

Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное
образовательное учреждение высшего образования
«Самарский национальный исследовательский университет
имени академика С.П. Королева»

Институт информатики, математики и электроники

Факультет информатики

Кафедра технической кибернетики

Отчет по курсу

Дисциплина: «Технологии сетевого программирования»

Выполнил: Афанасьев В.А.

Группа: 6301-010302D

Самара, 2025

СОДЕРЖАНИЕ

1 Концепция приложения	3
1.1 Назначение приложения	3
1.2 Состав приложения	3
1.3 Описание функциональных частей	3
1.3.1 Серверная часть	3
1.3.2 Клиентская часть	3
1.3.3 База данных	3
1.3.4 Сервис предобработки изображения	3
1.3.5 Сервис распознавания эмоций.....	4
1.4 Функционирование приложения	4
2 UseCases	5
3 Работа приложения	7
3.1 Регистрация и авторизация	7
3.2 Выбор конфигурации и получение информации.....	8
3.3 Предсказание эмоций	10
3.4 Просмотр истории предсказаний.....	12
4 Ссылка на репозиторий GitHub	13

1 Концепция приложения

1.1 Назначение приложения

Приложение предназначено для распознавания эмоций людей по лицевым выражениям по фотографиям.

1.2 Состав приложения

Приложение состоит из 5 частей:

- 1) Серверная часть приложения.
- 2) Клиентская часть приложения.
- 3) База данных.
- 4) Сервис предобработки изображения.
- 5) Сервис распознавания эмоций.

1.3 Описание функциональных частей

1.3.1 Серверная часть

Серверная часть приложения написана на языке программирования Python с использованием фреймворка Django.

Также в ходе работы использовался Django REST Framework для написания методов API.

1.3.2 Клиентская часть

Клиентская часть реализована на языке программирования Python с использованием фреймворка Django. Кроме того использовался язык разметки HTML с шаблонизацией из Django, язык стилей CSS и язык JavaScript для работы с формами и кнопками на web-страницах.

1.3.3 База данных

В качестве СУБД была выбрана PostgreSQL. Для работы с базой использовались ORM модели, предлагаемые фреймворком Django.

1.3.4 Сервис предобработки изображения

Сервис предобработки изображения выполняет функцию нахождения лица на изображении и последующую обрезку фотографии по рамке лица.

Сервис реализован на языке программирования Python. Для коммуникации с сервисом используется единственный API метод, реализованный с помощью фреймворка Flask.

Предобработка изображения производится при помощи библиотеки OpenCV и класса CascadeClassifier.

1.3.5 Сервис распознавания эмоций

Сервис предобработки распознаёт эмоции по готовым изображениям лиц людей.

Он так же реализован на Python и имеет единственный метод API, сделанный при помощи фреймворка Flask. Для нейронных сетей использовался фреймворк машинного обучения PyTorch.

На выбор пользователю предоставлены 10 архитектур нейронных сетей и 2 датасета.

1.4 Функционирование приложения

Приложение собрано через docker-compose.yml файл. Образы функциональных частей в сумме весят 17 гигабайт.

2 UseCases

Таблица 1 – Таблица описания актёров системы

Название	Описание
Система	Модуль, обрабатывающий сообщения от пользователя
Пользователь	Человек, который загружает анализируемые изображения
GUI	Модуль, позволяющий пользователю загружать изображения

Таблица 2 – Детальное описание вариантов использования

1	Название	Войти в систему
	Описание	Пользователь входит в свой профиль в системе
	Цель	Дать пользователю возможность взаимодействовать с системой
	Актёры	
	1	Инициатор – Пользователь П
	2	Участник – Система С
	3	Участник – GUI Гр
	Предусловия	Система включена, Гр включен
	Постусловия	В Гр хранятся токены доступа
	1	П вводит пароль и логин с Гр
	2	С получает информацию от Гр
	3	Гр получает токены от С
	Альтернативы	
	1.1	П вводит неправильный пароль или логин
	1.2	Гр просит пользователя ввести повторно
	Исключения	
	2.1	С не получает информацию от Гр
	3.1	Гр не получает токены от С
2	Название	Поменять конфигурацию
	Описание	Пользователь выбирает конфигурацию сетей
	Цель	Изменить конфигурацию сетей
	Актёры	
	1	Инициатор – Пользователь П
	2	Участник – Система С
	3	Участник – GUI Гр
	Предусловия	С включена, Гр включен, выполнен вход в систему
	Постусловия	В С установлена нужная конфигурация сетей
	1	П выбирает сеть и датасет в Гр

