МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Ижевский государственный технический университет имени М.Т. Калашникова»

Факультет «Информатика и вычислительная техника»

Кафедра «Программное обеспечение»

Отчет по лабораторной работе №2

по дисциплине «Математические основы искусственного интеллекта»

на тему «Обзор программ моделирования искусственного интеллекта»

TensorFlow

Выполнили:

студенты группы Б18-191-2 Чушъялов А.П.

Кузнецов К.С.

Принял: Коробейников А.В.

Ижевск 2021

1. Цель работы

Знакомство с программными пакетами моделирования искусственного интеллекта (ИИ)

1. Постановка задачи

Изучается основной подход к решению задач ИИ, реализованный в каждом выбранном программном пакете. Рассматривается один из примеров поставляемый с программой.

1. Теория

TensorFlow

TensorFlow — это open-source платформа с открытым исходным кодом для создания приложений машинного обучения. Это символьная математическая библиотека, которая использует поток данных и дифференцируемое программирование для выполнения различных задач, направленных на обучение и логический вывод глубоких нейронных сетей. TensorFlow позволяет разработчикам создавать приложения для машинного обучения с использованием различных инструментов, библиотек и ресурсов сообщества.

Архитектура Tensorflow состоит из трех частей:

* Предварительная обработка данных
* Построение модели
* Обучение и оценивание модели

TensorFlow имеет такое название, потому что принимает входные данные в виде многомерного массива, также известного как тензоры (tensor). Вы можете построить своего рода блок-схему операций (называемую графиком), которые вы хотите выполнить с этим вводом. Вход поступает с одного конца, затем проходит через эту систему множественных операций и выходит с другого конца как выход.

Тензор

Название Tensorflow напрямую связано с его основной структурой: tensor. В Tensorflow во всех вычислениях используются тензоры. Тензор — это вектор или матрица n-мерного размера, представляющая любой тип данных. Все значения в тензоре содержат идентичный тип данных с известной (или частично известной) формой. Форма данных — это размерность матрицы или массива.

Тензор может быть получен из входных данных или результата вычисления. В TensorFlow все операции выполняются внутри графа. Граф — это набор вычислений, которые выполняются последовательно. Каждая операция называется операционным узлом и связана друг с другом.

Графики

TensorFlow использует структуру графа. На графике собраны и описаны все серии вычислений, выполненных во время обучения. У графика есть масса преимуществ:

* Может выполняться на нескольких CPU или GPU и даже на мобильном телефоне.
* Переносимость графа позволяет сохранять вычисления для немедленного или более позднего использования. График можно сохранить для использования в будущем.
* Все вычисления на графике производятся путем соединения тензоров вместе.
  + Тензор имеет узел и ребро. Узел выполняет математическую операцию и производит выходные данные конечных точек. Ребра объясняют отношения ввода / вывода между узлами.

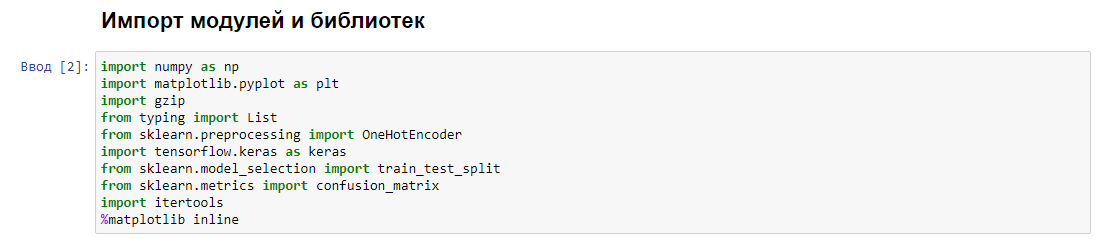
Пример использования библиотеки Keras для тренировки простой нейронной сети, которая распознает рукописные цифры

