ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «УРАЛЬСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ ПЕРВОГО ПРЕЗИДЕНТА РОССИИ Б.Н. ЕЛЬЦИНА»

(ФГАОУ ВО «УрФУ имени первого Президента России Б.Н. Ельцина»)

УТВЕРЖДАЮ

Директор Школы бакалавриата ИРИТ-РТФ,

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_М.Ю. Новиков

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

ОТЧЁТ

О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

Разработка программы для распознавания рукописного печатного текста на бланке

(заключительный)

Научный руководитель,

Директор Школы

бакалавриата ИРИТ-РТФ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.Ю. Новиков

г. Екатеринбург 2024

**Оглавление**

**1.Описание проекта3**

1.1 Проблема3

1.2 Решение3

1.3 Целевая аудитория3

1.4 Стек3

**2.Описание работы команды4**

2.1 План работы над проектом4

2.2 Анализ аналогов и конкурентов4

2.3 User Story5

2.4 Ход реализации5

2.5 Тестирование6

**3.Результаты проекта7**

3.1 Название продукта7

3.2 Описание7

3.3 Функциональность7

3.4 Преимущества7

3.5 Вывод7

**1. Описание проекта**

**1.1 Проблема**

Нецелесообразность использования рабочего труда в задаче считывания информации с бланка, невозможность интегрирования сторонних сервисов в свой проект.

**1.2 Решение**

Модуль Python “handtext decoder”, который принимает на вход фотографию, преобразовывает её в необходимый формат, выполняет сегментацию слова, отправляет каждый сегмент в сверточную нейросеть, которая на выходе прогнозирует символ, и выводит распознанное слово.

**1.3 Целевая аудитория**

Разработчики программного обеспечения для образовательные, медицинские, государственные и муниципальные учреждения, коммерческие компании.

**1.4 Стек для разработки**

Python, Keras, TensorFlow, OpenCV, Numpy, Matplotlib, AIOgram, Google Colab.

**2. Описание работы команды**

**2.1 План работы над проектом**

План работы над проектом представлен в Таблице 1.

Таблица 1 – План работы

|  |  |
| --- | --- |
| 19.02.24-03.03.24 | Поиск информации, аналитика |
| 04.03.24-14.04.24 | Обучение модели |
| 15.04.24-28.04.24 | Написание кода для работы с изображением |
| 29.04.24 – 05.05.24 | Составление модуля Python |
| 06.05.24-19.05.24 | Написание кода для тг-бота |
| 20.05.24-23.05.24 | Тестирование решения |
| 24.05.24-06.06.24 | Составление презентации, защитного слова, подготовка к предзащите |
| 07.06.24 | Предзащита |
| 08.06.24-23.06.24 | Деплой бота, исправление недочётов |
| 24.06.24-27.06.24 | Подготовка к защите |
| 28.06.24 | Защита |

**2.2 Анализ аналогов и конкурентов**

Анализ аналогов и конкурентов представлен в Таблице 2.

Таблица 2 - Анализ аналогов и конкурентов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Формат продукта | Точность | Скорость распознавания одного слова | Ограничения | Формат вывода |
| Apple | Встроено в гаджет | 11/29 | <1сек | - | Текст |
| Умная камера Yandex | Приложение | 11/29 | <1сек | - | Текст |
| Google объектив | Приложение | 17/29 | <1сек | - | Текст |
| PIC2TEXT | Тг-бот | 12/29 | ≈1мин | 3 бесплатных распознавания | Текстовое сообщение |
| SCANNERTEXT\_BOT | Тг-бот | 5/29 | <1сек | - | Текстовое сообщение |
| REHAND.RU | Сайт | 20/29 | <1сек | Несколько бесплатных распознаваний | Текcт |
| PYTESSERACT | Модуль Python | 21/29 | <1сек | - | Строка |
| HANDTEXT DECODER | Модуль Python | 24/29 | ≈0.8 сек | - | Строка |

**2.3 User Story**

Как: Разработчик программного обеспечения

Я хочу: Импортировать библиотеку для распознавания текста из изображений, использовать её для обработки изображений

Чтобы: Автоматизировать процесс извлечения текстовой информации из изображений бланков в своих приложениях

Подробное описание:

1. Разработчик находит библиотеку для распознавания текста и устанавливает её в свой проект.
2. Разработчик импортирует библиотеку в свой код.
3. Разработчик загружает изображение, которое нужно распознать.
4. Разработчик применяет функцию библиотеки для распознавания текста.
5. Разработчик получает строку с распознанным текстом и использует её в своем приложении.

**2.4 Ход реализации**

1. Инициация:

* определение целей и задач
* определение целевой аудитории проекта
* определение требований

1. Анализ и планирование

* Исследование аналогов и конкурентов
* Разработка технического задания
* Составление плана работы

1. Разработка

* Создание и обучение модели
* Написание кода для обработки изображений
* Составление модуля
* Разработка тг-бота

1. Тестирование

**2.5 Тестирование**

Корректность работы модуля, а также тг-бота была проверена на данных, предоставленных куратором. Было распознано 24 символа из 29.

**3. Результаты проекта**

**3.1 Название продукта:**

Модуль на Python для распознавания текста с изображений “handtext decoder”

**3.2 Описание:**

Разработанный модуль представляет собой библиотеку на языке Python, предназначенную для извлечения рукописного печатного текста с изображений бланков.

**3.3 Функциональность:**

* **Импорт библиотеки:** Библиотека легко интегрируется в существующие проекты.
* **Обработка изображений:** Модуль принимает изображение, преобразует его в необходимый формат и выполняет предобработку.
* **Сегментация слов:** Изображение разбивается на отдельные сегменты по границам бланка.
* **Распознавание символов:** Сегменты обрабатываются сверточной нейросетью, которая прогнозирует символы.
* **Формирование текста:** Прогнозируемые символы собираются в полноценное слово, и текст возвращается пользователю в виде строки.

**3.4 Преимущества:**

* **Удобство использования:** Простота интеграции и использования модуля в различных проектах.
* **Высокая точность и скорость**
* **Поддержка различных форматов изображений:** Модуль поддерживает множество форматов, включая JPEG, PNG.

**3.5 Вывод:**

Нам удалось обучить нейросеть под свою задачу, разработать модуль python, который удовлетворяет всем требованиям и во многом опережает аналоги в сфере распознавания рукописных печатных текстов с бланка, а также запустить тг-бота для визуализации работы нашего модуля