	2	Гр передаёт конфигурацию в С
	3	С устанавливает конфигурацию
	4	С передаёт значение точности по конфигурации на Гр
	5	Гр выводит П значение точности
	Альтернативы	
	1.1	П выбирает только часть конфигурации в Гр
	1.2	Гр говорит П выбрать конфигурацию полностью
	Исключения	
	2.1	С не получила данные от Гр
	4.1	Гр не получает точность от С
3	Название	Классифицировать изображения
	Описание	Осуществляется классификация эмоций на изображении
	Цель	Классифицировать эмоции на загруженном изображении
	Актёры	
	1	Инициатор – Пользователь П
	2	Участник – Система С
	3	Участник – GUI Гр
	Предусловия	С включена, Гр включен, выполнен вход в систему
	Постусловия	П передаётся вектор вероятностей эмоций
	1	П загружает в Гр картинку
	2	Гр передаёт картинку С
	3	С предобрабатывает картинку
	4	С классифицирует эмоции
	5	С передаёт изображение и вектор эмоции на Гр
	6	Гр выводит изображение и вектор эмоций пользователю
	Альтернативы	
	3.1	С не находит лица человека на изображении
	3.2	С передаёт на Гр просьбу о загрузке другого изображения
	3.3	Гр просит П загрузить другое изображение
	Исключения	
	2.1	Д не смогли отправить данные С
	5.1	С не смогла передать ответ Гр

3 Работа приложения

3.1 Регистрация и авторизация

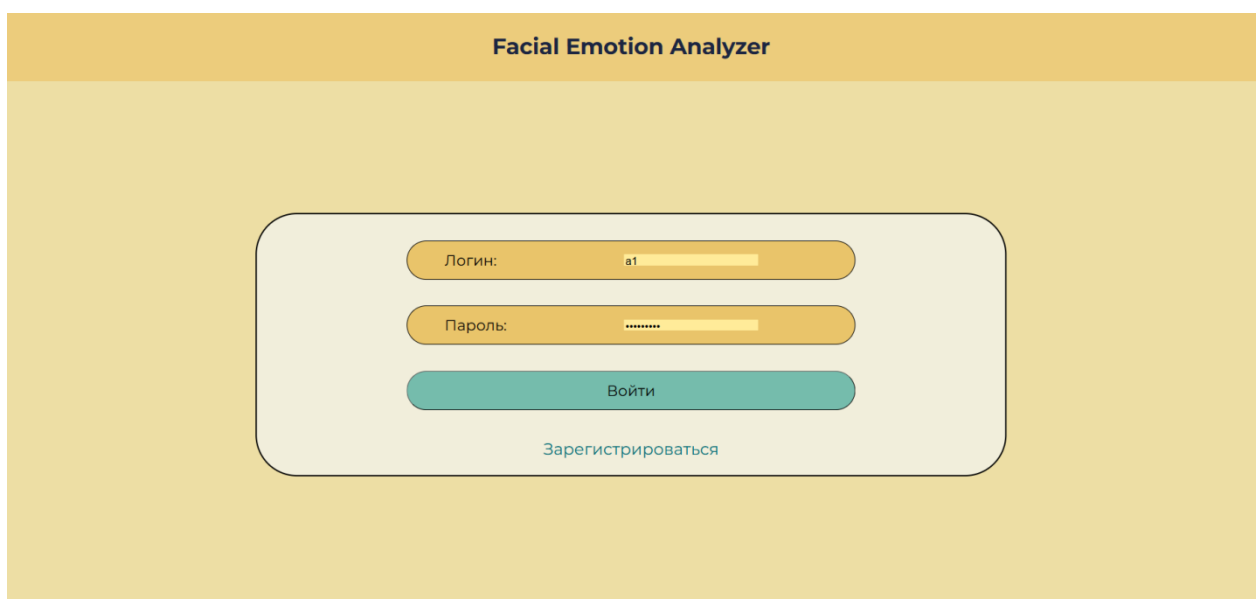
Для получения доступа к основному функционалу приложения необходим вход в учётную запись. Если учётной записи не существует, то её можно зарегистрировать.

