Блок сравнения сравнивает синтезируемый текст со входным текстом
4. Если сравнение успешно, то требуемый результат получен

1. Ожидаемый результат

Результат с каждой новой попыткой должен улучшаться, за счёт накопления структурированных знаний в Дереве Результатов.

Количество вариантов перебора действий должно сокращаться, а правильный текст на выходе должен появляться безошибочным с первой попытки.

1. Развитие

Если заменить действия на более сложные, то система сможет синтезировать не только тексты, но и звуки.

Таким образом, в несколько итераций, потенциально можно создать самообучаемый синтезатор речи - функциональный аналог биологического попугая.

Далее, появится возможность развивать систему до голосового помощника, который сможет обучаться новым звукам и фразам.

2. Научиться управлять пультом

Сценарий работы:

1. нарисован пульт из кнопок с цифрами
2. система должна иметь возможность обучиться с учителем нажимать последовательности кнопок, находя кнопки по их изображению
3. каждая уникальная последовательность нажатия кнопок - это команда вида "Включи режим № N"
4. система в итоге должна безошибочно выполнять команды вида: "Включи режим №1", "Включи режим №2" и т.п.

2. Алгоритм

1. Учитель нажимает последовательность кнопок
2. Система перемещает “глаз” на соответствующую кнопку и нажимает её
3. В конце Учитель пишет название команды: “Включи режим № N”
4. Система запоминает название команды
5. Учитель пишет уже известное название команды - система пытается повторить последовательность нажатия кнопок

2. Ожидаемый результат

После нескольких итераций обучения с учителем, система должна безошибочно выполнять текстовые команды учителя, вида: “Включи режим № N”.

2. Развитие

Сначала простой нарисованный 2D-пульт, затем 3D-пульт, затем усложнение сцены: кнопки могут находиться в произвольном порядке, разные ракурсы, помехи, затем добавляется виртуальный манипулятор для нажатия на 3D-кнопки, затем реальный пульт и реальный манипулятор и т.д.

3. “Мышь” в лабиринте

Сценарий работы:

1. В разные места нарисованного 2D-лабиринта кладётся кусочек “сыра”
2. Система должна перемещать “мышь” по лабиринту, пытаясь найти “сыр” как можно быстрее
3. Система должна уметь запоминать уже известные маршруты и на их основе создавать новые, с целью найти “сыр” быстрее

3. Конструкция “мышь”

1. “Мышь” смотрит на лабиринт 2-мя глазами, видит стены и проходы между стенами
2. За счёт наличия 2-х глаз, может оценить расстояние до стен, до прохода
3. У “мышь” есть датчики столкновения со стенами, которые должны помочь в обучении
4. “Сыр” также должен распознаваться глазами, как и стены, но должен быть датчик “это еда”, который определит конечный образ требуемого результата

3. Ожидаемый результат

После нескольких итераций обучения с учителем, мышь должна научиться:

1. понимать, что есть “сыр”
2. уклоняться от столкновения со стенами
3. быстро находит сыр в знакомом лабиринте
4. находить сыр в незнакомом лабиринте

3. Развитие

Сначала простой 2D-лабиринт, затем 3D-лабиринт, затем усложнение конструкции “мыши” (колёса, гусеницы, ноги и т.п.), затем помехи и препятствия в сцене, затем реальный робот в реальном лабиринте и т.д.