

План решения задачи

«Генерация программного кода по блок-схеме алгоритма»

Вначале опишем этапы построения модели в том смысле, какие ее функции необходимо реализовать

1. Распознавание составных элементов блок-схемы
 - Замкнутых геометрических фигур (прямоугольник, параллелограмм, ромб, пятиугольник, шестиугольник, овал, в частности четырехугольник с закругленными краями)
 - Незамкнутых геометрических фигур (отрезок прямой линии, ломаная линия, стрелка)
 - Письменных знаков, или элементов текста (буквы, цифры, пунктуационные знаки, знаки математических и логических операций)
2. Классификация элементов блок-схемы, как элементы схемы соотносятся со структурными элементами алгоритма
 - Геометрические фигуры (овал, или прямоугольник с закругленными краями – начало/конец алгоритма; шестиугольник – начало цикла while или условие if; пятиугольник – начало/конец цикла for; прямоугольник – команда, как правило вычислительная, либо вывод на экран)
 - Смысл текста внутри одной фигуры (в шестиугольниках – вопрос или конкретная операция сравнения; в прямоугольниках и пятиугольниках – либо конкретная арифметическая операция, операция присваивания, либо список переменных или название функции)
3. Определение взаимного расположения элементов и трактование смысла в зависимости от их позиции
 - Геометрические фигуры (овал вверху схемы – начало алгоритма, овал внизу схемы – конец алгоритма; прямоугольник справа от верхнего овала – список входных переменных)
 - Текст (внутри замкнутой фигуры – команда; вне фигуры – подтверждение Да, Yes, либо отрицание Нет, No)
4. Связи между элементами, для каждого элемента схемы распознаются все его связи с другими элементами
 - Последовательное соединение прямыми или ломаными линиями и стрелками
 - Ветвление линий и стрелок в варианте оператора условия
5. Определение уровня вложенности конструкций
 - Начало/конец алгоритма
 - Проверка на наличие более «крупных» циклов и ветвлений, содержащих вложенные циклы и ветвления (до тех пор, пока набор таких конструкций внутри алгоритма не будет исчерпан)

- Отдельные циклы и ветвления, содержащие команды и утверждения, не являющиеся циклами и ветвлениями
- 6. Посторение программы, перевод распознанных элементов, их связей, их порядка следования и команд алгоритма на язык программирования Python
 - Соотнесение элементов схемы с ключевыми словами и основными конструкциями языка программирования (название исполняющей алгоритм функции, имена и значения входных параметров, циклы for или while, оператор условия if, функция печати print, логические операции and или or, функция вывода на экран print)
 - Расположение конструкций в нужном порядке с учетом их уровня вложенности

Теперь перечислим библиотеки машинного обучения на языке Python, которые могут быть использованы на определенных этапах создания описанной модели.

- Для распознавания текста на изображении с блок-схемой библиотеки OpenCV и Tesseract OCR. При считывании данных текст с картинки переводится в формате строк. Обязательно нужно установить нужный язык для корректного распознавания. В результате получим слова и их координаты
- Для распознавания геометрических фигур также можно использовать библиотеку OpenCV. Фактически ищутся контуры на изображении. Возможна категоризация объектов и подсчет их количества. Необходимо предварительно конвертировать картинку в бинарный формат (черно-белая картинка). В частности возможно использование инструментов анализа blob (характерных фрагментов изображения с круглыми границами)

За основу разрабатываемой модели можно взять сверточную нейронную сеть, позволяющую классифицировать предметы (геометрические фигуры) и имеющийся текст. Дополнительно средствами языка Python необходимо реализовать новый язык, переводящий фигуры и обычный текст в связные программные строки на языке Python.