Учётная запись необходима для хранения истории распознавания.

Авторизация выполнена с помощью стандартных инструментов фреймворка Django. В качестве модели пользователя используется встроенная модель User. Для авторизации пользователя используются встроенные методы TokenObtainPairView и TokenRefreshView. Для выхода из учётной записи так же используется встроенный инструмент TokenBlacklistView.

Регистрация пользователя происходит через сериализатор, получающий значения полей и регистрирующий пользователя в базе данных.

Для регистрации пользователю необходимо указать логин, пароль и адрес электронной почты. Кроме того, пароль необходимо продублировать. Сам пароль должен быть длиннее 8 символов и содержать буквы латинского алфавита и цифры. Проверка на простоту пароля осуществляется неявно внутренними средствами фреймворка.



The image shows a web interface for a 'Facial Emotion Analyzer'. At the top, there is a yellow header bar with the text 'Facial Emotion Analyzer' in black. Below the header, the main content area has a light yellow background. In the center, there is a white rounded rectangle containing the login form. The form consists of three input fields: 'Логин:' (Login) with the value 'a1', 'Пароль:' (Password) with masked characters, and a 'Войти' (Login) button. Below the login button, there is a link 'Зарегистрироваться' (Register).

Рисунок 1 – Страница входа

Facial Emotional Analyzer

Логин: a1

Пароль:

Повторите пароль: Повторите пароль

Email: Email

Продолжить

Войти в аккаунт

Рисунок 2 – Страница регистрации

3.2 Выбор конфигурации и получение информации

На главной странице сайта доступен выбор конфигурации нейронной сети и датасета. По умолчанию пользователю выставляется следующая конфигурация: сеть VGG-11 и датасет AffectNet. После установки конфигурации пользователь видит расчетное значение метрики Ассурасу.

Установка конфигурации осуществляется выбором нейронной сети и датасета и последующим нажатием на кнопку подтверждения.

Facial Emotion Analyzer

Выберите модель и набор данных

Выберите модель

-- Выберите модель --

-- Выберите модель --

VGG-11

VGG-16

VGG-19

ResNet-18

ResNet-50

ResNet-152

SqueezeNet

DenseNet121

DenseNet201

Inception-v3

Подтвердить выбор

Выберите набор данных

-- Выберите набор данных --

Текущая конфигурация

DenseNet201

AffectNet

Рассчитанная точность

73.0%

Выйти из профиля

Настройка конфигурации

Информация о наборах данных

Информация о моделях

Информация об обученных моделях

История

Главная

Предсказать

Рисунок 3 – Выбор нейронной сети

Кроме того на главной странице можно посмотреть описание моделей, датасетов и значение точности для каждой конфигурации.

