## Задача 2. Распознавание показаний приборов учёта Nhận biết chỉ số đồng hồ

В уездном городке М возникла проблема: там очень хорошо работает ЖКХ. До того хорошо, что жители просят распознавать показания своих счётчиков по фотографиям. Например, включил на телефоне фонарик, засунул его в труднодоступное место, например, под раковину, и показания сами отобразились на экране смартфона. А ещё мастеру управляющей компании было бы очень удобно в подвале или на чердаке снимать показания с прибора учёта, который измеряет потребление ресурсов всем домом сразу. Một vấn đề đã nảy sinh ở thị trấn M thuộc tỉnh: các dịch vụ nhà ở và xã ở đó hoạt động rất tốt. Thật tốt khi người dân yêu cầu nhận ra chỉ số đồng hồ của họ từ các bức ảnh. Ví dụ, tôi bật đèn pin trên điện thoại của mình, dán nó ở một nơi khó tiếp cận, chẳng hạn như dưới bồn rửa và các chỉ số xuất hiện trên màn hình điện thoại thông minh. Sẽ rất thuận tiện cho quản đốc của công ty quản lý lấy số đọc từ đồng hồ đo ở tầng hầm hoặc gác mái, đồng hồ đo mức tiêu thụ tài nguyên của toàn bộ ngôi nhà cùng một lúc.

Задача состоит в разработке модели распознавания показаний индивидуальных и общедомовых приборов учёта. Nhiệm vụ là phát triển một mô hình để nhận biết số đọc của các thiết bị đo sáng cá nhân và chung.

Данные для построения модели:

1. Архив с фотографиям приборов учёта с описанием:

Lưu trữ hình ảnh các thiết bị đo đếm kèm theo mô tả:

<https://cloud.athene.tech/f/c37eab8f14884b348aef/?dl=1>.

Исходные данные:

1. Файл с заданиями на распознавание показаний ***/mnt/source/task.csv*** со следующими параметрами: Tệp có tác vụ nhận dạng đọc /mnt/source/task.csv với các tham số sau:
   * *file\_name* - имя файла c изображением. tên của tập tin hình ảnh.
2. Собственно фотографии приборов учёта для распознавания, находящиеся в той же папке, что и файл с заданиями. Trên thực tế, ảnh chụp các thiết bị đo sáng để nhận dạng, nằm trong cùng thư mục với tệp chứa các tác vụ.

Выходные данные:

1. Файл с распознанными показаниями ***/mnt/result/values.csv*** со следующими параметрами: Tệp có số đọc được nhận dạng /mnt/result/values.csv với các tham số sau:
   * *file\_name* - имя файла c изображением (такое же, как и в *task.csv*); tên của tệp có hình ảnh (giống như trong task.csv);
   * *data1* - целая часть показаний до запятой; phần nguyên của số đọc đến dấu thập phân;
   * *data2* - дробная часть показаний после запятой. phần phân số của số đọc sau dấu thập phân.

Примечания:

* Формат всех файлов CSV: с заголовком, разделители - «;». Định dạng của tất cả các tệp CSV: có tiêu đề, được phân cách bằng dấu “;”.

На проверку отправляется архив, в котором присутствует: Một kho lưu trữ được gửi để xác minh, trong đó có:

* Файл ***README.pdf*** с кратким описанием применённого решения.
* Исходные тексты программы, формирующей модель, в папке ***/src-train***.
* Исходные тексты программы, преобразующей исходные данные в выходные на основании заранее сформированной (обученной) модели в папке ***/src-predict***.
* Собственно модель в папке ***/model***.
* Файл ***Dockerfile***, в котором создаётся контейнер и запускается программа для преобразования исходных данных в выходные (из */src-predict*). Dockerfile tạo vùng chứa và chạy chương trình để chuyển đổi đầu vào thành đầu ra (từ /src-predict).

Требование самостоятельно описывать окружение через Dockerfile необходимо для унификации проверки. Пример файла «*docker-compose.yml*» для запуска задания на проверку: Yêu cầu mô tả độc lập môi trường thông qua Dockerfile là cần thiết để thống nhất xác minh. Một ví dụ về tệp “docker-compose.yml” để chạy công việc xác minh:

|  |
| --- |
| version: "3.1"  services:  app:  build: .  volumes:  - .:/app  - ${PATH\_SOURCE}:/mnt/source  - ${PATH\_RESULT}:/mnt/result |

, где ${PATH\_SOURCE} - путь до каталога с исходными данными, ${PATH\_RESULT} - путь до каталога с выходными данными для проверки. trong đó ${PATH\_SOURCE} là đường dẫn đến thư mục chứa dữ liệu nguồn, ${PATH\_RESULT} là đường dẫn đến thư mục chứa dữ liệu đầu ra cần kiểm tra.

То есть запуск программы будет осуществляться со следующими характеристиками: Tức là chương trình sẽ được khởi chạy với các đặc điểm sau:

* ОС - Linux, настроенная через Dockerfile для запуска программы; OS - Linux, được cấu hình qua Dockerfile để chạy chương trình;
* Данные их архива на проверку примонтированы в */app*; Dữ liệu lưu trữ của họ được gắn vào /app để xác minh;
* Соответственно, модель находится по пути */app/model*; Theo đó, mô hình nằm trên đường dẫn /app/model;
* Исходные данные примонтированы в */mnt/source*; Dữ liệu nguồn được gắn vào /mnt/source;
* Каталог для сбора выходных данных примонтирован в */mnt/result*. Thư mục tập hợp đầu ra được gắn vào /mnt/result.

Запуск контейнера будет осуществляться через команду «docker compose up --build» с заданными переменными окружения PATH\_SOURCE и PATH\_RESULT. Vùng chứa sẽ được khởi chạy thông qua lệnh “docker soạn thảo --build” với các biến môi trường PATH\_SOURCE và PATH\_RESULT được chỉ định.

**Если у Вас возникают трудности с написанием Dockerfile, опишите более подробно, как запустить программу predict для формирования выходного файла. Nếu bạn gặp khó khăn khi viết Dockerfile, vui lòng mô tả chi tiết hơn cách chạy dự đoán để tạo tệp đầu ra.**