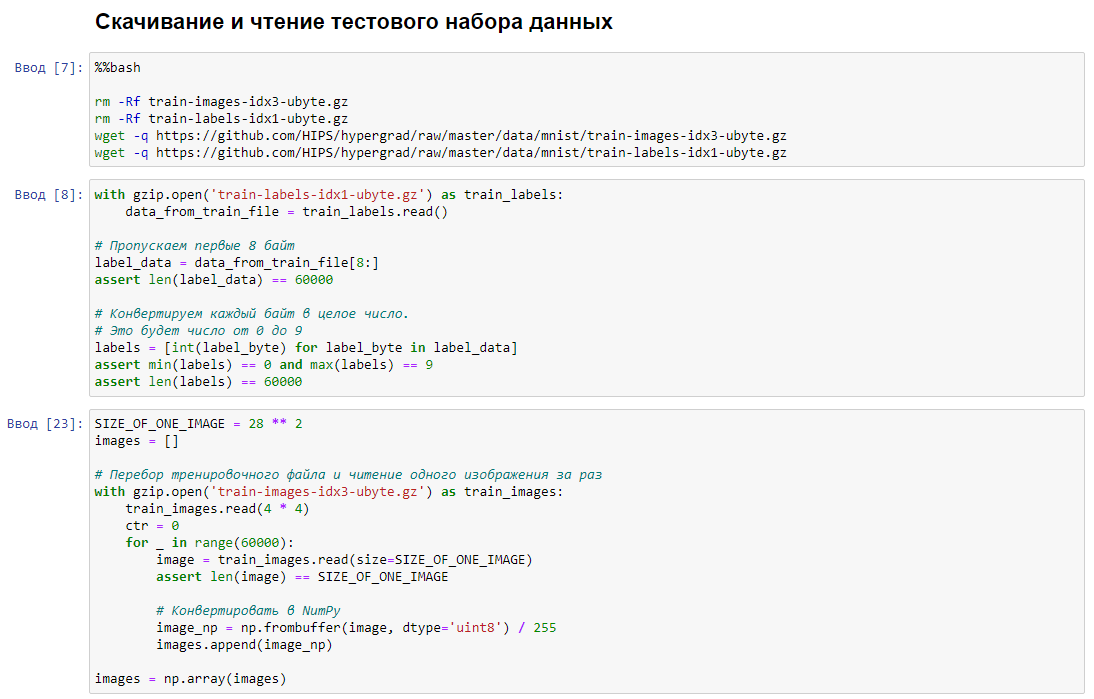
Keras предоставляет еще и определенную свободу: возможность выбрать количество слоев, число нейронов, тип слоя и функцию активации.

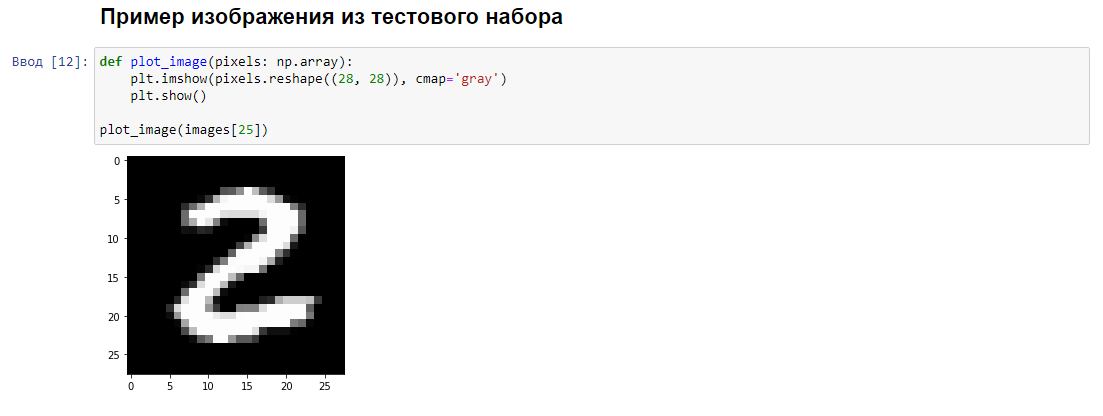
Как уже упоминалось, есть два специальных уровня, которые должны быть определены на основе конкретной проблемы: размер слоя ввода и размер слоя вывода. Все остальные «скрытые слои» используются для изучения сложных нелинейных абстракций задачи.

В этом примере будем использовать [Python и библиотеку](https://pythonru.com/biblioteki) Keras для предсказания рукописных цифр из базы данных MNIST.