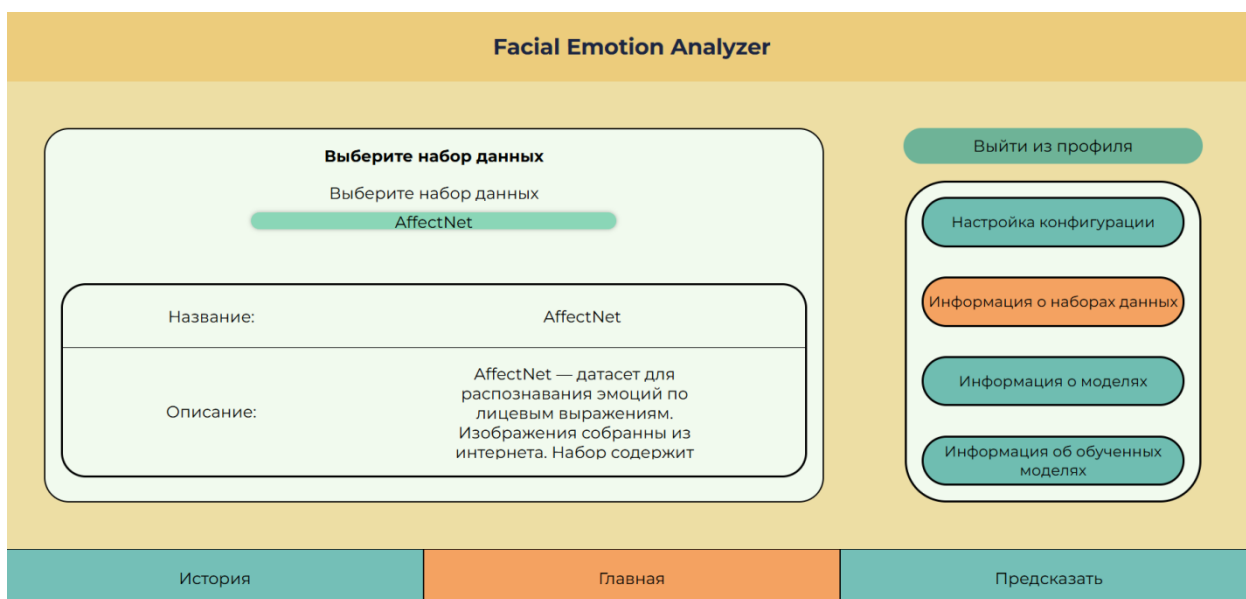


Рисунок 4 – Информация о датасете

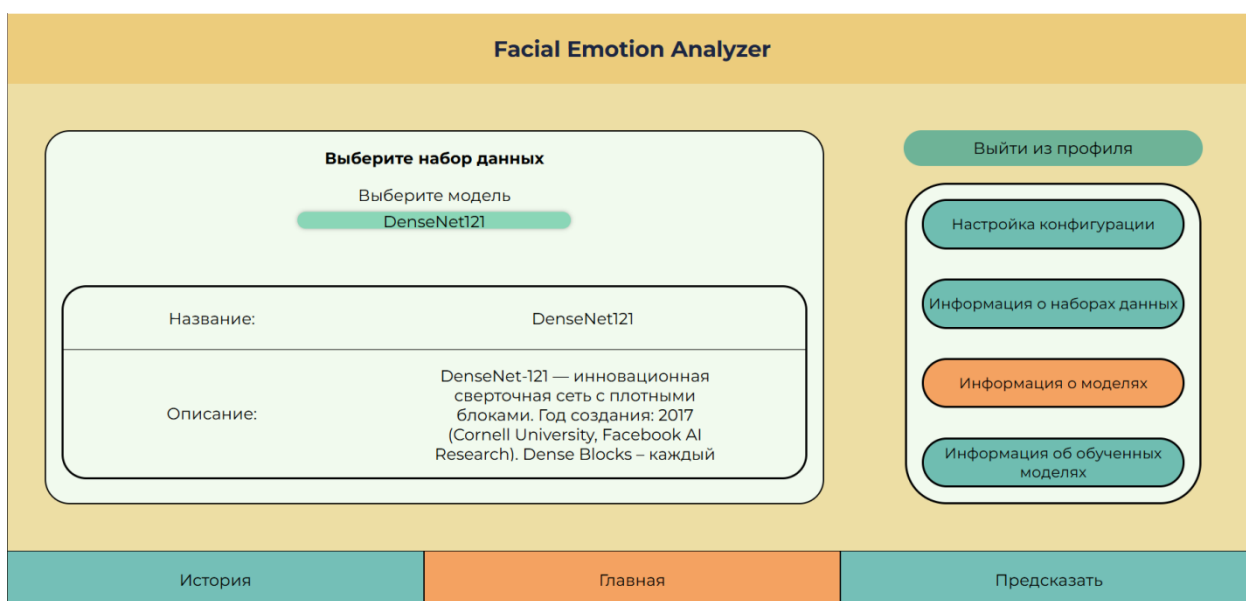


Рисунок 5 – Информация о модели

