

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

СОГЛАСОВАНО  
Научный руководитель ИЦ СИИП  
Университета ИТМО

\_\_\_\_\_ А. В. Бухановский  
\_\_\_\_\_ 2023 г.

УТВЕРЖДАЮ  
Директор СПб ФИЦ РАН

\_\_\_\_\_ А. Л. Ронжин  
\_\_\_\_\_ 2023 г.

**БИБЛИОТЕКА АЛГОРИТМОВ СИЛЬНОГО ИИ  
ДЛЯ ЗАДАЧ ПРОМЫШЛЕННОСТИ**

**КОМПОНЕНТ БИБЛИОТЕКИ АЛГОРИТМОВ СИЛЬНОГО ИИ В ЧАСТИ  
АЛГОРИТМОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ПОВЕДЕНИЯ ЧЕЛОВЕКА  
НА ОСНОВЕ ЕГО МУЛЬТИМОДАЛЬНЫХ ДАННЫХ, ОБЕСПЕЧИВАЮЩИХ  
ОЦЕНИВАНИЕ УРОВНЯ ОТДЕЛЬНЫХ ПЕРСОНАЛЬНЫХ КАЧЕСТВ  
ЛИЧНОСТИ ЧЕЛОВЕКА (OCEAN-AI)**

РУКОВОДСТВО ПРОГРАММИСТА

ЛИСТ УТВЕРЖДЕНИЯ

RU.CHAB.00853-01 33 11-ЛУ

Инв. № подл.	Подп. и дата	Инв. № дубл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Инв. № подл.

Представители  
Организации-разработчика

Руководитель разработки

\_\_\_\_\_ А. А. Карпов  
\_\_\_\_\_ 2023 г.

Нормоконтролер

\_\_\_\_\_ Н. А. Александрова  
\_\_\_\_\_ 2023 г.

2023

УТВЕРЖДЕН  
RU.СНАБ. 00853-01 33 11-ЛУ

**БИБЛИОТЕКА АЛГОРИТМОВ СИЛЬНОГО ИИ  
ДЛЯ ЗАДАЧ ПРОМЫШЛЕННОСТИ**

**КОМПОНЕНТ БИБЛИОТЕКИ АЛГОРИТМОВ СИЛЬНОГО ИИ В ЧАСТИ  
АЛГОРИТМОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ПОВЕДЕНИЯ ЧЕЛОВЕКА  
НА ОСНОВЕ ЕГО МУЛЬТИМОДАЛЬНЫХ ДАННЫХ, ОБЕСПЕЧИВАЮЩИХ  
ОЦЕНИВАНИЕ УРОВНЯ ОТДЕЛЬНЫХ ПЕРСОНАЛЬНЫХ КАЧЕСТВ  
ЛИЧНОСТИ ЧЕЛОВЕКА (OCEAN-AI)**

РУКОВОДСТВО ПРОГРАММИСТА

RU.СНАБ. 00853-01 33 11

ЛИСТОВ 70

Подп. и дата	
Инв. №	
Взам. и	
Подп. и дата	
Инв. №	

2023

### АННОТАЦИЯ

Документ содержит описание компонента библиотеки алгоритмов сильного ИИ RU.СНАБ. 00853-01 13 11 в части алгоритмов интеллектуального анализа поведения человека на основе его мультимодальных данных, обеспечивающих оценивание уровня отдельных персональных качеств личности человека (OCEAN-AI). Аббревиатура OCEAN означает пять оцениваемых компонентом OCEAN-AI персональных качеств личности человека (ПКЛЧ): «Открытость опыту» (от англ. Openness to experience/Openness), «Добросовестность» (от англ. Conscientiousness), «Экстраверсия» (от англ. Extraversion), «Доброжелательность» (от англ. Agreeableness), «Эмоциональная стабильность» (от англ. Non-Neuroticism); AI – искусственный интеллект (от англ. Artificial Intelligence). Компонент OCEAN-AI входит в состав библиотеки алгоритмов сильного ИИ и разработан в рамках плана Исследовательского центра в сфере искусственного интеллекта «Сильный ИИ в промышленности» (ИЦ ИИ) в рамках соглашения с АНО «Аналитический центр при Правительстве Российской Федерации» (ИГК 000000D730321P5Q0002), № 70–2021–00141, с целью создания одномодальных и мультимодальных систем, построенных на различных принципах (экспертных и нейросетевых), т.е. основанных на композитных моделях, для предсказания отдельных ПКЛЧ для решения следующих задач:

- 1) упрощения дистанционного отбора потенциальных кандидатов на профессии;
- 2) прогнозирования предпочтений по выбору промышленных потребительских товаров;
- 3) формирования эффективных рабочих коллективов.

**СОДЕРЖАНИЕ**

1. ОБЩИЕ СВЕДЕНИЯ .....	4
2. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММЫ .....	4
3. ОПИСАНИЕ ПРИКЛАДНЫХ ЗАДАЧ .....	7
4. ХАРАКТЕРИСТИКА ПРОГРАММЫ .....	26
5. ОБРАЩЕНИЕ К ПРОГРАММЕ .....	32
6. ПРОВЕРКА ПРОГРАММЫ .....	55
7. ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ .....	62
8. СООБЩЕНИЯ .....	64
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	68
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ .....	70

## **1. ОБЩИЕ СВЕДЕНИЯ**

Компонент OCEAN-AI библиотеки алгоритмов сильного ИИ RU.СНАБ. 00853-01 33 11 разработан в соответствии с мероприятием М1 «Выполнение опережающих исследований и разработок в области алгоритмов сильного ИИ» (1.1.6-1.1.7 Разработка и испытания экспериментального образца библиотеки алгоритмов сильного ИИ в части интеллектуальной детекции событий и смыслов на основе мультимодальных данных (текст, звук, изображения в разных диапазонах) для задач распознавания режимов работы и неисправностей оборудования) программы ИЦ ИИ «Сильный ИИ в промышленности» в рамках федерального проекта «Искусственный интеллект». OCEAN-AI предназначен для мультимодального анализа группы людей с целью оценивания их персональных качеств личности: «Открытость опыту» (от англ. Openness to experience/Openness), «Добросовестность» (от англ. Conscientiousness), «Экстраверсия» (от англ. Extraversion), «Доброжелательность» (от англ. Agreeableness), «Эмоциональная стабильность» (от англ. Non-Neuroticism). Аббревиатура OCEAN означает 5 оцениваемых персональных качеств, AI – искусственный интеллект (от англ. Artificial Intelligence).

Компонент OCEAN-AI разработан на языке Python в средах разработки Microsoft Visual Studio 2022/PyCharm 2020.2.3/Jupyter Lab 3.4.5 с использованием библиотек: ipython 8.18.1, jupyterlab 3.4.5, tensorflow 2.15.0, keras 2.15.0, Keras-Applications 1.0.8, numpy 1.26.2, scipy 1.11.4, pandas 2.1.3, requests 2.31.0, opensmile 2.5.0, librosa 0.10.1, audioread 3.0.1, scikit-learn 1.3.2, opencv-contrib-python 4.8.1, pymediainfo 6.1.0, mediapipe 0.9.0, liwc 0.5.0, transformers 4.36.0, sentencepiece 0.1.99, torch 2.0.1+cpu, torchaudio 2.0.2+cpu. Допускается использование более новых версий библиотек при условии сохранения текущего функционала. Компонент OCEAN-AI размещен по адресу <https://gitlab.actcognitive.org/itmo-sai-code/oceanai>. Для его использования необходим компилятор Python 3.9/3.10.

## **2. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММЫ**

### **2.1. Назначение программного компонента**

Компонент OCEAN-AI реализует функции сильного ИИ в части комплексирования акустических, визуальных и текстовых признаков, построенных на различных принципах (экспертных и нейросетевых), т.е. компонент OCEAN-AI реализует подходы композитного (гибридного) ИИ для анализа аудиовизуальной и текстовой информации с последующей оценкой ПКЛЧ.

## **2.2. Область применения**

Компонент OCEAN-AI предоставляет оценки ПКЛЧ, которые могут быть применимыми в различных прикладных областях, связанных с промышленностью:

1. HR-процессы. С помощью оценок ПКЛЧ возможно ранжирование потенциальных кандидатов для выполнения профессиональных обязанностей для упрощения отбора высококвалифицированных кадров, а также формирование эффективных рабочих коллективов.

2. Таргетированный маркетинг. Люди со схожими персональными качествами и психотипом личности могут иметь схожие предпочтения, следовательно, возможно моделировать намерения покупателей при выборе определенных промышленных потребительских товаров на основе оценок их ПКЛЧ.

Также оценки ПКЛЧ могут быть применимы и в других прикладных областях: образование, медицина и здравоохранение, сфера развлечений, интеллектуальные персональные ассистенты (умные гаджеты, умные комнаты интеллектуальные пространства и т.д.).

## **2.3. Функциональные условия применения**

Компонент OCEAN-AI имеет следующие функциональные условия применения:

- мультимодальные файлы должны храниться в файлах формата «avi», «mp4» или «mov»;
- текстовая информация с экспертной транскрипцией должна храниться в файлах формата «txt»;
- файлы, содержащие записи одного человека, должны иметь одинаковые имена без учета расширения файлов (т.е. должны быть одноименными);
- продолжительность мультимодальных файлов неограниченна, при этом оптимальная продолжительность для мультимодальных файлов на английском языке составляет 15 секунд, на русском языке – 2,5 секунды;
- площадь области лица в кадре должна составлять не менее 10% от площади кадра;
- количество человек в кадре – 1;
- рекомендованное расположение человека в кадре по пояс (голова и верхняя часть туловища);
- частота кадров в секунду для мультимодальных файлов должна быть не менее 6 и не более 30;

- высота видеокадров должна быть не менее 682 и не более 2160 пикселей (оптимальная высота видеокадра 720 пикселей);
- ширина видеокадров должна быть не менее 406 и не более 3840 пикселей (оптимальная ширина видеокадра 1280 пикселей);
- возраст человека должен быть не менее 16 и не более 60 лет;
- нейросетевые модели для извлечения гипотез предсказаний работают при следующих настройках:

- значения аргументов метода `get_video_union_predictions()` класса `Video` для извлечения визуальных признаков: понижение кадровой частоты (`reduction_fps`) до 5 кадров в секунду; размер окна сегмента сигнала (`window`) - 10 кадров; шаг сдвига окна сегмента сигнала (`step`) – 5 кадров. При этом могут быть заданы другие значения аргументов метода `get_video_union_predictions()` класса `Video`, однако нейросетевые модели для получения оценок ПКЛЧ должны быть замены;

- значения аргументов метода `get_audio_union_predictions()` класса `Audio` для извлечения акустических признаков: частота дискретизации – 44100 Гц; размер окна сегмента сигнала – 2 секунды; шаг сдвига сегмента сигнала – 1 секунда. При этом могут быть заданы другие значения аргументов метода `get_audio_union_predictions()` класса `Audio`, однако нейросетевые модели для получения оценок ПКЛЧ должны быть замены;

- значения аргументов метода `get_text_union_predictions()` класса `Text` для извлечения текстовых признаков: выбор язык – `'ru'` (для русского языка) и `'en'` (для английского языка); использование модуля распознавания речи – `False` (использование ручной транскрипции) или `True` (использование автоматической транскрипции);

- метод `get_acoustic_features()` класса `Audio` не работает при размере окна сегмента аудиосигнала менее 0,05 секунды.

Компонент OCEAN-AI функционирует в среде выполнения программного кода Jupyter Lab 3.4.5 и компиляторе Python 3.9/3.10.

#### 2.4. Технические условия применения

Для запуска компонента OCEAN-AI необходим x64-совместимый персональный компьютер (ПК). ПК должен удовлетворять следующим минимальным характеристикам:

- операционная система – Windows/Linux/macOS;

- компилятор – Python 3.9/3.10;
- количество CPU x количество ядер – 1 x 8;
- тактовая частота CPU, ГГц – 3,8;
- количество GPU x количество ядер – 1 x 3584;
- тактовая частота GPU, ГГц – 1,4;
- оперативная память, Гб – 32;
- дисковая память, Гб - 100.

### **3. ОПИСАНИЕ ПРИКЛАДНЫХ ЗАДАЧ**

#### **3.1. Классы решаемых задач**

С использованием компонента OCEAN-AI потенциально могут быть решены следующие классы решаемых задач:

1. Оценивание ПКЛЧ по голосовым характеристикам.
2. Оценивание ПКЛЧ по лицевым характеристикам.
3. Оценивание ПКЛЧ по текстовым характеристикам разговорной речи человека.
4. Мультимодальное оценивание ПКЛЧ по голосовым, лицевым и текстовым характеристикам.

Также компонент OCEAN-AI предоставляет классы решаемых прикладных задач на основе полученных гипотез предсказаний оценок ПКЛЧ:

1. Ранжирование потенциальных кандидатов для выполнения профессиональных обязанностей:
  - по группам профессий;
  - по профессиональным навыкам.
2. Прогнозирование потребительских предпочтений по выбору промышленных потребительских товаров:
  - на примере характеристик автомобиля;
  - на примере категорий применения мобильного устройства.
3. Формирование эффективных рабочих коллективов:
  - поиск подходящего младшего коллеги;
  - поиск подходящего старшего коллеги.

#### **3.2. Примеры решения задач**

Компонент OCEAN-AI работает:



- с англоязычными данными и использует модели, обученные на корпусе First Impressions V2 [1],
- с русскоязычными данными и использует модели, обученные на корпусе MuPTA [2],
- русскоязычными данными и использует модели, обученные на корпусе First Impressions V2, при этом выполняется автоматический перевод речи с русского языка на английский.

Примеры решения задач для 10 мультимодальных файлов из First Impressions V2 и MuPTA корпусов и трех возможных вариантов работы компонента OCEAN-AI представлены по адресу:

[https://oceanai.readthedocs.io/ru/latest/user\\_guide/samples/pipeline\\_practical\\_tasks.html](https://oceanai.readthedocs.io/ru/latest/user_guide/samples/pipeline_practical_tasks.html).

Ниже представлены примеры решения задач для корпуса First Impressions V2.