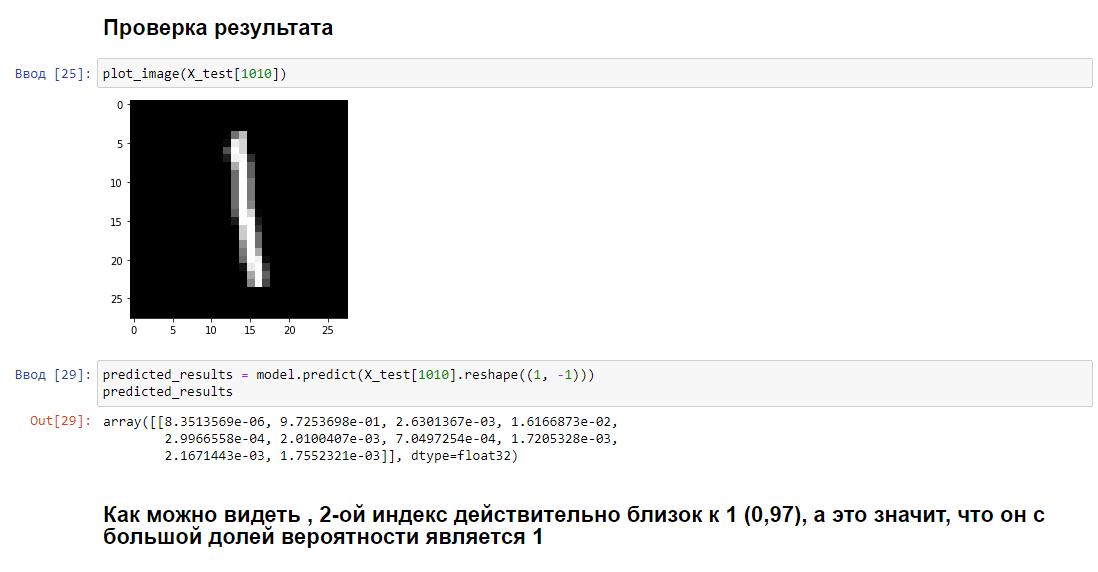
База данных MNIST — это огромная база данных рукописных цифр, которая используется как бенчмарк и точка знакомства с машинным обучением и системами обработки изображений. Она идеально подходит, чтобы сосредоточиться именно на процессе обучения нейронной сети. MNIST — очень чистая база данных, а это роскошь в мире машинного обучения.











Библиотеки Sci-Kit Learn и Keras значительно понизили порог входа в машинное обучение — так же, как Python снизил порог знакомства с программированием.

2D genetic cars

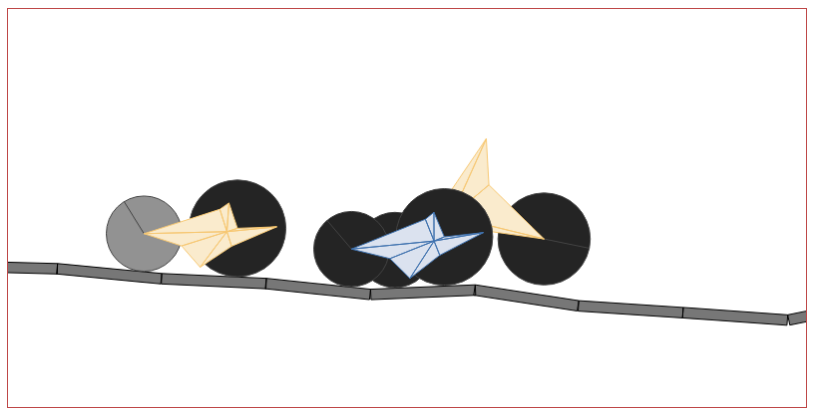


Рис. 1. 2D Genetic Cars

Программа использует простой генетический алгоритм для преобразования случайных двухколесных форм в автомобили на протяжении нескольких поколений.

Геном состоит из:

* Форма (8 генов, по 1 на вершину)
* Размер колеса (2 гена, по 1 на колесо)
* Положение колеса (2 гена, по 1 на колесо)
* Плотность колес (2 гена, по 1 на колесо) более темные колеса означают более плотные колеса
* Плотность шасси (1 ген) более темный корпус означает более плотное шасси

Каждая машина представляет собой набор из 8 случайно выбранных векторов: направления и величины. Все векторы исходят из центральной точки (0,0) и соединены треугольниками.

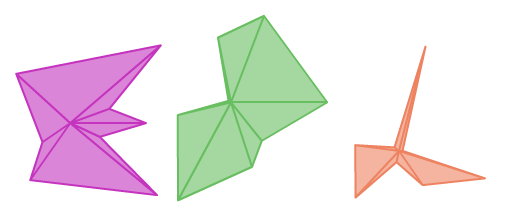


Рис. 2. Примеры векторов.