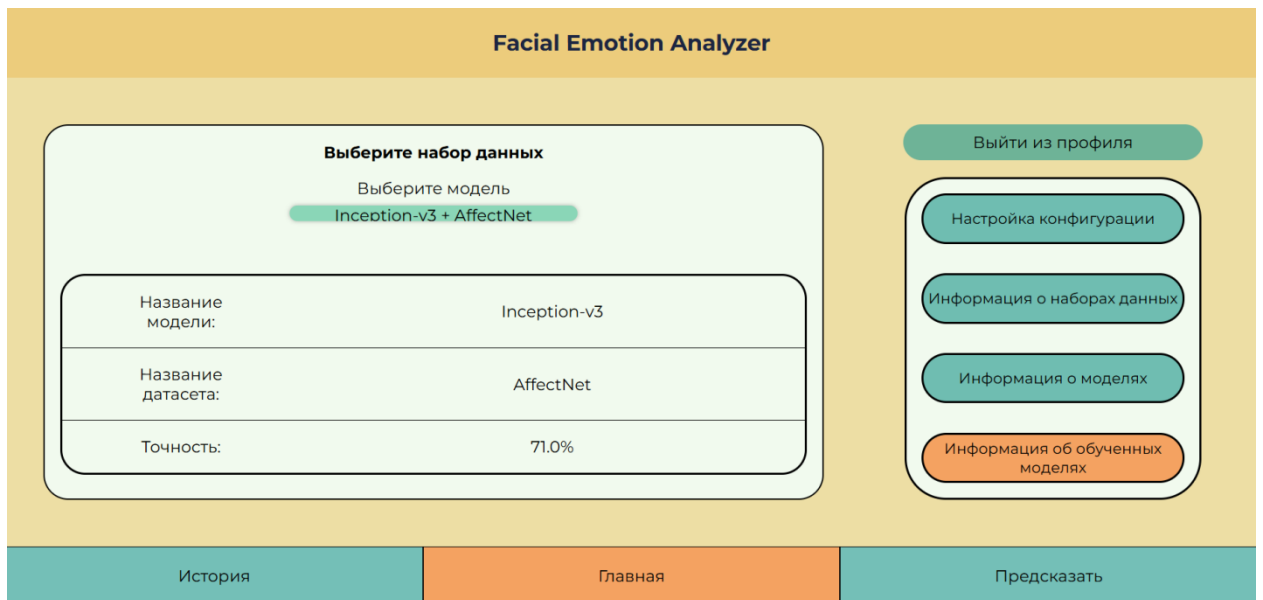


Рисунок 6 – Информация о датасете

3.3 Предсказание эмоций

Для предсказания эмоций необходимо выбрать картинку из файлов системы.