3.2.1. Базовая задача: Мультимодальное оценивание ПКЛЧ по голосовым, лицевым и текстовым характеристикам

Исходные данные: 10 мультимодальных файлов формата «mp4» из тестовой выборки корпуса First Impressions V2.

Решение задачи:

```
from oceanai.modules.lab.build import Run

# Создание экземпляра класса
_b5 = Run()

# Настройка ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг

# Формирование аудиомоделей
res_load_model_hc = _b5.load_audio_model_hc()
res_load_model_nn = _b5.load_audio_model_nn()

# Загрузка весов аудиомоделей
url = _b5.weights_for_big5_['audio']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_audio_model_weights_hc(url = url)

url = _b5.weights_for_big5_['audio']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_audio_model_weights_nn(url = url)

# Формирование видеомоделей
res_load_model_hc = _b5.load_video_model_hc(lang='en')
res_load_model_deep_fe = _b5.load_video_model_deep_fe()
res_load_model_nn = _b5.load_video_model_nn()

# Загрузка весов видеомоделей
url = _b5.weights_for_big5_['video']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_video_model_weights_hc(url = url)

url = _b5.weights_for_big5_['video']['fi']['fe']['sberdisk']
res_load_model_weights_deep_fe = _b5.load_video_model_weights_deep_fe(url = url)
```

```

url = _b5.weights_for_big5_['video']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_video_model_weights_nn(url = url)

# Формирование вспомогательных моделей для обработки текстовой информации
_b5.load_text_features()
_b5.setup_translation_model()
_b5.setup_bert_encoder()

# Формирование текстовых моделей
res_load_model_hc = _b5.load_text_model_hc(corpus='fi')
res_load_model_nn = _b5.load_text_model_nn(corpus='fi')

# Загрузка весов текстовых моделей
url = _b5.weights_for_big5_['text']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_text_model_weights_hc(url = url)

url = _b5.weights_for_big5_['text']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_text_model_weights_nn(url = url)

# Формирование модели для мультимодального объединения информации
res_load_model_b5 = _b5.load_av_model_b5()

# Загрузка весов модели для мультимодального объединения информации
url = _b5.weights_for_big5_['avt']['fi']['b5']['sberdisk']
res_load_model_weights_b5 = _b5.load_avt_model_weights_b5(url = url)

PATH_TO_DIR = './video_FI/'
PATH_SAVE_VIDEO = './video_FI/test/'

_b5.path_to_save_ = PATH_SAVE_VIDEO

# Загрузка 10 тестовых мультимодальных файлов из корнуса First Impressions V2
# URL: https://chalearnlap.cvc.uab.cat/dataset/24/description/
domain = 'https://download.sberdisk.ru/download/file/'
tets_name_files = [
    '429713680?token=FqHdMLSSh7zYSZt&filename=_plk5k7PBEG.003.mp4',
    '429713681?token=Hz9b4lQkrLfic33&filename=be0DQawtVKE.002.mp4',
    '429713683?token=EgUXS9Xs8xHm5gz&filename=2d6btbaNdfo.000.mp4',
    '429713684?token=1U26753kmPYdIgt&filename=300gK3CnzW0.003.mp4',
    '429713685?token=LyigAWLTzDNwKJO&filename=300gK3CnzW0.001.mp4',
    '429713686?token=EpFRbCKHyuc4HPu&filename=cLaZxEf1nE4.004.mp4',
    '429713687?token=FNTkwqBr4jOS95l&filename=g24JGYuT74A.004.mp4',
    '429713688?token=qDT95nz7hfm2Nki&filename=JZNMxa3OKHY.000.mp4',
    '429713689?token=noLguEGXDpbckhg&filename=nvlqJbHk_Lc.003.mp4',
    '429713679?token=9L7RQ0hgdJlcek6&filename=4vdJGgZpj4k.003.mp4'
]

for curr_files in tets_name_files:
    _b5.download_file_from_url(url = domain + curr_files, out = False)

# Получение прогнозов
_b5.path_to_dataset_ = PATH_TO_DIR # Директория набора данных
_b5.ext_ = ['.mp4'] # Расширения искоемых файлов

# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk']
_b5.get_avt_predictions(url_accuracy = url_accuracy, lang='en')

```

Результаты оценивания ПКЛЧ по мультимодальному объединению информации представлены на рис. 1 и 2.

	Path	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism
<b>Person ID</b>						
1	2d6btbaNdfo.000.mp4	0.581159	0.628822	0.466609	0.622129	0.553832
2	300gK3CnzW0.001.mp4	0.463991	0.418851	0.41301	0.493329	0.423093
3	300gK3CnzW0.003.mp4	0.454281	0.415049	0.39189	0.485114	0.420741
4	4vdJGgZpj4k.003.mp4	0.588461	0.643233	0.530789	0.603038	0.593398
5	be0DQawtVKE.002.mp4	0.633433	0.533295	0.523742	0.608591	0.588456
6	cLaZxEf1nE4.004.mp4	0.636944	0.542386	0.558461	0.570975	0.558983
7	g24JGYuT74A.004.mp4	0.531518	0.376987	0.393309	0.4904	0.447881
8	JZNMxa3OKHY.000.mp4	0.610342	0.541418	0.563163	0.595013	0.569461
9	nvlqJbHk_Lc.003.mp4	0.495809	0.458526	0.414436	0.469152	0.435461
10	_plk5k7PBEg.003.mp4	0.60707	0.591893	0.520662	0.603938	0.565726

Рисунок 1 – Предсказанные оценки ПКЛЧ, полученные в результате мультимодального оценивания ПКЛЧ.

Рис. 1 показывает следующую информацию: Person ID – порядковый номер человека; Path – название мультимодального файла; Openness («Открытость опыту»), Conscientiousness («Добросовестность»), Extraversion («Экстраверсия»), Agreeableness («Доброжелательность») и Non-Neuroticism («Эмоциональная стабильность») – пять ПКЛЧ, оцениваемые компонентом OCEAN-AI, и соответствующие им гипотезы предсказаний для каждой человека. Практическое применение гипотез предсказаний оценок ПКЛЧ описано в разделах 3.2.2, 3.2.3, 3.2.4.

	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism	Mean
<b>Metrics</b>						
<b>MAE</b>	0.0589	0.0612	0.0864	0.0697	0.0582	0.0669
<b>Accuracy</b>	0.9411	0.9388	0.9136	0.9303	0.9418	0.9331

Рисунок 2 – Показатели оценивания алгоритма мультимодального объединения информации МОИ для 10 тестовых мультимодальных файлов

Рис. 2 показывает следующую информацию: Metrics – показатели оценивания алгоритма МОИ; MAE – средняя абсолютная ошибка; Accuracy – 1-MAE. Показатели оценивания вычисляются независимо для каждого из пяти ПКЛЧ. Mean – среднее значение по всем показателям качества.

Показатели оценивания гипотез предсказаний оценок ПКЛЧ, полученных компонентом OCEAN-AI, показывают, что *mean Accuracy* составляет 93,31% на 10

тестовых мультимодальных файлов, такой результат говорит о надежности компонента OCEAN-AI, так как *mean Accuracy* современных исследований [3, 4] не выше 91,63% для 2000 тестовых мультимодальных файлов из исследовательского корпуса First Impressions V2 [1] при использовании только голосовых, лицевых и текстовых характеристик человека.

3.2.2. Прикладная задача 1. Ранжирование потенциальных кандидатов для выполнения профессиональных обязанностей

Исходные данные: 10 мультимодальных файлов формата «mp4» из тестовой выборки корпуса First Impressions V2.

Пример решения задач расположен по адресу:  
[https://oceanai.readthedocs.io/ru/latest/user\\_guide/notebooks/Pipeline\\_practical\\_task\\_1.html](https://oceanai.readthedocs.io/ru/latest/user_guide/notebooks/Pipeline_practical_task_1.html).

3.2.2.1. Ранжирование потенциальных кандидатов для выполнения профессиональных обязанностей по группам профессий

Решение задачи:

На первом этапе выполняется получение оценок ПКЛЧ по аудио-, видео-, текстовой или мультимодальной информации. Мультимодальное оценивание ПКЛЧ описано в 3.2.1.

На втором этапе выполняется ранжирование кандидатов для выполнения профессиональных обязанностей по группам профессий. В качестве примера рассматривается ранжирование потенциальных кандидатов на основе предсказанных оценок, полученных в 3.2.1, и весовых коэффициентов. Комплексная оценка для каждого кандидата рассчитывается по формуле:

$$S_j = \sum_{i=1}^5 p_{ij} \times w_i \quad (1)$$

где  $S_j$  – комплексная оценка для одного кандидата  $j$ ;  $p_{ij}$  – предсказанная оценка (вычисленная автоматически используемым алгоритмом) для одного ПКЛЧ  $i$  кандидата  $j$ ,  $w_i$  – весовой коэффициент для ПКЛЧ  $i$ .

Весовые коэффициенты устанавливаются программистом компонента OCEAN-AI из предложенной таблицы (`traits_priority_for_professions.csv`) или произвольно, при этом сумма коэффициентов не должна превышать 100.

Загрузка таблицы с весовыми коэффициентами и вызов метода ранжирования потенциальных кандидатов выполняется с помощью кода:

```
# Загрузка датафрейма с весовыми коэффициентами
```

```
url =
'https://download.sberdisk.ru/download/file/478675798?token=ff5fNZVpthQlEV0&filename=traits_pr
iority_for_professions.csv'
traits_priority_for_professions = pd.read_csv(url)

traits_priority_for_professions.index.name = 'ID'
traits_priority_for_professions.index += 1
traits_priority_for_professions.index = traits_priority_for_professions.index.map(str)

# Ранжирование кандидатов на должность инженера-проектировщика
weights = traits_priority_for_professions.iloc[4].values[1:]
weights = list(map(int, weights))

_b5._candidate_ranking(
    weights_openness = weights[0],
    weights_conscientiousness = weights[1],
    weights_extraversion = weights[2],
    weights_agreeableness = weights[3],
    weights_neuroticism = weights[4],
    out = False
)

_b5._save_logs(df = _b5.df_files_ranking_, name = 'candidate_ranking_engineers', out = True)
```

На рис. 3 представлена информация с весовыми коэффициентами для каждого ПКЛЧ, которые различны в зависимости от профессии. В качестве примера представлены весовые коэффициенты для пяти групп профессий: менеджеры/руководители (Managers/executives); профессии в сфере предпринимательства (Entrepreneurship); социальные/некоммерческие профессии (Social/Non profit making professions); профессии государственного сектора (Public sector professions); ученые/исследователи и инженеры (Scientists/researchers, and engineers). Весовые коэффициенты вычислены на основе научных статей [5-7].

	Profession	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism
ID						
1	Managers/executives	15	35	15	30	5
2	Entrepreneurship	30	30	5	5	30
3	Social/Non profit making professions	5	5	35	35	20
4	Public sector professions	15	50	15	15	5
5	Scientists/researchers, and engineers	50	15	5	15	15

Рисунок 3 – Весовые коэффициенты вычислены на основе [5-7]

Рис. 3 показывает следующую информацию: ID – порядковые номера групп профессий; Profession – профессии; Openness, Conscientiousness, Extraversion, Agreeableness и Non-Neuroticism – ПКЛЧ и соответствующие им весовые коэффициенты для каждой профессии.

Для ранжирования кандидатов для выполнения профессиональных обязанностей (на примере инженера-проектировщика) весовые коэффициенты ПКЛЧ устанавливаются

равными весовым коэффициентам группы ID 5. На рис. 4 представлен пример вывода информации с потенциальными кандидатами, ранжированными по убыванию их комплексной оценки (Candidate score).

	Path	OPE	CON	EXT	AGR	NNEU	Candidate score
<b>Person ID</b>							
5	be0DQawtVKE.002.mp4	0.633	0.533	0.524	0.609	0.588	60.246
6	cLaZxEf1nE4.004.mp4	0.637	0.542	0.558	0.571	0.559	59.725
4	4vdJGgZpj4k.003.mp4	0.588	0.643	0.531	0.603	0.593	59.672
10	_plk5k7PBEG.003.mp4	0.607	0.592	0.521	0.604	0.566	59.380
8	JZNMxa3OKHY.000.mp4	0.610	0.541	0.563	0.595	0.569	58.921
1	2d6btbaNdfo.000.mp4	0.581	0.629	0.467	0.622	0.554	58.463
7	g24JGYuT74A.004.mp4	0.532	0.377	0.393	0.490	0.448	48.271
9	nvIqJbHk_Lc.003.mp4	0.496	0.459	0.414	0.469	0.435	47.310
2	300gK3CnzW0.001.mp4	0.464	0.419	0.413	0.493	0.423	45.294
3	300gK3CnzW0.003.mp4	0.454	0.415	0.392	0.485	0.421	44.487

Рисунок 4 – Пример вывода информации с ранжированными потенциальными кандидатами для выполнения профессиональных обязанностей инженера-проектировщика

Рис. 4 показывает следующую информацию: Person ID – порядковый номер человека; Path – имя мультимодального файла; OPE (Openness), CON (Conscientiousness), EXT (Extraversion), AGR (Agreeableness) и NNEU (Non-Neuroticism) – ПКЛЧ и соответствующие им предсказанные оценки для каждого кандидата; Candidate score – комплексная (суммарная) оценка для каждого кандидата. Из рис. 4 следует, что для выполнения профессиональных обязанностей инженера-проектировщика наиболее приоритетным является кандидат Person ID=5 (мультимодальный файл с именем be0DQawtVKE.002.mp4).

### 3.2.2.2. Ранжирование потенциальных кандидатов для выполнения профессиональных обязанностей по профессиональным навыкам

Решение задачи:

На первом этапе выполняется получение оценок ПКЛЧ для аудио-, видео-, текстовой или мультимодальной информации. Мультимодальное оценивание ПКЛЧ описано в 3.2.1.

