

Государственное бюджетное общеобразовательное учреждение
средняя общеобразовательная школа № 297 Пушкинского района Санкт-Петербурга

**Алгоритм методики определения, распознавания и
оцифровки рукописного текста
в рамках инновационного проекта**

**«Платформа «Умка – вектор добра» - основа цифровой
экосистемы образования с применением технологий
Искусственного интеллекта»**



Санкт-Петербург

2023

Алгоритм методики определения, распознавания и оцифровки рукописного текста.

1. Выбор инструмента для распознавания рукописного текста.

2. Подготовка обучающих данных.

2.1. Сбор образцов рукописного текста и его преобразование в формат, используемый для обучения модели.

3. Обучение модели на собранных данных.

Этот процесс наиболее трудоемок и включает в себя ряд мероприятий, таких как:

3.1. Сбор обучающих изображений в формате JPEG.

3.2. Подготовка разметки (создание текстового файла) для каждого изображения, с указанием распознаваемого текста.

3.3. Подготовка обучающего корпуса, объединив изображения и файлы разметки.

3.4. Генерация box – файлов, содержащих информацию о расположении символов на изображении.

3.5. Создание выходного файла.

3.6. Непосредственное обучение модели и сбор полученных данных.

3.7. Перенос файлов с обученной моделью и конфигурационные файлы в корневой каталог.

3.8. Установка обученной модели.

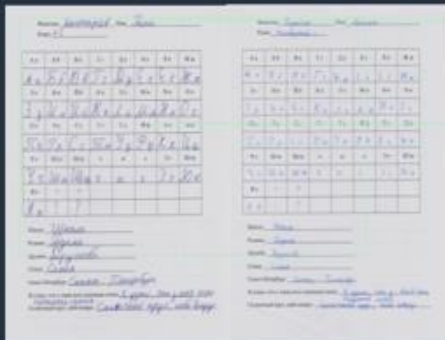
4. Интеграция модели в цифровую платформу «УМКА – вектор добра».

4.1. После обучения модели становится возможным ее интегрирование в приложение, чтобы она могла принимать ввод в виде изображений с рукописным текстом и возвращать печатный текст.

ПРОЦЕСС

СОЗДАНИЯ И ТРЕНИРОВКИ НЕЙРОСЕТИ

ПОДГОТОВКА ОБУЧАЮЩИХ ДАННЫХ.
СБОР ОБРАЗЦОВ РУКОПИСНОГО ТЕКСТА



ОБУЧЕНИЕ МОДЕЛИ НА СОБРАННЫХ ДАННЫХ

```
import tensorflow  
def main():  
    # Загрузка данных  
    data_loader = tf.keras.preprocessing.image.ImageDataGenerator(  
        rotation_range=180,  
        width_shift_range=0.1,  
        height_shift_range=0.1,  
        shear_range=0.1,  
        zoom_range=0.1,  
        channel_shift_range=0.1,  
        fill_mode='nearest')  
    # Генерация данных  
    data_loader.flow_from_directory('data',  
                                  target_size=(28, 28),  
                                  class_mode='categorical')  
    # Обучение модели  
    model = tf.keras.Sequential([  
        tf.keras.layers.Flatten(),  
        tf.keras.layers.Dense(100, activation='relu'),  
        tf.keras.layers.Dense(10, activation='softmax')  
    ])  
    model.compile(optimizer='adam', loss='categorical_crossentropy',  
                 metrics=['accuracy'])  
    model.fit_generator(data_loader.flow_from_directory('data',  
                                                       target_size=(28, 28),  
                                                       class_mode='categorical'),  
                       steps_per_epoch=100, epochs=10, validation_data=(  
                            data_loader.flow_from_directory('validation',  
                                                           target_size=(28, 28),  
                                                           class_mode='categorical')  
                        ))
```

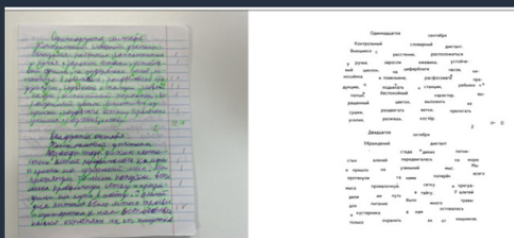
РАБОЧИЙ АНАЛИЗ.
ПОДГОТОВКА РАЗМЕТКИ



ГЕНЕРАЦИЯ ВОХ - ФАЙПОВ

Школа	C:\Users\hp\Downloads\tekst Degtiarev1.jpg
Родина	C:\Users\hp\Downloads\tekst Degtiarev2.jpg
Дружба	C:\Users\hp\Downloads\tekst Degtiarev3.jpg
Семья	C:\Users\hp\Downloads\tekst Degtiarev4.jpg
Санкт-Петербург	C:\Users\hp\Downloads\tekst Degtiarev5.jpg

НЕПОСРЕДСТВЕННОЕ ОБУЧЕНИЕ МОДЕЛИ И СБОР ПОЛУЧЕННЫХ ДАННЫХ



КОНТРОЛЬ ЗА РЕЗУЛЬТАТОМ

Вьющиеся растение, расположить у ручья, заросли ежевики, устойчивый циклон, сучья акации, на циферблате часов, киносъемка в павильоне, расфасовать продукцию, подъехать к станции, ребячи пальцы, беспокойный характер, выращенный в оранжерее цветок, выложить из сумел и, раздвигать ветки, прилагать усилия, разжечь костёр.