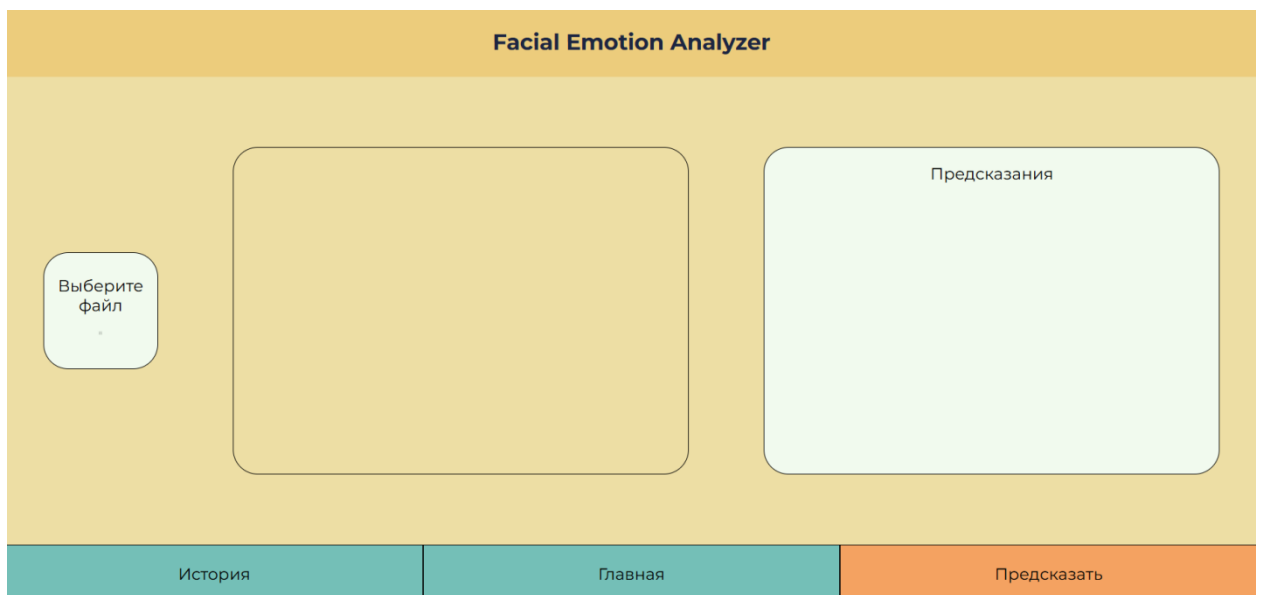


Рисунок 7 – Страница предсказания

После выбора изображения система выдаст один из 2 возможных вариантов:

- 1) Результат с вектором вероятностей эмоций.

2) Сообщение об ошибке в процессе анализа эмоций или предобработки изображения.

В первом случае вектор вероятностей эмоций будет упорядочен по убыванию и представлен целочисленными значениями в процентах.

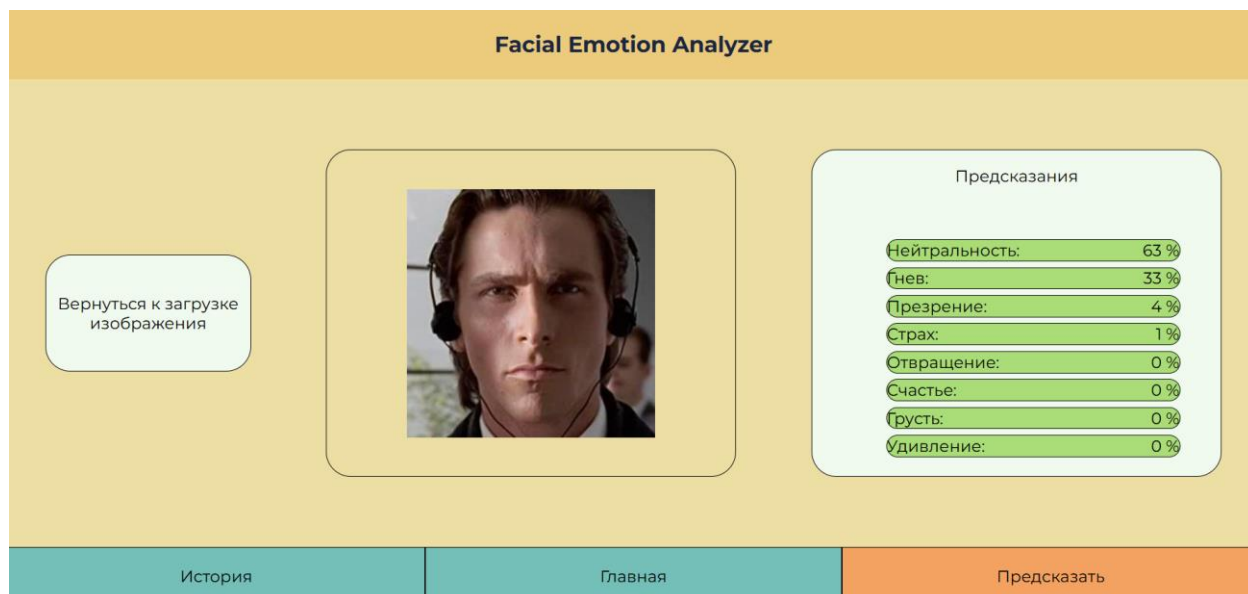


Рисунок 8 – Предсказание эмоций

После получения любого из вариантов результатов можно вернуться к загрузке изображения.