На втором этапе выполняется ранжирование кандидатов для выполнения профессиональных обязанностей по профессиональным навыкам. В качестве примера рассматривается ранжирование потенциальных кандидатов на основе предсказанных

оценок, полученных в 3.2.1, и коэффициентов корреляции. Для каждого качества  $i$  существует матрица коэффициентов корреляции  $W^i = \begin{pmatrix} w_{11}^i & \dots & w_{1K}^i \\ w_{21}^i & \dots & w_{2K}^i \end{pmatrix}$ , где  $K$  – количество профессиональных навыков. Первая строка матрицы предназначена для оценок ПКЛЧ выше или равных пороговому значению  $T$  и показывает коэффициенты корреляции для оценок «высокого уровня». Вторая строка матрицы предназначена для оценок ПКЛЧ ниже порогового значения  $T$  и показывает коэффициенты корреляции для оценок «низкого уровня». Пороговое значение  $T$  определяет противоположность персональных качеств и устанавливается программистом, но как правило равно 0,5. Таким образом для каждого кандидата и его оценок ПКЛЧ необходимо определить уровень оценки, чтобы подобрать соответствующий уровню коэффициент корреляции согласно формуле:

$$w_j^i = \begin{cases} w_1^i, & p_{ij} \geq T \\ w_2^i, & p_{ij} < T \end{cases} \quad (2)$$

Затем по сумме взвешенных оценок ПКЛЧ для профессиональных навыков вычисляется комплексный (суммарный) вектор оценок навыков для каждого кандидата  $j$  согласно формуле:

$$S_j = \sum_{i=1}^5 p_{ij} \times w_j^i \quad (3)$$

где  $S_j$  – вектор комплексных (суммарных) оценок навыков для кандидата  $j$ , при этом  $S \in \mathbb{R}^{N \times K}$  ( $N$  – количество кандидатов,  $K$  – количество навыков);  $p_{ij}$  – предсказанная оценка (вычисленная автоматически используемым алгоритмом) для  $i$ -го ПКЛЧ кандидата  $j$ ,  $w_j^i$  – весовой коэффициент для  $i$ -го ПКЛЧ кандидата  $j$ .

Профессиональные навыки и их коэффициенты корреляции с ПКЛЧ устанавливаются компонентом OCEAN-AI из предложенной таблицы (professional\_skills.csv). Могут быть установлены коэффициенты корреляций для других профессиональных навыков в виде таблиц, аналогичных professional\_skills.csv.

Загрузка таблицы с коэффициентами корреляции и вызов метода ранжирования потенциальных кандидатов по профессиональным навыкам выполняется с помощью кода:

```
# Загрузка датафрейма с коэффициентами корреляций
url =
'https://download.sberdisk.ru/download/file/478678231?token=0qiZwliLthHWYmV&filename=professional_skills.csv '
df_professional_skills = pd.read_csv(url)

df_professional_skills.index.name = 'ID'
```

```
df_professional_skills.index += 1
df_professional_skills.index = df_professional_skills.index.map(str)

# Ранжирование кандидатов по профессиональным навыкам
_b5._priority_skill_calculation(
    correlation_coefficients = df_professional_skills,
    threshold = 0.5,
    out = False
)

_b5._save_logs(df = _b5.df_files_priority_skill_, name = 'skill_candidate_ranking', out =
True)
```

На рис. 5 представлена информация с профессиональными навыками и их коэффициенты корреляции с ПКЛЧ. В качестве примера представлены четыре профессиональных навыка и их коэффициенты корреляции, представленные в статье [8]:

1. Analytical (Аналитические навыки). Умение эффективно решать новые задачи, требующие глубокого анализа.
2. Interactive (Навыки межличностного общения). Умение убеждать и достигать компромиссов с заказчиками и коллегами.
3. Routine (Способность выполнять рутинную работу). Умение эффективно управлять рутинными задачами, соблюдая точность и внимание к деталям.
4. Non-Routine (Способность выполнять нестандартную работу). Умение реагировать и решать проблемы, не имеющие установленного порядка, проявляя адаптивность и креативные навыки в решении задач.

	Trait	Score_level	Analytical	Interactive	Routine	Non-Routine
ID						
1	Openness	high	0.082	0.348	0.571	0.510
2	Openness	low	0.196	0.152	0.148	0.218
3	Conscientiousness	high	0.994	1.333	1.507	1.258
4	Conscientiousness	low	0.241	0.188	0.191	0.267
5	Extraversion	high	0.169	-0.060	0.258	0.017
6	Extraversion	low	0.181	0.135	0.130	0.194
7	Agreeableness	high	1.239	0.964	1.400	1.191
8	Agreeableness	low	0.226	0.180	0.189	0.259
9	Non-Neuroticism	high	0.636	0.777	0.876	0.729
10	Non-Neuroticism	low	0.207	0.159	0.166	0.238

Рисунок 5 – Предложенные в [8] профессиональные навыки и их коэффициенты корреляции с ПКЛЧ

Рис. 5 показывает следующую информацию: Trait – название каждого ПКЛЧ; Score\_level – уровни оценок ПКЛЧ; Analytical, Interactive, Routine, Non-Routine –



профессиональные навыки и соответствующие ПКЛЧ и уровню оценок коэффициенты корреляции для каждого навыка.

На рис. 6 представлен результат ранжирования кандидатов для выполнения профессиональных обязанностей по профессиональным навыкам с комплексными (суммарными) оценками навыков.

	Path	OPE	CON	EXT	AGR	NNEU	Analytical	Interactive	Routine	Non-Routine
<b>Person ID</b>										
4	4vdJGgZpj4k.003.mp4	0.588	0.643	0.531	0.603	0.593	0.380	0.415	0.561	0.454
1	2d6btbaNdfo.000.mp4	0.581	0.629	0.467	0.622	0.554	0.376	0.427	0.539	0.465
10	_plk5k7PBeg.003.mp4	0.607	0.592	0.521	0.604	0.566	0.367	0.398	0.543	0.439
5	be0DQawtVke.002.mp4	0.633	0.533	0.524	0.609	0.588	0.360	0.389	0.534	0.431
8	JZNMxa3OKHY.000.mp4	0.610	0.541	0.563	0.595	0.569	0.357	0.383	0.528	0.425
6	cLaZxEf1nE4.004.mp4	0.637	0.542	0.558	0.571	0.559	0.350	0.379	0.523	0.421
9	nvlqJbHk_Lc.003.mp4	0.496	0.459	0.414	0.469	0.435	0.096	0.074	0.075	0.107
2	300gK3CnzW0.001.mp4	0.464	0.419	0.413	0.493	0.423	0.093	0.072	0.073	0.104
3	300gK3CnzW0.003.mp4	0.454	0.415	0.392	0.485	0.421	0.091	0.071	0.072	0.102
7	g24JGYuT74A.004.mp4	0.532	0.377	0.393	0.490	0.448	0.082	0.094	0.119	0.136

Рисунок 6 – Пример вывода информации с ранжированными потенциальными кандидатами для выполнения профессиональных обязанностей по профессиональным навыкам

Рис. 6 показывает следующую информацию: Person ID – порядковый номер человека; Path – имя мультимодального файла; OPE (Openness), CON (Conscientiousness), EXT (Extraversion), AGR (Agreeableness) и NNEU (Non-Neuroticism) – ПКЛЧ и соответствующие им предсказанные оценки для каждого кандидата; Analytical, Interactive, Routine, Non-Routine – профессиональные навыки и соответствующие им комплексные (суммарные) оценки навыков. Из рис. 6 следует, что Person ID=4 (мультимодальный файл с именем 4vdJGgZpj4k.003.mp4) имеет наиболее высокие комплексные (суммарные) оценки для двух навыков (Analytical, Routine) из четырех. Тогда как Person ID=1 (мультимодальный файл с именем 2d6btbaNdfo.000.mp4) напротив имеет превосходство по другим навыкам (Interactive, Non-Routine). Этот пример показывает, какой кандидат лучше подходит для выполнения определенной профессиональных обязанностей, а также как целесообразно распределять обязанности между сотрудниками для более продуктивной работы.

3.2.3. Прикладная задача 2: Прогнозирование потребительских предпочтений по выбору промышленных потребительских товаров

Исходные данные: 10 мультимодальных файлов формата «mp4» из тестовой выборки корпуса First Impressions V2.

Пример решения задач расположен по адресу:  
[https://oceanai.readthedocs.io/ru/latest/user\\_guide/notebooks/Pipeline\\_practical\\_task\\_2.html](https://oceanai.readthedocs.io/ru/latest/user_guide/notebooks/Pipeline_practical_task_2.html).

3.2.3.1. Прогнозирование потребительских предпочтений по выбору промышленных потребительских товаров на примере характеристик автомобиля

Решение задачи:

На первом этапе выполняется получение оценок ПКЛЧ для аудио-, видео-, текстовой или мультимодальной информации. Мультимодальное оценивание ПКЛЧ описано в 3.2.1.

На втором этапе выполняется прогнозирование потребительских предпочтений на промышленные потребительские товары. В качестве примера рассматривается прогнозирование потребительских предпочтений на промышленные потребительские товары на основе предсказанных оценок, полученных в 3.2.1, и коэффициентов предпочтений. Потребительские предпочтения рассчитываются по формулам:

$$\bar{p}_j = \begin{cases} p_{ij}, & p_{ij} \geq T \\ -p_{ij}, & p_{ij} < T \end{cases} \quad (4)$$

$$\bar{c}_j = \sum_{i=1}^5 \bar{p}_{ij} \times c_i \quad (5)$$

где  $\bar{p}_j$  – нормализованная матрица предсказанных оценок ПКЛЧ для потребителя  $j$ ;  $T$  – пороговое значение противоположности ПКЛЧ;  $\bar{c}_j$  – взвешенный вектор сумм коэффициентов предпочтений при этом  $\bar{c} \in \mathbb{R}^{N \times M}$  ( $N$  – количество потребителей,  $K$  – количество товаров/услуг или их характеристик, среди которых необходимо определить приоритетность);  $C$  – исходная матрица коэффициентов предпочтений  $C \in \mathbb{R}^{5 \times M}$ . Выбираются  $L$  приоритетов для каждого потребителя  $j$ , чья взвешенная сумма коэффициентов предпочтений больше.

На рис. 7 представлена информация с коэффициентами предпочтений, заимствованными из научной статьи [9]. В качестве примера представлены коэффициенты предпочтений для ПКЛЧ и одиннадцати характеристик легкового автомобиля: производительность (Performance); классические характеристики (Classic car features); элементы люкса (Luxury additions); мода и внимание (Fashion and attention); комфортность (Recreation); технология (Technology); семейный (Family friendly); безопасный и надежный

(Safe and reliable); практичный и простой в использовании (Practical and easy to use); экономичный/низкая стоимость (Economical/low cost); базовые характеристики (Basic features).

	Trait	Performance	Classic car features	Luxury additions	Fashion and attention	Recreation	Technology	Family friendly	Safe and reliable	Practical and easy to use	Economical/low cost	Basic features
ID												
1	Openness	0.020000	-0.033333	-0.030000	-0.050000	0.033333	0.013333	-0.030000	0.136667	0.106667	0.093333	0.006667
2	Conscientiousness	0.013333	-0.193333	-0.063333	-0.096667	-0.096667	0.086667	-0.063333	0.280000	0.180000	0.130000	0.143333
3	Extraversion	0.133333	0.060000	0.106667	0.123333	0.126667	0.120000	0.090000	0.136667	0.043333	0.073333	0.050000
4	Agreeableness	-0.036667	-0.193333	-0.133333	-0.133333	-0.090000	0.046667	-0.016667	0.240000	0.160000	0.120000	0.083333
5	Non-Neuroticism	0.016667	-0.006667	-0.010000	-0.006667	-0.033333	0.046667	-0.023333	0.093333	0.046667	0.046667	-0.040000

Рисунок 7 – Пример вывода информации с предложенными в [9] коэффициентами предпочтений для характеристик легкового автомобиля

Рис. 7 показывает следующую информацию: ID – порядковый номер ПКЛЧ; Trait – одно из ПКЛЧ; имена одиннадцати характеристик легкового автомобиля, перечисленных выше, и значения коэффициентов предпочтений.

Коэффициенты предпочтений устанавливаются компонентом OCEAN-AI из предложенной таблицы (auto\_characteristics.csv) или из других таблиц, аналогичных auto\_characteristics.csv.

Загрузка таблицы с коэффициентами предпочтений и вызов метода ранжирования приоритетности промышленных потребительских товаров выполняется с помощью кода:

```
# Загрузка таблицы с коэффициентами предпочтений
url =
'https://download.sberdisk.ru/download/file/478675818?token=EjflMq0eK8cfn0u&filename=auto_char
acteristics.csv'
df_correlation_coefficients = pd.read_csv(url)
df_correlation_coefficients = pd.DataFrame(
    df_correlation_coefficients.drop(['Style and performance', 'Safety and practicality'],
axis = 1)
)

df_correlation_coefficients.index.name = 'ID'
df_correlation_coefficients.index += 1
df_correlation_coefficients.index = df_correlation_coefficients.index.map(str)

# Прогнозирование потребительский предпочтений
_b5._priority_calculation(
    correlation_coefficients = df_correlation_coefficients,
    col_name_ocean = 'Trait',
    threshold = 0.55,
    number_priority = 3,
    number_importance_traits = 3,
    out = False
)
_b5._save_logs(df = _b5.df_files_priority_, name = 'auto_characteristics_priorities', out =
True)
_b5.df_files_priority_
```

На рис. 8 представлена информация с потенциальными тремя наиболее приоритетными характеристиками легкового автомобиля для каждого человека (Priority\_1 - Priority\_3) в порядке убывания приоритетности. Количество приоритетных характеристик

устанавливается программистом. На рис. 8 представлены три наиболее важные качества человека (Trait\_Importance\_1 - Trait\_Importance\_3) по убыванию их важности, которые повлияли на приоритетность характеристик легкового автомобиля у каждого человека. Количество наиболее важных качеств устанавливается программистом.