Для каждого колеса он случайным образом выбирает вершину для установки оси и выбирает угол оси от 0 до 2 пи. Если он выбирает -1 для вершины, это колесо выключается. Нет ничего, что могло бы помешать тому, чтобы несколько колес находились на одной вершине.

В конце каждого поколения необходимо выбрать пары родителей, чтобы произвести потомство для следующего поколения.

Две машины кроссовера A и B производят родители AB и BA. Программа использует двухточечный кроссовер, что означает, что выбираются две случайные точки вдоль хромосомы, и все, что между ними, меняется местами.

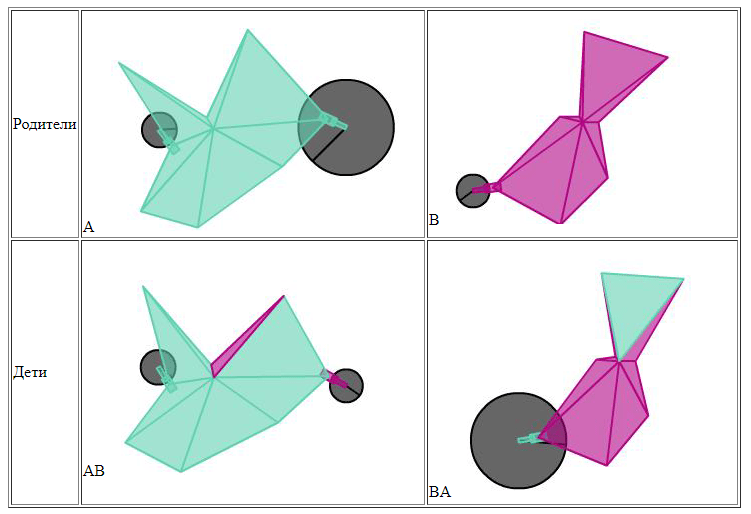


Рис. 3. Кроссовер

Помимо кроссовера, в каждом поколении хромосомы претерпевают мутацию. Это означает, что существует вероятность того, что каждый аспект автомобиля (или переменная в хромосоме) изменится, что определяется ползунком скорости мутации, установленным пользователем. Когда переменная мутирует, новое значение выбирается случайным образом в желаемом диапазоне.

Яндекс Автопоэт

Структура

Ожидается, что стих состоит из последовательности строк, сгруппированных в строфы. Для этих строк должны выполняться некоторые ритмические условия. В силлабическом, тоническом, силлабо-тоническом и других стихах эти условия формулируются по-разному, но так или иначе всё сводится к оценке ударности и длительности каждого слога. Требуется некоторая фонетическая согласованность между разными строками - это, например, может быть рифма или аллитерация.

Алгоритм, который определяет, подходят ли две строки друг другу.

Строится фонетическая транскрипция, считаются слоги и определяется длительность каждого из них, устанавливается соответствие строки желаемому стихотворному размеру и проверяются две строки на наличие взаимной рифмы. Если у нас есть очень много текстовых строк, мы можем их автоматически отфильтровать и составить случайные наборы, согласованные с точки зрения поэтической структуры.

Алгоритм, которые рифмует строки

Сравниваются концовки фонетических транскрипций запросов, начиная от последнего ударного слога. Если концовки совпадают, можно считать, что строки рифмуются. Надо лишь убедиться, что мы не срифмовали слово само с собой.

Для формализации принципов, по которым происходит идентификация стиля конкретного писателя используются нейросети. Нейросеть была собрана, обучена строить фонетику и заставлена «прочитать» огромное число русской поэзии.

Нейросеть не была обучена самостоятельно рифмовать строки, хотя стихотворный размер удерживает достаточно хорошо. Однако это легко исправить, добавив эвристику

Пример

купить в москве ботинки

хочу к тебе картинки

игра плохие свинки

смотреть кино новинки.

---------------------------------

и взор смущенный уж нежданный

ее призывный приговор

все чувства ваши постоянный

мой холоден певец и вор

----------------------------------

нет ты не знаешь ты не спросишь

но он душою обуян

ты не прочтешь и ты не бросишь

густой порой лежит фонтан

Вывод

Мы познакомились с фреймворком Tensorflow для создания искусственного интеллекта, программой 2D Genetic Cars, использующую генетический алгоритм и сервисов Яндекс.Автопоэт. Разобрались с их структурами, а также продемонстрировали небольшие примеры программ: распознавания рукописный цифр, пример стихотворений.