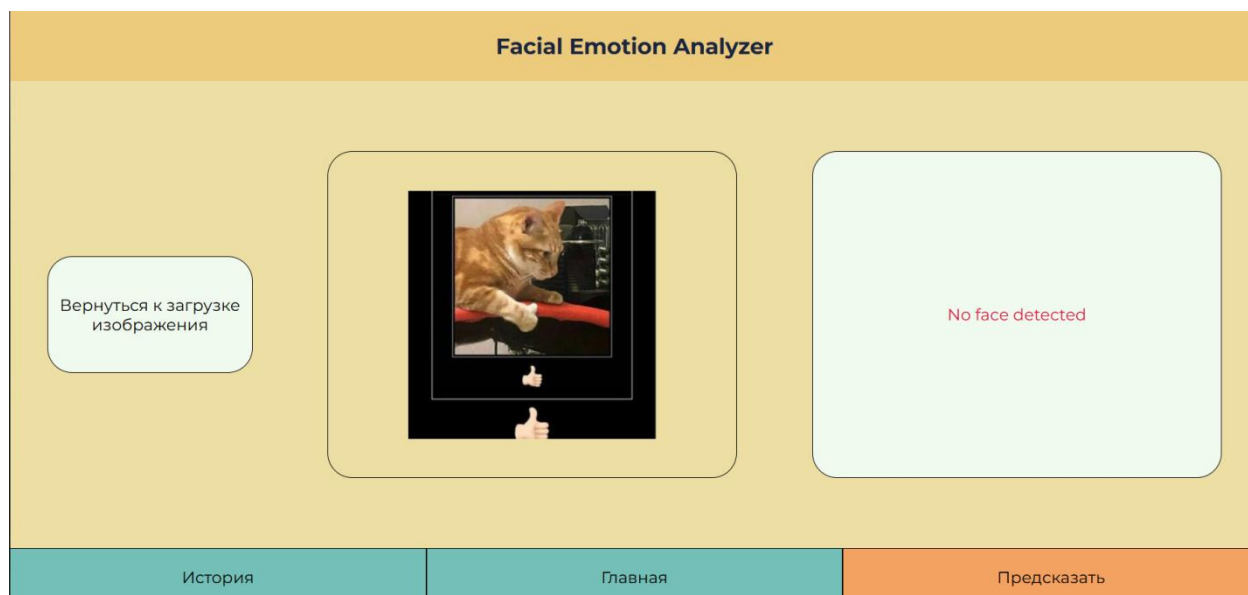


Рисунок 9 – Вывод ошибки предсказания

3.4 Просмотр истории предсказаний

Также пользователю доступна возможность просмотра истории его удачных предсказаний (т.е. только те, в ходе которых не возникло ошибок).