Person ID	Path	OPE	CON	EXT	AGR	NNEU	Priority 1	Priority 2	Priority 3	Trait importance 1	Trait importance 2	Trait importance 3
1	2d6btbaNdf0.000.mp4	0.581	0.629	0.467	0.622	0.554	Safe and reliable	Practical and easy to use	Economical/low cost	Conscientiousness	Agreeableness	Openness
2	300gK3CnzW0.001.mp4	0.464	0.419	0.413	0.493	0.423	Classic car features	Fashion and attention	Luxury additions	Agreeableness	Conscientiousness	Openness
3	300gK3CnzW0.003.mp4	0.454	0.415	0.392	0.485	0.421	Classic car features	Fashion and attention	Luxury additions	Agreeableness	Conscientiousness	Openness
4	4vdJGgZpj4k.003.mp4	0.588	0.643	0.531	0.603	0.593	Safe and reliable	Practical and easy to use	Economical/low cost	Conscientiousness	Agreeableness	Openness
5	be0DQawtVtE.002.mp4	0.633	0.533	0.524	0.609	0.588	Practical and easy to use	Safe and reliable	Economical/low cost	Agreeableness	Openness	Non-Neuroticism
6	cLaZxF1nE4.004.mp4	0.637	0.542	0.558	0.571	0.559	Safe and reliable	Economical/low cost	Practical and easy to use	Agreeableness	Openness	Extraversion
7	g24JGYuT74A.004.mp4	0.532	0.377	0.393	0.490	0.448	Classic car features	Fashion and attention	Luxury additions	Agreeableness	Conscientiousness	Openness
8	JZNMxa3OKHY.000.mp4	0.610	0.541	0.563	0.595	0.569	Safe and reliable	Economical/low cost	Practical and easy to use	Agreeableness	Openness	Extraversion
9	nvlqJbHk_Lc.003.mp4	0.496	0.459	0.414	0.469	0.435	Classic car features	Fashion and attention	Luxury additions	Agreeableness	Conscientiousness	Openness
10	_plk5k7P8Eg.003.mp4	0.607	0.592	0.521	0.604	0.566	Safe and reliable	Practical and easy to use	Economical/low cost	Conscientiousness	Agreeableness	Openness

Рисунок 8 – Пример вывода информации с ранжированными предпочтениями на примере характеристик легкового автомобиля для 10 человек

Из рис. 8 следует, что наиболее приоритетными характеристиками легкового автомобиля для 10 рассмотренных человек являются:

1. По первому приоритету (Priority 1) – «Безопасность и надежность» (Safe and reliable) в 5 из 10 случаев.
2. По второму приоритету (Priority 2) – «Практичность и простота использования» (Practical and easy to use) в 4 из 10 случаев и «Мода и внимание» (Safe and reliable) в 4 из 10 случаев.
3. По третьему приоритету (Priority 3) – «Экономичность и низкая стоимость» (Economical/low cost) в 4 из 10 случаев и «Элементы люкса» (Luxury additions) в 4 из 10 случаев.

За выбор данных характеристик отвечают преимущественно три качества: Добросовестность (Conscientiousness), Доброжелательность (Agreeableness) и Открытость опыту (Openness).

3.2.3.2. Прогнозирование потребительских предпочтений по выбору промышленных потребительских товаров на примере категорий применения мобильного устройства

Решение задачи:

Решение задачи аналогично 3.2.3.1 только меняются коэффициенты предпочтений на характеристики промышленных товаров. На рис. 9 представлена информация с коэффициентами предпочтений, заимствованными из научной статьи [10]. В качестве

примера представлены коэффициенты предпочтений для ПКЛЧ и шестнадцати категорий применения мобильного устройства (application categories): общение (Communication); экшен игры (Game Action); настольные игры (Game Board); азартные игры (Game Casino); образовательные (развивающие) игры (Game Educational); игры-симуляторы (Game Simulation); игры-викторины (Game Trivia); развлечения (Entertainment); финансы (Finance); здоровье и фитнес (Health and Fitness); медиа и видео (Media and Video); музыка и аудио (Music and Audio); новости и журналы (News and Magazines); персонализация (Personalisation); путешествия и местный досуг (Travel and Local); погода (Weather).

ID	Trait	Communication	Game Action	Game Board	Game Casino	Game Educational	Game Simulation	Game Trivia	Entertainment	Finance	Health and Fitness	Media and Video	Music and Audio	News and Magazines	Personalisation	Travel and Local	Weather
1	Openness	0.118	0.056	0.079	0.342	0.027	0.104	0.026	0.000	0.006	0.002	0.000	0.000	0.001	0.004	0.002	0.004
2	Conscientiousness	0.119	0.043	0.107	0.448	0.039	0.012	0.119	0.000	0.005	0.001	0.000	0.002	0.002	0.001	0.001	0.003
3	Extraversion	0.246	0.182	0.211	0.311	0.102	0.165	0.223	0.001	0.003	0.000	0.001	0.001	0.001	0.004	0.009	0.003
4	Agreeableness	0.218	0.104	0.164	0.284	0.165	0.122	0.162	0.000	0.003	0.001	0.000	0.002	0.002	0.001	0.004	0.003
5	Non-Neuroticism	0.046	0.047	0.125	0.515	0.272	0.179	0.214	0.002	0.030	0.001	0.000	0.005	0.003	0.008	0.004	0.007

Рисунок 9 – Пример вывода информации с предложенными в [10] коэффициентами предпочтений для категорий применения мобильного устройства

Рис. 9 показывает следующую информацию: ID – порядковый номер ПКЛЧ; Trait – одно из ПКЛЧ; имена шестнадцати категорий применения мобильного устройства, перечисленных выше, и значения коэффициентов предпочтений.

Коэффициенты предпочтений устанавливаются компонентом OCEAN-AI из предложенной таблицы (device\_characteristics.csv) или других таблиц, аналогичных device\_characteristics.csv.

Загрузка таблицы с коэффициентами предпочтений и вызов метода ранжирования приоритетности промышленных потребительских товаров выполняется с помощью кода:

```
# Загрузка таблицы с коэффициентами предпочтений
url =
'https://download.sberdisk.ru/download/file/478676690?token=7KcAxPqMpWiYQnx&filename=device_ch
aracteristics.csv'
df_correlation_coefficients = pd.read_csv(url)

df_correlation_coefficients.index.name = 'ID'
df_correlation_coefficients.index += 1
df_correlation_coefficients.index = df_correlation_coefficients.index.map(str)

# Прогнозирование потребительский предпочтений
_b5._priority_calculation(
    correlation_coefficients = df_correlation_coefficients,
    col_name_ocean = 'Trait',
    threshold = 0.55,
    number_priority = 3,
    number_importance_traits = 3,
    out = False
)
_b5._save_logs(df = _b5.df_files_priority_, name = 'device_characteristics_priorities', out =
True)
_b5.df_files_priority_
```

На рис. 10 представлена информация с потенциальными тремя наиболее приоритетными категориями применения мобильного устройства для каждого человека (Priority\_1 - Priority\_3) в порядке убывания приоритетности. Количество приоритетных характеристик устанавливается программистом. На рис. 10 представлены три наиболее важные качества человека (Trait\_Importance\_1 - Trait\_Importance\_3) по убыванию их важности, которые повлияли на приоритетность категорий применения мобильного устройства у каждого человека. Количество наиболее важных качеств устанавливается программистом.

	Path	OPE	CON	EXT	AGR	NNEU	Priority 1	Priority 2	Priority 3	Trait importance 1	Trait importance 2	Trait importance 3
Person ID												
1	2d6btbaNdf0.000.mp4	0.581	0.629	0.467	0.622	0.554	Game Casino	Game Educational	Game Trivia	Non-Neuroticism	Conscientiousness	Agreeableness
2	300gK3CnzW0.001.mp4	0.464	0.419	0.413	0.493	0.423	Media and Video	Entertainment	Health and Fitness	Conscientiousness	Agreeableness	Extraversion
3	300gK3CnzW0.003.mp4	0.454	0.415	0.392	0.485	0.421	Media and Video	Entertainment	Health and Fitness	Conscientiousness	Agreeableness	Extraversion
4	4vdJGgZpj4k.003.mp4	0.588	0.643	0.531	0.603	0.593	Game Casino	Game Educational	Game Trivia	Non-Neuroticism	Conscientiousness	Agreeableness
5	be0DQawtVkE.002.mp4	0.633	0.533	0.524	0.609	0.588	Game Casino	Game Educational	Game Simulation	Non-Neuroticism	Agreeableness	Openness
6	cLaZxEf1nE4.004.mp4	0.637	0.542	0.558	0.571	0.559	Game Casino	Game Simulation	Game Educational	Non-Neuroticism	Agreeableness	Extraversion
7	g24JGYuT74A.004.mp4	0.532	0.377	0.393	0.490	0.448	Media and Video	Entertainment	Health and Fitness	Conscientiousness	Agreeableness	Extraversion
8	JZNMxa3OKHY.000.mp4	0.610	0.541	0.563	0.595	0.569	Game Casino	Game Simulation	Game Educational	Non-Neuroticism	Agreeableness	Extraversion
9	nvlqJbHk_Lc.003.mp4	0.496	0.459	0.414	0.469	0.435	Media and Video	Entertainment	Health and Fitness	Conscientiousness	Agreeableness	Extraversion
10	_plk5k7PBEG.003.mp4	0.607	0.592	0.521	0.604	0.566	Game Casino	Game Educational	Game Trivia	Non-Neuroticism	Agreeableness	Conscientiousness

Рисунок 10 – Пример вывода информации с ранжированными предпочтениями на примере категорий применения мобильного устройства для 10 человек

Из рис. 10 следует, что наиболее приоритетными категориями применения мобильного устройства для 10 рассмотренных человек являются:

1. По первому приоритету (Priority 1) – «Азартные игры» (Game Casino) в 6 из 10 случаев.
2. По второму приоритету (Priority 2) – «Образовательные игры» (Game Educational) в 4 из 10 случаев и «Развлечения» (Entertainment) в 4 из 10 случаев.
3. По третьему приоритету (Priority 3) – «Здоровье и фитнес» (Health and Fitness) в 4 из 10 случаев.

За выбор данных характеристик отвечают преимущественно три качества: Добросовестность (Conscientiousness), Доброжелательность (Agreeableness) и Эмоциональная стабильность (Non-Neuroticism).

### 3.2.4. Прикладная задача 3: Формирование эффективных рабочих коллективов

Исходные данные: 10 мультимодальных файлов формата «mp4» из тестовой выборки корпуса First Impressions V2.

Пример решения задач расположен по адресу:  
[https://oceanai.readthedocs.io/ru/latest/user\\_guide/notebooks/Pipeline\\_practical\\_task\\_3.html](https://oceanai.readthedocs.io/ru/latest/user_guide/notebooks/Pipeline_practical_task_3.html).

#### 3.2.4.1. Поиск младшего коллеги

Решение задачи:

На первом этапе выполняется получение оценок ПКЛЧ для аудио-, видео-, текстовой или мультимодальной информации. Мультимодальное оценивание ПКЛЧ описано в 3.2.1.

На втором этапе выполняется поиск коллеги. В качестве примера выполняется поиск коллеги на основе предсказанных оценок, полученных в 3.2.1, и двух коэффициентов корреляции совместимости двух людей для каждого ПКЛЧ. Эти коэффициенты должны показывать, как изменится оценка качества одного человека, если она будет больше или меньше оценки качества другого человека. Коэффициенты корреляции совместимости людей представлены в виде матрицы  $R = \begin{pmatrix} r_{11} & \dots & r_{15} \\ r_{21} & \dots & r_{25} \end{pmatrix}$ , где первая строка показывает коэффициенты корреляции совместимости людей по ПКЛЧ для случая, когда оценки качеств целевого коллеги (того, для кого проводится анализ, при условии, что мы ищем младшего коллегу), выше, чем у тестового коллеги. Вторая строка показывает коэффициенты для противоположной ситуации. Взвешенная суммарная оценка для младшего коллеги рассчитывается по формулам:

$$\bar{p}_{ij} = \begin{cases} p_{ij} \times r_{1i}, & p_{ij} > t_i \\ p_{ij} \times \tau, & p_{ij} = t_i \\ p_{ij} \times r_{2i}, & p_{ij} < t_i \end{cases} \quad (6)$$

$$S_j = \sum_{i=1}^5 \bar{p}_{ij} \quad (7)$$

где  $\bar{p}_{ij}$  – нормализованная матрица предсказанных оценок;  $p_{ij}$  – предсказанные оценки для качества  $i$  коллеги  $j$ ;  $t_i$  – экспертные оценки для качества  $i$  целевого коллеги  $t$ ;  $S_j$  – взвешенная сумма оценок для коллеги  $j$ ;  $R$  – матрица коэффициентов корреляции;  $\tau$  – коэффициент корреляции при равенстве оценок ПКЛЧ.

Коэффициенты корреляции совместимости людей устанавливаются компонентом OCEAN-AI из предложенной таблицы (colleague\_ranking.csv) или других таблиц, аналогичных colleague\_ranking.csv.

Загрузка таблицы с коэффициентами корреляции совместимости людей и вызов метода поиска коллег выполняется с помощью кода:

```
# Загрузка датафрейма с коэффициентами корреляции
```



```
url =
'https://download.sberdisk.ru/download/file/478675819?token=LuB7L1QsEY0UuSs&filename=colleague
_ranking.csv'
df_correlation_coefficients = pd.read_csv(url)

df_correlation_coefficients.index.name = 'ID'
df_correlation_coefficients.index += 1
df_correlation_coefficients.index = df_correlation_coefficients.index.map(str)

# Оценки качеств целевого коллеги (того, для кого проводится поиск) – устанавливаются
программистом
target_scores = [0.527886, 0.522337, 0.458468, 0.51761, 0.444649]

# Поиск кандидата
_b5._colleague_ranking(
    correlation_coefficients = df_correlation_coefficients,
    target_scores = target_scores,
    colleague = 'minor', # для поиска младшего коллеги
    equal_coefficients = 0.5,
    out = False
)

_b5._save_logs(df = _b5.df_files_priority_skill_, name = 'minor_colleague_ranking', out =
True)
```

На рис. 11 представлена информация по коэффициентам совместимости двух людей на каждое ПКЛЧ, предложенные в статье [11]:

	Score_comparison	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism
ID						
1	higher	-0.0602	0.0471	-0.1070	-0.0832	0.190
2	lower	-0.1720	-0.1050	0.0772	0.0703	-0.229

Рисунок 11 – Пример вывода информации с предложенными в [11] коэффициентами совместимости двух людей

Рис. 11 показывает следующую информацию: ID – порядковый номер уровня сравнения оценок; Score\_comparison – категория сравнимости оценок ПКЛЧ; Openness, Conscientiousness, Extraversion, Agreeableness и Non-Neuroticism – ПКЛЧ и значения коэффициентов корреляции совместимости двух людей.

На рис. 12 представлена информация поиска подходящего младшего коллеги для целевого коллеги с оценками ПКЛЧ: Openness=0,527886, Conscientiousness=0,522337, Extraversion=0,458468, Agreeableness=0,51761, Non-Neuroticism=0,444649.



	Path	OPE	CON	EXT	AGR	NNEU	Match
Person ID							
9	nvlqJbHk_Lc.003.mp4	0.496	0.459	0.414	0.469	0.435	-0.009
3	300gK3CnzW0.003.mp4	0.454	0.415	0.392	0.485	0.421	-0.010
2	300gK3CnzW0.001.mp4	0.464	0.419	0.413	0.493	0.423	-0.013
8	JZNMxa3OKHY.000.mp4	0.610	0.541	0.563	0.595	0.569	-0.207
6	cLaZxEf1nE4.004.mp4	0.637	0.542	0.558	0.571	0.559	-0.211
1	2d6btbaNdfo.000.mp4	0.581	0.629	0.467	0.622	0.554	-0.213
10	_plk5k7PBEg.003.mp4	0.607	0.592	0.521	0.604	0.566	-0.213
5	be0DQawtVkE.002.mp4	0.633	0.533	0.524	0.609	0.588	-0.216
4	4vdJGgZpj4k.003.mp4	0.588	0.643	0.531	0.603	0.593	-0.221
7	g24JGYuT74A.004.mp4	0.532	0.377	0.393	0.490	0.448	-0.259

Рисунок 12 – Пример вывода информации по результатам поиска подходящего младшего коллеги из 10 человек

Рис. 12 показывает следующую информацию: Person ID – порядковый номер человека; Path – имя мультимодального файла; OPE (Openness), CON (Conscientiousness), EXT (Extraversion), AGR (Agreeableness) и NNEU (Non-Neuroticism) – ПКЛЧ и соответствующие им предсказанные оценки для каждого кандидата; Match – взвешенная суммарная оценка для каждого человека. Из рис. 12 следует, что по результатам поиска подходящим младшим коллегой является Person ID 9 (мультимодальный файл с именем nvlqJbHk\_Lc.003.mp4).

#### 3.2.4.2. Поиск старшего коллеги

Решение задачи:

Решение задачи аналогично 3.2.4.1 с некоторыми поправками. Коэффициенты корреляции совместимости людей  $R = \begin{pmatrix} r_{11} & \dots & r_{15} \\ r_{21} & \dots & r_{25} \end{pmatrix}$  для поиска старшего коллеги считается противоположно младшему коллеге, первая строка показывает коэффициенты корреляции совместимости людей по ПКЛЧ для случая, когда оценки качеств целевого коллеги (того, для кого проводится анализ, при условии, что мы ищем старшего коллегу для работы), ниже, чем у тестового коллеги. Вторая строка показывает коэффициенты для противоположной ситуации. Взвешенная суммарная оценка для старшего коллеги рассчитывается аналогично формуле (6), однако нормализованная матрица предсказанных оценок рассчитывается по формуле:

$$\bar{P}_{ij} = \begin{cases} p_{ij} \times r_{2i}, & p_{ij} > t_i \\ p_{ij} \times \tau, & p_{ij} = t_i \\ p_{ij} \times r_{1i}, & p_{ij} < t_i \end{cases} \quad (7)$$

Загрузка таблицы с коэффициентами корреляции совместимости людей останется без изменений, а вызов метода поиска коллег изменится в соответствии с кодом:

```
# Поиск кандидата
_b5._colleague_ranking(
    correlation_coefficients = df_correlation_coefficients,
    target_scores = target_scores,
    colleague = 'major', # для поиска старшего коллеги
    equal_coefficients = 0.5,
    out = False
)

_b5._save_logs(df = _b5.df_files_priority_skill_, name = 'major_colleague_ranking', out =
True)
```

На рис. 13 представлена информация поиска подходящего старшего коллеги для целевого коллеги с оценками ПКЛЧ: Openness=0,527886, Conscientiousness=0,522337, Extraversion=0,458468, Agreeableness=0,51761, Non-Neuroticism=0,444649.

	Path	OPE	CON	EXT	AGR	NNEU	Match
<b>Person ID</b>							
7	g24JGYuT74A.004.mp4	0.532	0.377	0.393	0.490	0.448	0.078
4	4vdJGgZpj4k.003.mp4	0.588	0.643	0.531	0.603	0.593	0.001
1	2d6btbaNdfo.000.mp4	0.581	0.629	0.467	0.622	0.554	-0.002
10	_plk5k7PBEg.003.mp4	0.607	0.592	0.521	0.604	0.566	-0.007
5	be0DQawtVKE.002.mp4	0.633	0.533	0.524	0.609	0.588	-0.008
8	JZNMxa3OKHY.000.mp4	0.610	0.541	0.563	0.595	0.569	-0.013
6	cLaZxEf1nE4.004.mp4	0.637	0.542	0.558	0.571	0.559	-0.014
3	300gK3CnzW0.003.mp4	0.454	0.415	0.392	0.485	0.421	-0.154
2	300gK3CnzW0.001.mp4	0.464	0.419	0.413	0.493	0.423	-0.154
9	nvlqJbHk_Lc.003.mp4	0.496	0.459	0.414	0.469	0.435	-0.168

Рисунок 13 – Пример вывода информации по результатам поиска подходящего старшего коллеги из 10 человек

Рис. 13 показывает следующую информацию: Person ID – порядковый номер человека; Path – имя мультимодального файла; OPE (Openness), CON (Conscientiousness), EXT (Extraversion), AGR (Agreeableness) и NNEU (Non-Neuroticism) – ПКЛЧ и соответствующие им предсказанные оценки для каждого кандидата; Match – взвешенная суммарная оценка для каждого человека. Из рис. 13 следует, что по результатам поиска подходящим старшим коллегой является Person ID 7 (мультимодальный файл с именем g24JGYuT74A.004.mp4).

Таким образом, библиотека OCEAN-AI предоставляет инструменты для решения следующих прикладных задач:

1. Анализа персональных качеств личности кандидатов и их соответствия требованиям должности, что может значительно улучшить процесс подбора персонала и помочь в принятии более объективных и систематизированных решений при ранжировании кандидатов.
2. Анализа персональных качеств личности потребителей, что полезно для предсказания того, что их заинтересует. Это позволит компаниям более точно адаптировать свои промышленные товары/услуги к предпочтениям потребителей, делая их более уникальными и персонализированными.
3. Поиска подходящих коллег по их персональным качествам личности, что может помочь в формировании эффективных рабочих коллективов, улучшении коммуникации и сокращении конфликтов в коллективе.

#### **4. ХАРАКТЕРИСТИКА ПРОГРАММЫ**

##### **4.1. Режимы работы ключевых алгоритмов**

Компонент OCEAN-AI состоит из четырех ключевых алгоритмов:

1. Алгоритм анализа аудиоинформации (ААИ) для оценивания ПКЛЧ по голосовым характеристикам.
2. Алгоритм анализа видеоинформации (АВИ) для оценивания ПКЛЧ по лицевым характеристикам.
3. Алгоритм анализа текстовой информации (АТИ) для оценивания ПКЛЧ по текстовым характеристикам разговорной речи человека.
4. Алгоритм мультимодального объединения информации (МОИ) для оценивания ПКЛЧ по голосовым, лицевым и текстовым характеристикам.

Компонент OCEAN-AI позволяет обрабатывать мультимодальные файлы, содержащие только аудио-, видео- или текстовую информацию, а также все три источника информации. Компонент OCEAN-AI не анализирует статические изображения. Алгоритмы ААИ, АВИ и АТИ реализует подходы композитного (гибридного) ИИ.

В работе [12] исследованы различные методы представления признаков для распознавания нескольких паралингвистических явлений речи, среди которых и признаки, извлекаемые с помощью инструментария openSMILE [13], log-Mel спектрограммы и MFCC. Разрабатываемые методы, представленные в [12], применяются в компоненте OCEAN-AI алгоритма ААИ для оценивания ПКЛЧ. Log-Mel спектрограмм используются для

представления нейросетевых акустических признаков. На рис. 2 статьи [12], детально проиллюстрирован метод на основе log-Mel спектрограмм. Признаки openSMILE используются для представления экспертных акустических признаков, пример работы с признаками представлен на рис. 3 статьи [12]. На рис. 3 статьи [2] впервые представлен метод извлечения нейросетевых признаков и получения гипотез предсказаний по ним на базе log-mel спектрограмм и VGG-16 для оценивания ПКЛЧ.

Для обнаружения графических областей лиц людей на динамических изображениях в условиях их повышенной окклюзии в алгоритме АВИ используется метод, предложенный в [14]. В основе метода лежит нейросетевая каскадная архитектура модели с остаточными слоями, что позволяет быстро и с высоким показателем качества обнаруживать графические области лица. Подробное описание реализации метода представлено в [14], работа метода проиллюстрирована на рис. 4 статьи [14]. Из обнаруженной графической области лица затем извлекаются лицевые нейросетевые признаки.

Последовательность эмоциональных реакций разных людей на одни и те же стимулы отражает их различия в ПКЛЧ. В связи с этим в алгоритме АВИ для извлечения нейросетевых признаков используется эмоциональная модель, предложенная в [15]. Модель продемонстрировала высокие показатели качества в задаче распознавания эмоций человека, которые достигнуты за счет экспериментального подбора следующих гиперпараметров:

1. Графика скорости обучения.
2. Алгоритма оптимизации.
3. Архитектуры нейросетевой модели.
4. Метода взвешивания несбалансированных классов.
5. Метода регуляризации.
6. Количества нейронов в полносвязном слое.
7. Техники аугментации данных.

Экспериментальные результаты по подбору гиперпараметров, а также сравнение показателей качества разработанного метода с современными методами представлены в табл. 4 и 5 статьи [15]. На рис. 1, 5 и 7 статьи [16] впервые представлен метод извлечения нейросетевых признаков и получения гипотез предсказаний по ним на базе нейросетевого экстрактора признаков [15] и трех контекстных моделей для оценивания ПКЛЧ.

Алгоритмы ААИ, АВИ и АТИ работают с экспертными и нейросетевыми признаками. Для объединения гипотез предсказаний, полученных от разных признаков и модальностей, используется метод на основе полносвязной нейронной сети [17]. Данный

метод позволяет объединять аудио- и видеoinформацию на позднем уровне. Метод показал свою эффективность не только в задаче оценивания ПКЛЧ, но и при решении других задач, связанных с одновременным мультимодальным анализом аудио- и видеoinформации, в частности, распознавания психоэмоциональных состояний [12] и аудиовизуального распознавания речи [17].

В табл. 1 представлены экспериментальные исследования алгоритмов ИИ, апробированных на корпусах First Impressions V2 [1] и MuPTA [2]. Корпус First Impressions V2 поделен на три выборки: обучающая (6000 мультимодальных файлов), валидационная (2000), тестовая (2000). Корпус MuPTA также поделен на три выпорки: обучающая (2322), валидационная (774) и тестовая (774). В процессе обучения моделей использовались обучающие и валидационные выборки. На основе валидационных выборок вычислялась априорная оценка алгоритмов, при этом на прогнозирование ПКЛЧ не оказывалось значительное внешнее воздействие. В процессе апостериорной оценки прогнозирование ПКЛЧ для тестовых выборок выполнялось с учетом оптимизированных на валидационных выборках параметров моделей и весов. Достигнутые в табл. 1 значения показателей качества (средняя абсолютная ошибка  $MAE$ , точность  $ACC$  и коэффициент корреляции согласования  $CCC$ ) сравниваются с современными алгоритмами [2, 3]. Показатели качества рассчитываются по формулам:

$$MAE_i = \frac{1}{N} \sum_{j=1}^N |t_{ij} - p_{ij}|, \quad (8)$$

$$ACC_i = 1 - MAE_i, \quad (9)$$

$$mACC = \frac{1}{M} \sum_{i=1}^5 ACC_i, \quad (10)$$

$$CCC = \frac{2 \cdot \sigma_t \cdot \sigma_p}{\sigma_t^2 + \sigma_p^2 + (\mu_t + \mu_p)^2} \quad (11)$$

где  $i = 1, \dots, 5$  – порядковый номер ПКЛЧ;  $N$  – количество мультимодальных файлов;  $j = 1, \dots, N$ ;  $t_{ij}$  – целевая оценка (вычисленная экспертами для данных исследовательского корпуса, целевые оценки можно считать эталоном) одного образца  $j$ , принадлежащего ПКЛЧ  $i$ ;  $p_{ij}$  – прогнозная оценка (вычисленная автоматически используемым алгоритмом) одного образца  $j$ , принадлежащего ПКЛЧ  $i$ ;  $\mu_t$  и  $\mu_p$  – среднее значение целевых и

прогнозных оценок для ПКЛЧ всех мультимодальных файлов;  $\sigma_t$  и  $\sigma_p$  – среднее отклонения целевых и прогнозных оценок для ПКЛЧ всех мультимодальных файлов. Показатель *mean Accuracy* являются базовыми для автоматического оценивания ПКЛЧ на корпусе First Impressions V2 [1]. Данный показатель был установлен авторами корпуса и организаторами соревнований в рамках международной конференции ECCV 2016 [18], что позволяет сравнить результаты показателей качества разработанных алгоритмов компонента OCEAN-AI с результатами современных алгоритмов.