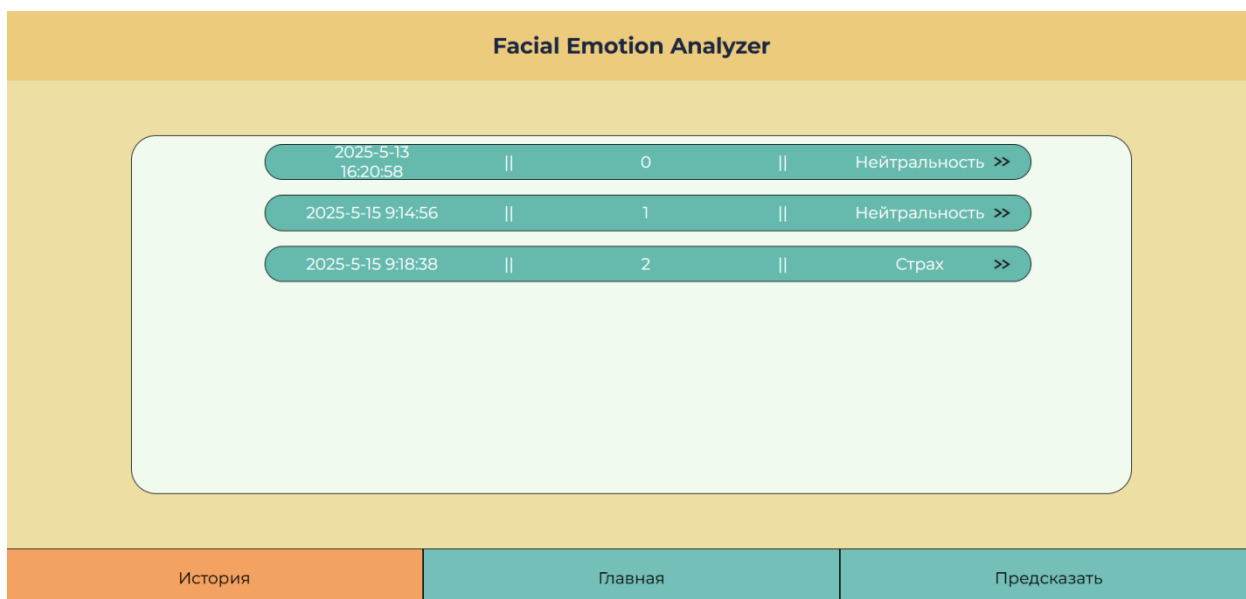


Рисунок 10 – Страница истории

При нажатии на конкретное предсказание будет выведено его детальное описание как на странице предсказаний.

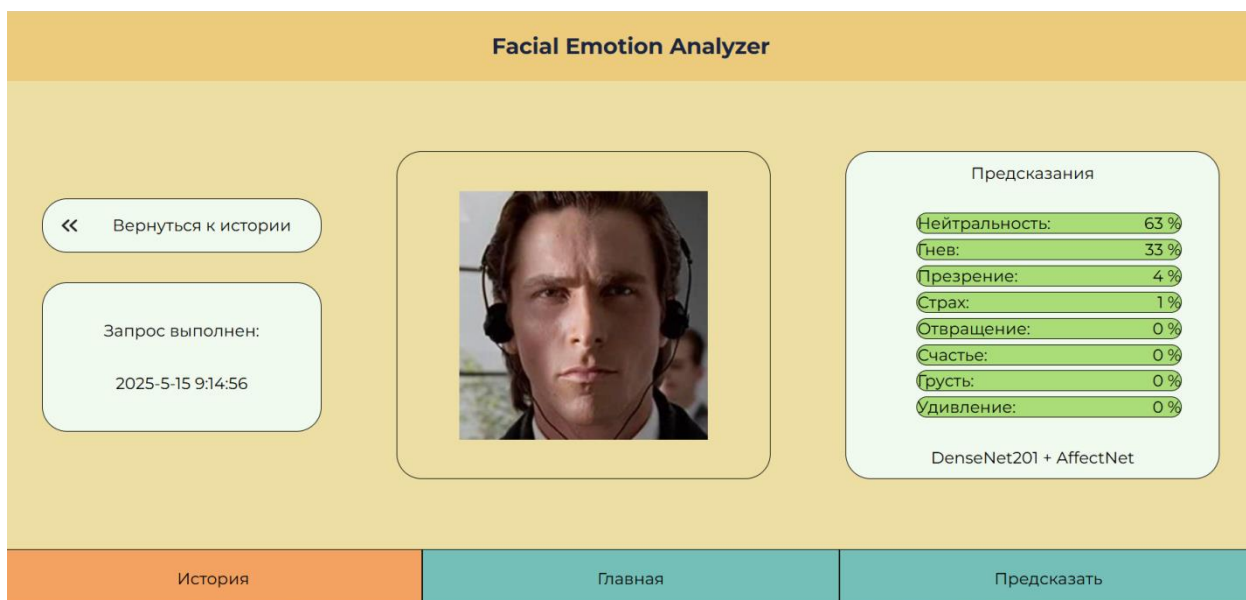


Рисунок 11 – Детальное предсказание

Детальное предсказание на странице с историей в отличии от страницы предсказания содержит ещё и время выполнения запроса с указанием конфигурации сети и датасета.

4 Ссылка на репозиторий GitHub

<https://github.com/listussr/network-programming-technologies>