Таблица 1 – Экспериментальные исследования алгоритмов ИИ

Признаки	First Impressions V2				MuPTA			
	Априорная оценка		Апостериорная оценка		Априорная оценка		Апостериорная оценка	
	mACC	CCC	mACC	CCC	mACC	CCC	mACC	CCC
Анализ аудиоинформации (ААИ)								
Экспертные	0,899	0,382	0,897	0,382	0,902	0,574	0,891	0,460
Нейросетевые	0,908	0,531	0,906	0,523	<b>0,906</b>	0,576	0,892	0,473
Объединение	<b>0,910</b>	<b>0,585</b>	<b>0,907</b>	<b>0,574</b>	0,903	<b>0,599</b>	<b>0,898</b>	<b>0,524</b>
Анализ видеоинформация (АВИ)								
Экспертные	0,910	0,543	0,908	0,556	0,896	0,576	0,893	0,521
Нейросетевые	0,914	0,646	0,913	0,648	0,901	0,638	0,896	0,561
Объединение	<b>0,916</b>	<b>0,659</b>	<b>0,915</b>	<b>0,661</b>	<b>0,906</b>	<b>0,647</b>	<b>0,897</b>	<b>0,574</b>
Анализ текстовой информации (АТИ)								
Экспертные	0,888	0,236	0,888	0,251	0,904	0,588	0,890	0,466
Нейросетевые	0,889	0,240	0,889	0,260	0,903	0,601	0,891	0,485
Объединение	<b>0,890</b>	<b>0,255</b>	<b>0,889</b>	<b>0,274</b>	<b>0,907</b>	<b>0,604</b>	<b>0,894</b>	<b>0,490</b>
Мультимодальное объединение информации (МОИ)								
Аудио и видео	0,918	0,682	0,917	0,682	0,904	0,633	0,901	0,584
Аудио, видео, текст	<b>0,930</b>	<b>0,740</b>	<b>0,928</b>	<b>0,734</b>	<b>0,918</b>	<b>0,708</b>	<b>0,907</b>	<b>0,637</b>
Аудио, видео, текст [3]	-	-	0,915	-	-	-	-	-
Аудио, видео, текст [4]	-	-	0,916	-	-	-	-	-
Аудио, видео [2]	-	-	-	-	0,903	0,672	0,899	0,614

Результаты экспериментов показывают, что на тестовых выборках (апостериорная оценка) достигаются аналогичные показатели качества алгоритмов, как и на валидационных выборках (априорная оценка). Показатель качества алгоритмов CCC демонстрирует, что в зависимости от анализируемой информации (аудио, видео или текст)

и мультимодального объединения, абсолютные приросты  $CCC$  (в %) на валидационной и тестовой выборках составляют 50,4% (0,236 против 0,740) и 48,3% (0,251 против 0,734) для корпуса First Impressions V2, соответственно. Для корпуса MuPTA приросты составили 13,4% (0,574 против 0,708) и 17,7% (0,460 против 0,637), соответственно. Такой результат свидетельствует о надежности предсказания оценок ПКЛЧ разработанными алгоритмами. Базовый показатель качества ( $mACC$ ) разработанных алгоритмов превосходят показатели качества современных алгоритмов [2, 3, 4] для обоих корпусов. В зависимости от операционной системы (Windows/Linux/macOS) гипотезы предсказаний оценок ПКЛЧ отличаются на тысячные, что влияет на показатели качества алгоритмов.

Экспериментальные исследования ключевых алгоритмов производились с использованием открытого корпуса First Impressions V2 [1]. Мультимодальные файлы, представленные в корпусе, имеют различные значения ширины (от 406 до 1280 пикселей), высоты кадров (от 682 до 720 пикселей) и отношения размера лица человека к размеру кадра (от 2 до 40%), различную частоту кадров (от 7 до 30 кадров в секунду), распределение записей с этими параметрами в корпусе неоднородное. Достаточно очевидно, что чем выше оптическое разрешение кадров, больше отношение площади лица человека к площади всего кадра, а также выше частота видеок кадров, тем потенциально более информативной является запись и гипотезы предсказаний оценок ПКЛЧ могут быть более достоверны. Мультимодальные данные корпуса MuPTA [2] собирались в офисных условиях, параметры корпуса однородные с незначительными вариациями в освещении и фоне, а также удаленности лица человека от камеры.

#### 4.2. Порядок оценивания качества алгоритмов.

Для оценивания качества алгоритмов используются два показателя:  $mMAE$ ,  $mACC$ , где  $mMAE$  – противоположное значение  $mACC$ . Алгоритмы разрабатывались с использованием общедоступных корпусов First Impressions V2 и MuPTA. Для оценивания показателей качества алгоритмов, сравнимых с современными показателями и представленными в табл. 1, необходимо получить доступ к корпусам по адресу: <https://chalearnlap.cvc.uab.cat/dataset/24/description/> (для First Impressions V2) и <https://hci.nw.ru/ru/pages/mupta-corpus/> (для MuPTA). Корпус MuPTA собирался исследовательской группой СПб ФИЦ РАН (правообладатель), имеет государственную регистрацию в Реестре баз данных №2023623779 от 16.11.2023 г. ([https://new.fips.ru/registers-doc-view/fips\\_servlet?DB=DB&DocNumber=2023624011](https://new.fips.ru/registers-doc-view/fips_servlet?DB=DB&DocNumber=2023624011)), доступен для некоммерческого использования после подписания лицензионного

соглашения. В том числе, предусмотрена загрузка по URL-адресам по 10 тестовых мультимодальных файлов из обоих корпусов, которые расположены по адресу: <https://hci.nw.ru/ru/pages/ocean-ai>. Пример загрузки мультимодальных файлов и их URL-адреса представлены в разделе 3.2.1. Директорию к тестовым мультимодальным файлам необходимо установить через свойство `path_to_dataset_` класса `oceanai.modules.lab.build.Run` и использовать следующий код:

```
_b5 = Run()
# Настройки ядра
_b5.path_to_dataset_ = '/Users/dl/GitHub/oceanai/oceanai/dataset/First Impressions'
# Директория набора данных
format = 'mp4' # для корпуса First Impressions V2 или 'mov' для корпуса MuPTA
corpus = 'fi' # для корпуса First Impressions V2 или 'mupta' для корпуса MuPTA
_b5.ext_ = [format] # Расширения искоемых файлов

# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_[corpus]['sberdisk']
res_get_video_union_predictions = _b5.get_video_union_predictions(
    depth = 2, # Глубина иерархии для получения аудио и видеоданных
    recursive = False, # Рекурсивный поиск данных
    reduction_fps = 5, # Понижение кадровой частоты
    window = 10, # Размер окна сегмента сигнала (в секундах)
    step = 5, # Шаг сдвига окна сегмента сигнала (в секундах)
    accuracy = True, # Вычисление точности
    url_accuracy = url_accuracy,
    logs = True, # При необходимости формировать LOG файл
    out = True, # Отображение
    runtime = True, # Подсчет времени выполнения
    run = True # Блокировка выполнения
)
```

Целевые оценки для корпусов инициализируются аргументом `url_accuracy` методов `get_video_union_predictions()`, `get_audio_union_predictions()`, `get_text_union_predictions()`, `get_av_union_predictions()` и `get_avt_union_predictions()` и расположены по адресу <https://hci.nw.ru/ru/pages/ocean-ai>. Мультимодальная оценка ПКЛЧ в методе `get_avt_union_predictions()` (согласно последовательности действий, представленной в разделе 3.2.1) выполняется в несколько этапов:

1. Вычисление низкоуровневых акустических признаков.
2. Вычисление среднеуровневых акустических признаков.
3. Вычисление низкоуровневых визуальных признаков.
4. Вычисление среднеуровневых визуальных признаков.
5. Вычисление низкоуровневых текстовых признаков.
6. Вычисление среднеуровневых текстовых признаков.
7. Вычисление статистических функционалов (средних значений и стандартных отклонений) для среднеуровневых акустических и визуальных признаков.
8. Вычисление мультимодальных гипотез предсказаний ПКЛЧ.



9. Оценивание качества алгоритмов путем вычисления показателей между целевыми оценками ПКЛЧ и гипотезами предсказаний.

Низкоуровневые признаки – это признаки общего назначения, полученные из необработанной аудио-, видео- и текстовой информации о человеке, используемые для решения разных задач анализа человека. В нашем случае это экспертные и нейросетевые признаки. Среднеуровневые признаки – это специальные признаки для решения конкретной задачи анализа человека. Они строятся на основе низкоуровневых признаков и направлены на моделирование ПКЛЧ. Высокоуровневые признаки – это гипотезы предсказаний ПКЛЧ, созданные на основе среднеуровневых признаков и используемые для их интерпретации при решении практических задач.

## **5. ОБРАЩЕНИЕ К ПРОГРАММЕ**

### **5.1. Точки входа в программу**

Создание экземпляра класса:

- `oceanai.modules.lab.build.Run()`.

Свойства класса:

- `lang (str)` – получение текущего языка интерфейса;
- `color_simple (str)` – получение/установка цвета обычного текста;
- `color_info (str)` – получение/установка цвета текста, содержащего информацию;
- `color_err (str)` – получение/установка цвета текста, содержащего ошибку;
- `color_true (str)` – получение/установка цвета текста, содержащего положительную информацию;
- `bold_text (bool)` – получение/установка жирного начертания текста;
- `num_to_df_display (int)` – получение/установка количества строк для отображения в таблицах;
- `text_runtime (str)` – текст времени выполнения;
- `metadata (bool)` - отображение информации о библиотеке.

Общие аргументы, используемые во всех базовых методах реализованных классов:

- `out (bool)` – отображение;
- `runtime (bool)` – подсчет времени выполнения;
- `run (bool)` – блокировка выполнения;
- `last (bool)` – замена последнего сообщения.

1. Метод извлечения визуальных признаков: `get_visual_features()`.

Аргументы:

- `path (str)` – путь к мультимодальному файлу;
- `reduction_fps (int)` – понижение кадровой частоты;
- `window (int)` – размер окна сегмента сигнала (в кадрах);
- `step (int)` – шаг сдвига окна сегмента сигнала (в кадрах);
- `lang (str)` – выбор языка извлечения признаков.

2. Метод извлечения акустических признаков: `get_acoustic_features()`.

Аргументы:

- `path (str)` – путь к мультимодальному файлу;
- `sr (int)` – частота дискретизации;
- `window (Union[int, float])` – размер окна сегмента сигнала (в секундах);
- `step (Union[int, float])` – шаг сдвига окна сегмента сигнала (в секундах).

3. Метод извлечения текстовых признаков: `get_text_features()`.

Аргументы:

- `path (str)` – путь к мультимодальному файлу;
- `asr (bool)` – автоматическое распознавание речи;
- `lang (str)` – выбор моделей по языку обучающего корпуса;
- `show_text (bool)` – отображение распознанного или считанного из файла текста.

4. Методы формирования нейросетевых моделей:

- `load_audio_model_hc();`
- `load_audio_model_nn();`
- `load_audio_models_b5();`
- `load_video_model_deep_fe();`
- `load_video_model_hc();`
- `load_video_model_nn();`
- `load_text_model_hc();`
- `load_text_model_hc();`
- `load_text_model_nn();`
- `load_text_model_b5();`
- `load_av_models_b5();`
- `load_avt_model_b5();`

Аргументы:

- `show_summary (bool)` – отображение сформированной нейросетевой архитектуры модели.

5. Методы загрузки вспомогательных инструментов и моделей для извлечения текстовых признаков:

- `load_text_features();`
- `setup_translation_model();`
- `setup_bert_encoder().`

Аргументы:

- `force_reload (bool)` – принудительная загрузка файлов.

6. Методы загрузки весов нейросетевых моделей:

- `load_video_model_weights_hc();`
- `load_video_model_weights_nn();`
- `load_video_model_weights_deep_fe();`
- `load_audio_model_weights_hc();`
- `load_audio_model_weights_nn();`
- `load_text_model_weights_hc();`
- `load_text_model_weights_nn().`

Аргументы:

- `url (str)` – полный путь к файлу с весами нейросетевой модели;
- `force_reload (bool)` – принудительная загрузка файла с весами нейросетевой

модели из сети;

- `corpus (str)` – выбор моделей по обучающему корпусу (для только тестовых моделей извлечения признаков);

- `lang (str)` – выбор моделей по языку обучающего корпуса (только для визуальной модели извлечения экспертных признаков).

7. Методы загрузки весов нейросетевых моделей для получения результатов оценки ПКЛЧ:

- `load_video_models_weights_b5();`
- `load_audio_models_weights_b5();`
- `load_text_model_weights_b5();`
- `load_av_models_weights_b5();`
- `load_avt_model_weights_b5().`

Аргументы:

- `url_openness (str)` – полный путь к файлу с весами нейросетевой модели (открытость опыту);
- `url_conscientiousness (str)` – полный путь к файлу с весами нейросетевой модели (добросовестность);
- `url_extraversion (str)` – полный путь к файлу с весами нейросетевой модели (экстраверсия);
- `url_agreeableness (str)` – полный путь к файлу с весами нейросетевой модели (доброжелательность);
- `url_non_neuroticism (str)` – полный путь к файлу с весами нейросетевой модели (эмоциональная стабильность);
- `url (str)` – полный путь к файлу с весами нейросетевой модели (только для методов `load_text_model_weights_b5()` и `load_avt_model_weights_b5()`);
- `force_reload (bool)` – принудительная загрузка файлов с весами нейросетевых моделей из сети.

8. Метод получения объединенных гипотез предсказаний по аудиомодальности: `get_audio_union_predictions()`.

Аргументы:

- `sr (int)` – частота дискретизации;
- `window (Union[int, float])` – размер окна сегмента сигнала (в секундах);
- `step (Union[int, float])` – шаг сдвига окна сегмента сигнала (в секундах);

9. Метод получения объединенных гипотез предсказаний по видеомодальности: `get_video_union_predictions()`.

Аргументы:

- `reduction_fps (int)` – понижение кадровой частоты;
- `window (int)` – размер окна сегмента сигнала (в кадрах);
- `step (int)` – шаг сдвига окна сегмента сигнала (в кадрах).

10. Метод получения объединенных гипотез предсказаний по текстовой модальности: `get_text_union_predictions()`.

Аргументы:

- `asr (bool)` – автоматическое распознавание речи;
- `lang (str)` – выбор моделей по языку обучающего корпуса.

11. Метод получения мультимодальных гипотез предсказаний по видео- и аудиомодальностям: `get_av_union_predictions()`.

Аргументы:

- `sr (int)` – частота дискретизации;
- `window_audio (Union[int, float])` – размер окна сегмента аудиосигнала (в секундах);
- `step_audio (Union[int, float])` – шаг сдвига окна сегмента аудиосигнала (в секундах);
- `reduction_fps (int)` – понижение кадровой частоты;
- `window_video (int)` – размер окна сегмента видеосигнала (в кадрах);
- `step_video (int)` – шаг сдвига окна сегмента видеосигнала (в кадрах).

12. Метод получения мультимодальных гипотез предсказаний по видео-, аудио- и текстовой модальностям: `get_avt_predictions()`.

Аргументы:

- `sr (int)` – частота дискретизации;
- `window_audio (Union[int, float])` – размер окна сегмента аудиосигнала (в секундах);
- `step_audio (Union[int, float])` – шаг сдвига окна сегмента аудиосигнала (в секундах);
- `reduction_fps (int)` – понижение кадровой частоты;
- `window_video (int)` – размер окна сегмента видеосигнала (в кадрах);
- `step_video (int)` – шаг сдвига окна сегмента видеосигнала (в кадрах);
- `asr (bool)` – автоматическое распознавание речи;
- `lang (str)` – выбор моделей по языку обучающего корпуса.

Общие аргументы методов 8-12:

- `accuracy (bool)` – вычисление точности;
- `url_accuracy (str)` – полный путь к файлу с верными предсказаниями для подсчета точности.
- `logs (bool)` – при необходимости формировать LOG файл.
- `depth (int)` – глубина иерархии для получения данных;
- `recursive (bool)` – рекурсивный поиск данных.

## 5.2. Базовые функции программы

Компонент OCEAN-AI разрабатывался с использованием двух корпусов First Impressions V2 [1] и MuPTA [2], поэтому предоставляются веса моделей, обученные на двух

корпусах. Для загрузки нужных весов моделей, обученных на корпусе First Impressions V2, необходимо использовать комбинацию обозначений **'fi'** (сокращение от названия корпуса First Impressions V2) и **'en'** в соответствующих аргументах в методах как показано ниже, для MuPTA – комбинацию **'mupta'** и **'ru'**. Для англоязычных мультимодальных данных допускается использование комбинации **'fi'** и **'en'**. Для русскоязычных мультимодальных данных – обе комбинации.

Обозначения **'fi'** и **'mupta'** задаются свойством `weights_for_big5_`, а также в методах `load_text_model_hc()`, `load_text_model_nn()` аргументом `corpus`. Обозначения **'ru'** и **'en'** задаются внутри методов `get_visual_features()`, `load_video_model_hc()`, `get_video_union_predictions()`, `get_text_features()`, `get_text_union_predictions()`, `get_av_union_predictions()`, `get_avt_union_predictions()` аргументом `lang`, а также используются при выборе языка интерфейса при создании экземпляра класса `oceanai.modules.lab.build.Run`. Ниже приведены примеры для загрузки весов моделей, обученных на корпусе First Impressions V2.

Создание экземпляра класса для сборки компонента OCEAN-AI:

```
from oceanai.modules.lab.build import Run

_b5 = Run(
    lang = 'ru',          # Язык интерфейса
    color_simple = #FFF,  # Цвет обычного текста (шестнадцатеричный код)
    color_info = #1776D2, # Цвет текста, содержащего информацию (шестнадцатеричный код)
    color_err = #FF0000,  # Цвет текста, содержащего ошибку (шестнадцатеричный код)
    color_true = #008001, # Цвет текста, содержащего положительную информацию
    # (шестнадцатеричный код)
    bold_text = True,     # Жирное начертание текста
    num_to_df_display = 30, # Количество строк для отображения в таблицах
    text_runtime = Время выполнения, # Текст времени выполнения
    bold_text = True,     # Жирное начертание текста
    metadata = True,      # Отображение информации о библиотеке
)
```

5.2.1. Предобработка аудиоинформации, вычисление акустических признаков и выдача по ним гипотез предсказаний оценок ПКЛЧ.

Предобработка аудиоинформации и вычисление акустических экспертных признаков и log-mel спектрограмм:

```
# Настройки ядра
sr = 44100 # Частота дискретизации
# Путь к мультимодальному файлу
path = '/Users/dl/GitHub/oceanai/oceanai/dataset/test80_01/glgfB3vFewc.004.mp4'
hc_features, melspectrogram_features = _b5.get_acoustic_features(
    path = path,          # Путь к мультимодальному файлу
    sr = sr,              # Частота дискретизации
    window = 2,           # Размер окна сегмента сигнала (в секундах)
    step = 1,             # Шаг сдвига окна сегмента сигнала (в секундах)
    out = True,           # Отображение
    runtime = True,       # Подсчет времени выполнения
    run = True            # Блокировка выполнения
```

)

Для выдачи среднеуровневых признаков и гипотез предсказаний оценок ПКЛЧ по акустическим признакам, необходимо:

1. Сформировать нейросетевые архитектуры моделей:

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по экспертным признакам:

```
res_load_model_hc = _b5.load_audio_model_hc(  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,           # Отображение  
    runtime = True,       # Подсчет времени выполнения  
    run = True            # Блокировка выполнения  
)
```

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ из log-mel спектрограмм:

```
res_load_model_nn = _b5.load_audio_model_nn(  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,           # Отображение  
    runtime = True,       # Подсчет времени выполнения  
    run = True            # Блокировка выполнения  
)
```

- для вычисления объединенных гипотез предсказаний оценок ПКЛЧ по

аудиоинформации и результатов оценивания методов:

```
res_load_models_b5 = _b5.load_audio_models_b5(  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,           # Отображение  
    runtime = True,       # Подсчет времени выполнения  
    run = True            # Блокировка выполнения  
)
```

2. Загрузить веса нейросетевых моделей:

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по экспертным признакам:

```
# Настройки ядра  
_b5.path_to_save_ = './models' # Директория для сохранения файла  
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг  
url = _b5.weights_for_big5['audio']['fi']['hc']['sberdisk']  
res_load_model_weights_hc = _b5.load_audio_model_weights_hc(  
    url = url, # Полный путь к файлу с весами нейросетевой модели  
    force_reload = True, # Принудительная загрузка файла с весами нейросетевой модели из сети  
    out = True, # Отображение  
    runtime = True, # Подсчет времени выполнения  
    run = True # Блокировка выполнения  
)
```

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ из log-mel спектрограмм:

```
# Настройки ядра  
_b5.path_to_save_ = './models' # Директория для сохранения файла  
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг  
url = _b5.weights_for_big5['audio']['fi']['nn']['sberdisk']  
res_load_model_weights_nn = _b5.load_audio_model_weights_nn(  
    url = url, # Полный путь к файлу с весами нейросетевой модели
```

```

force_reload = False, # Принудительная загрузка файла с весами нейросетевой модели из сети
out = True,          # Отображение
runtime = True,      # Подсчет времени выполнения
run = True           # Блокировка выполнения
)

```

• для вычисления объединенных гипотез предсказаний оценок ПКЛЧ по аудиоинформации и результатов оценивания методов:

```

# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url_OPE = _b5.weights_for_big5_['audio']['fi']['b5']['openness']['sberdisk']
url_CON = _b5.weights_for_big5_['audio']['fi']['b5']['conscientiousness']['sberdisk']
url_EXT = _b5.weights_for_big5_['audio']['fi']['b5']['extraversion']['sberdisk']
url_AGR = _b5.weights_for_big5_['audio']['fi']['b5']['agreeableness']['sberdisk']
url_NNEU = _b5.weights_for_big5_['audio']['fi']['b5']['non_neuroticism']['sberdisk']
res_load_models_weights_b5 = _b5.load_audio_models_weights_b5(
    url_openness = url_OPE,          # Открытость опыту
    url_conscientiousness = url_CON, # Добросовестность
    url_extraversion = url_EXT,      # Экстраверсия
    url_agreeableness = url_AGR,     # Доброжелательность
    url_non_neuroticism = url_NNEU,  # Эмоциональная стабильность
    force_reload = False, # Принудительная загрузка файла с весами нейросетевой модели из сети
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)

```

3. Вызвать функцию (метод) для получения объединенных гипотез предсказаний оценок ПКЛЧ по аудиоинформации и результатов оценивания методов:

```

# Настройки ядра
_b5.path_to_dataset_ = '/Users/dl/GitHub/oceanai/oceanai/dataset/First_Impressions'
# Директория набора данных
# Директории не входящие в выборку
_b5.ignore_dirs_ = []
# Названия ключей для DataFrame набора данных
_b5.keys_dataset_ = ['Path', 'Openness', 'Conscientiousness', 'Extraversion', 'Agreeableness',
'Non-Neuroticism']
_b5.ext_ = ['.mp4'] # Расширения искомых файлов
_b5.path_to_logs_ = './logs' # Директория для сохранения LOG файлов
# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk'] # Ключ 'fi' или 'mupta'
res_get_audio_union_predictions = _b5.get_audio_union_predictions(
    depth = 2,          # Глубина иерархии для получения аудио и видеоданных
    recursive = False, # Рекурсивный поиск данных
    sr = 44100,         # Частота дискретизации
    window = 2,        # Размер окна сегмента сигнала (в секундах)
    step = 1,          # Шаг сдвига окна сегмента сигнала (в секундах)
    accuracy = True,   # Вычисление точности
    url_accuracy = url_accuracy,
    logs = True,      # При необходимости формировать LOG файл
    out = True,       # Отображение
    runtime = True,   # Подсчет времени выполнения
    run = True       # Блокировка выполнения
)

```

5.2.2. Предобработка видеоинформации, вычисление визуальных признаков и выдача по ним гипотез предсказаний оценок ПКЛЧ.



Для выполнения предварительной обработки видеoinформации и вычисления визуальных признаков необходимо:

Для выполнения предварительной обработки видеoinформации и вычисления визуальных признаков необходимо:

1. Сформировать нейросетевую архитектуру модели для извлечения нейросетевых признаков по видеомодальности:

```
res_load_model_deep_fe = _b5.load_video_model_deep_fe(  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,           # Отображение  
    runtime = True,       # Подсчет времени выполнения  
    run = True            # Блокировка выполнения  
)
```

2. Загрузить веса нейросетевой модели для извлечения нейросетевых признаков по видеомодальности:

```
# Настройки ядра  
_b5.path_to_save_ = './models' # Директория для сохранения файла  
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг  
url = _b5.weights_for_big5_['video']['fi']['fe']['sberdisk']  
res_load_model_weights_deep_fe = _b5.load_video_model_weights_deep_fe(  
    url = url,                # Полный путь к файлу с весами нейросетевой модели  
    force_reload = True,     # Принудительная загрузка файла с весами нейросетевой модели из сети  
    out = True,              # Отображение  
    runtime = True,         # Подсчет времени выполнения  
    run = True              # Блокировка выполнения  
)
```

3. Вызвать функцию (метод) для предобработки видеoinформации и вычисления визуальных экспертных и нейросетевых признаков:

```
# Путь к видеофайлу  
path = '/Users/dl/GitHub/oceanai/oceanai/dataset/test80_01/glgfB3vFewc.004.mp4'  
hc_features, nn_features = _b5.get_visual_features(  
    path = path,             # Путь к мультимодальному файлу  
    reduction_fps = 5,       # Понижение кадровой частоты  
    window = 10,            # Размер окна сегмента сигнала (в кадрах)  
    step = 5,               # Шаг сдвига окна сегмента сигнала (в кадрах)  
    lang = 'en',            # Выбор языка для моделей First Impressions V2 'en' для MuPTA 'ru'  
    out = True,              # Отображение  
    runtime = True,         # Подсчет времени выполнения  
    run = True              # Блокировка выполнения  
)
```

Для выдачи среднеуровневых признаков и гипотез предсказаний оценок ПКЛЧ по визуальным признакам необходимо:

1. Сформировать нейросетевые архитектуры моделей:

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по экспертным признакам:

```
res_load_model_hc = _b5.load_video_model_hc(  
    lang = 'en',            # Выбор языка для моделей First Impressions V2 'en' для MuPTA 'ru'  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,           # Отображение  
    runtime = True,       # Подсчет времени выполнения  
    run = True            # Блокировка выполнения  
)
```

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по нейросетевым признакам:

```
res_load_model_nn = _b5.load_video_model_nn(
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)
```

- для вычисления объединенных гипотез предсказаний оценок ПКЛЧ по

видеоинформации и результатов оценивания методов:

```
res_load_models_b5 = _b5.load_video_models_b5(
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)
```

## 2. Загрузить веса нейросетевых моделей:

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по экспертным признакам:

```
# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url = _b5.weights_for_big5['video']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_video_model_weights_hc(
    url = url, # Полный путь к файлу с весами нейросетевой модели
    force_reload = True, # Принудительная загрузка файла с весами нейросетевой модели из сети
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)
```

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по нейросетевым признакам:

```
# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url = _b5.weights_for_big5['video']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_video_model_weights_nn(
    url = url, # Полный путь к файлу с весами нейросетевой модели
    force_reload = True, # Принудительная загрузка файла с весами нейросетевой модели из сети
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)
```

- для вычисления объединенных гипотез предсказаний оценок ПКЛЧ по

видеоинформации и результатов оценивания методов:

```
# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url_OPE = _b5.weights_for_big5['video']['fi']['b5']['openness']['sberdisk']
url_CON = _b5.weights_for_big5['video']['fi']['b5']['conscientiousness']['sberdisk']
url_EXT = _b5.weights_for_big5['video']['fi']['b5']['extraversion']['sberdisk']
url_AGR = _b5.weights_for_big5['video']['fi']['b5']['agreeableness']['sberdisk']
url_NNEU = _b5.weights_for_big5['video']['fi']['b5']['non_neuroticism']['sberdisk']
res_load_models_weights_b5 = _b5.load_video_models_weights_b5(
```

```

url_openness = url_OPE,          # Открытость опыту
url_conscientiousness = url_CON, # Добросовестность
url_extraversion = url_EXT,      # Экстраверсия
url_agreeableness = url_AGR,     # Доброжелательность
url_non_neuroticism = url_NNEU,  # Эмоциональная стабильность
force_reload = True,             # Принудительная загрузка файла с весами нейросетевой модели из сети
out = True,                      # Отображение
runtime = True,                  # Подсчет времени выполнения
run = True # Блокировка выполнения
)

```

3. Вызвать функцию (метод) для получения объединенных гипотез предсказаний оценок ПКЛЧ по видеоинформации и результатов оценивания методов:

```

# Настройки ядра
_b5.path_to_dataset_ = '/Users/dl/GitHub/oceanai/oceanai/dataset/First Impressions'
# Директория набора данных
# Директории не входящие в выборку
_b5.ignore_dirs_ = []
# Названия ключей для DataFrame набора данных
_b5.keys_dataset_ = ['Path', 'Openness', 'Conscientiousness', 'Extraversion', 'Agreeableness',
'Non-Neuroticism']
_b5.ext_ = ['.mp4'] # Расширения искоемых файлов
_b5.path_to_logs_ = './logs' # Директория для сохранения LOG файлов
# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk']
res_get_video_union_predictions = _b5.get_video_union_predictions(
    depth = 1,          # Глубина иерархии для получения аудио и видеоданных
    recursive = False,  # Рекурсивный поиск данных
    reduction_fps = 5,  # Понижение кадровой частоты
    window = 10,        # Размер окна сегмента сигнала (в секундах)
    step = 5,           # Шаг сдвига окна сегмента сигнала (в секундах)
    lang = 'en',        # Выбор языка для моделей First Impressions V2 'en' для MuPTA 'ru'
    accuracy = True,     # Вычисление точности
    url_accuracy = url_accuracy,
    logs = True,         # При необходимости формировать LOG файл
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)

```

5.2.3. Предобработка текстовой информации, вычисление текстовых признаков и выдача по ним гипотез предсказаний оценок ПКЛЧ.

Для выполнения предварительной обработки текстовой информации и вычисления визуальных признаков необходимо:

1. Загрузить словарь с экспертными признаками:

```

res_load_text_features = _b5.load_text_features(
    force_reload = True, # Принудительная загрузка файла
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)

```

2. Сформировать токенизатор и нейросетевую модель машинного перевода (RU -> EN):

```

res_setup_translation_model = _b5.setup_translation_model(
    out = True,          # Отображение
    runtime = True,      # Подсчет времени выполнения
    run = True           # Блокировка выполнения
)

```

)

3. Сформировать токенизатор и нейросетевую модель BERT (для кодирования слов):

```
res_setup_translation_model = _b5.setup_bert_encoder(  
    force_reload = True, # Принудительная загрузка файла  
    out = True,         # Отображение  
    runtime = True,     # Подсчет времени выполнения  
    run = True          # Блокировка выполнения  
)
```

4. Вызвать функцию (метод) для предобработки текстовой информации и вычисления текстовых экспертных и нейросетевых признаков:

```
# Путь к видеофайлу  
path = '/Users/dl/GitHub/oceanai/oceanai/dataset/test80_01/glgfB3vFewc.004.mp4'  
hc_features, nn_features = _b5.get_text_features(  
    path = path,         # Путь к мультимодальному файлу  
    asr = False,        # Распознавание речи  
    lang = 'en',        # Выбор языка корпуса 'en' или 'ru'  
    out = True,         # Отображение  
    runtime = True,     # Подсчет времени выполнения  
    run = True          # Блокировка выполнения  
)
```

Для выдачи среднеуровневых признаков и гипотез предсказаний оценок ПКЛЧ по текстовым признакам необходимо:

4. Сформировать нейросетевые архитектуры моделей:

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по экспертным признакам:

```
res_load_text_model_hc-fi = _b5.load_text_model_hc(  
    corpus = 'fi',      # Выбор весов моделей 'fi' или 'mupta'  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,         # Отображение  
    runtime = True,     # Подсчет времени выполнения  
    run = True          # Блокировка выполнения  
)
```

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по нейросетевым признакам:

```
res_load_model_nn = _b5.load_text_model_nn(  
    corpus = 'fi',      # Выбор весов моделей 'fi' или 'mupta'  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,         # Отображение  
    runtime = True,     # Подсчет времени выполнения  
    run = True          # Блокировка выполнения  
)
```

- для вычисления объединенных гипотез предсказаний оценок ПКЛЧ по

текстовой информации и результатов оценивания методов:

```
res_load_models_b5 = _b5.load_text_model_b5(  
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели  
    out = True,         # Отображение  
    runtime = True,     # Подсчет времени выполнения  
    run = True          # Блокировка выполнения  
)
```

5. Загрузить веса нейросетевых моделей:

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по экспертным признакам:

```
# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url = _b5.weights_for_big5_['text']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_text_model_weights_hc(
    url = url, # Полный путь к файлу с весами нейросетевой модели
    force_reload = True, # Принудительная загрузка файла с весами нейросетевой модели из сети
    out = True, # Отображение
    runtime = True, # Подсчет времени выполнения
    run = True # Блокировка выполнения
)
```

- для вычисления среднеуровневых признаков и гипотез предсказаний оценок

ПКЛЧ по нейросетевым признакам:

```
# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url = _b5.weights_for_big5_['text']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_text_model_weights_nn(
    url = url, # Полный путь к файлу с весами нейросетевой модели
    force_reload = True, # Принудительная загрузка файла с весами нейросетевой модели из сети
    out = True, # Отображение
    runtime = True, # Подсчет времени выполнения
    run = True # Блокировка выполнения
)
```

- для вычисления объединенных гипотез предсказаний оценок ПКЛЧ по текстовой информации и результатов оценивания методов:

```
# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url = _b5.weights_for_big5_['text']['fi']['b5']['sberdisk']
res_load_text_model_weights_b5 = _b5.load_text_model_weights_b5(
    url = url, # Полный путь к файлу с весами нейросетевой модели
    force_reload = True, # Принудительная загрузка файла с весами нейросетевой модели из сети
    out = True, # Отображение
    runtime = True, # Подсчет времени выполнения
    run = True # Блокировка выполнения
)
```

6. Вызвать функцию (метод) для получения объединенных гипотез предсказаний оценок ПКЛЧ по текстовой информации и результатов оценивания методов:

```
_b5.path_to_dataset_ = '/Users/dl/GitHub/oceanai/oceanai/dataset/First Impressions'
# Директория набора данных
# Названия ключей для DataFrame набора данных
_b5.keys_dataset_ = ['Path', 'Openness', 'Conscientiousness', 'Extraversion', 'Agreeableness',
'Non-Neuroticism']
_b5.ext_ = ['.mp4'] # Расширения искоемых файлов
_b5.path_to_logs_ = './logs' # Директория для сохранения LOG файлов
# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk']
res_get_text_union_predictions = _b5.get_text_union_predictions(
    depth = 1, # Глубина иерархии для получения аудио и видеоданных
    recursive = False, # Рекурсивный поиск данных
    asr = False, # Распознавание речи
    lang = 'en', # Выбор языка для моделей First Impressions V2 'en' для MuPTA 'ru'
    accuracy = True, # Вычисление точности
    url_accuracy = url_accuracy,
```

```

logs = True,      # При необходимости формировать LOG файл
out = True,       # Отображение
runtime = True,   # Подсчет времени выполнения
run = True        # Блокировка выполнения
)

```

5.2.4. Выполнение мультимодального объединения среднеуровневых признаков, полученных от аудио-, видео- и текстовой информации, и получение мультимодальных гипотез предсказаний оценок ПКЛЧ.

Для выполнения мультимодального объединения среднеуровневых признаков и получения гипотез предсказаний оценок ПКЛЧ необходимо:

1. Сформировать нейросетевую архитектуру модели для выдачи мультимодальных гипотез предсказаний оценок ПКЛЧ и результатов оценивания методов:

```

res_load_avt_model_b5 = _b5.load_avt_model_b5(
    show_summary = False, # Отображение сформированной нейросетевой архитектуры модели
    out = True, # Отображение
    runtime = True, # Подсчет времени выполнения
    run = True # Блокировка выполнения
)

```

2. Загрузить веса нейросетевой модели для получения мультимодальных гипотез предсказаний оценок ПКЛЧ и результатов оценивания методов:

```

# Настройки ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг
url = _b5.weights_for_big5_['avt']['fi']['b5']['openness']['sberdisk']
res_load_avt_model_weights_b5 = _b5.load_avt_model_weights_b5(
    url = url, # Полный путь к файлу с весами нейросетевой модели
    force_reload = False, # Принудительная загрузка файла с весами нейросетевой модели из сети
    out = True, # Отображение
    runtime = True, # Подсчет времени выполнения
    run = True # Блокировка выполнения
)

```

3. Вызвать функцию (метод) для получения мультимодальных гипотез предсказаний оценок ПКЛЧ и результатов оценивания методов:

```

_b5.path_to_dataset_ = '/Users/dl/GitHub/OCEANAI.private/oceanai/dataset//First Impressions' #
Директория набора данных
# Названия ключей для DataFrame набора данных
_b5.keys_dataset_ = ['Path', 'Openness', 'Conscientiousness', 'Extraversion', 'Agreeableness',
'Non-Neuroticism']
_b5.ext_ = ['.mp4'] # Расширения искоемых файлов
_b5.path_to_logs_ = './logs' # Директория для сохранения LOG файлов
# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk']
_b5.get_av_union_predictions(
    depth = 2, # Глубина иерархии для получения аудио и видеоданных
    recursive = False, # Рекурсивный поиск данных
    sr = 44100, # Частота дискретизации
    window_audio = 2, # Размер окна сегмента аудио сигнала (в секундах)
    step_audio = 1, # Шаг сдвига окна сегмента аудио сигнала (в секундах)
    reduction_fps = 5, # Понижение кадровой частоты
    window_video = 10, # Размер окна сегмента видео сигнала (в секундах)
    step_video = 5, # Шаг сдвига окна сегмента видео сигнала (в секундах)
    asr = False, # Распознавание речи
    lang = 'en', # Выбор языка для моделей First Impressions V2 'en' для MuPTA 'ru'
    accuracy = True, # Вычисление точности
    url_accuracy = url_accuracy,
)

```



```
logs = True,      # При необходимости формировать LOG файл
out = True,       # Отображение
runtime = True,   # Подсчет времени выполнения
run = True        # Блокировка выполнения
)
```

### 5.3. Пайплайны компонента на основе его функций

Структура (пайплайн) компонента OCEAN-AI для оценивания ПКЛЧ по мультимодальным данным представлена на рисунке 14.

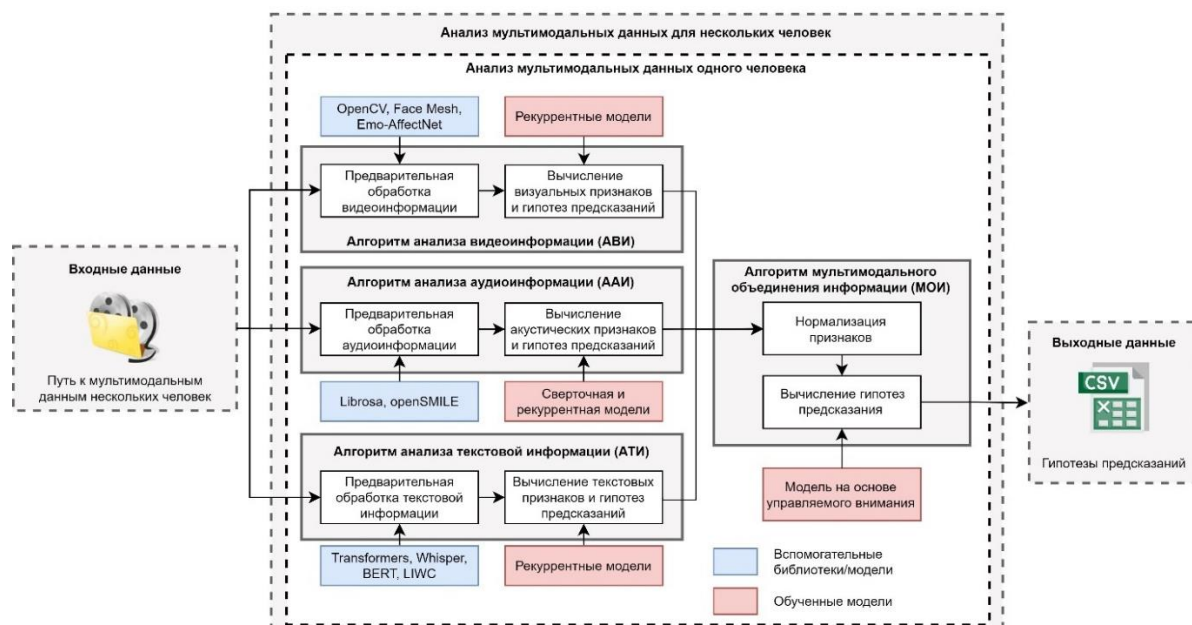


Рисунок 14 – Структура (пайплайн) компонента OCEAN-AI для оценивания ПКЛЧ по мультимодальным данным

Согласно рис. 14 на вход компонента OCEAN-AI поступает множество мультимодальных данных нескольких человек. Компонент обрабатывает мультимодальные данные последовательно с помощью алгоритмов ААИ, АВИ, АТИ и МОИ. Ниже подробно рассмотрена работа каждого алгоритма компонента OCEAN-AI.

#### 5.3.1. Структура (пайплайн) для алгоритма анализа аудиоинформации ААИ

Структура (пайплайн) компонента OCEAN-AI для алгоритма ААИ для оценивания ПКЛЧ по голосовым характеристикам представлена на рис. 15.



Рисунок 15 – Структура (пайплайн) компонента OCEAN-AI для алгоритма анализа аудиоинформации ААИ для оценивания ПКЛЧ по голосовым характеристикам

Для получения оценок ПКЛЧ по аудиоинформации согласно рис. 15 для 10 тестовых мультимодальных файлов из корпуса First Impressions V2 [1] необходимо выполнить последовательность действий:

```
# Импорт необходимых инструментов
import os
import pandas as pd

# Импорт модуля
from oceanai.modules.lab.build import Run

# Создание экземпляра класса
_b5 = Run()

# Настройка ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг

# Формирование аудиомоделей
res_load_model_hc = _b5.load_audio_model_hc()
res_load_model_nn = _b5.load_audio_model_nn()

# Загрузка весов аудиомоделей
url = _b5.weights_for_big5['audio']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_audio_model_weights_hc(url = url)

url = _b5.weights_for_big5['audio']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_audio_model_weights_nn(url = url)

# Формирование модели для анализа аудиоинформации
res_load_model_b5 = _b5.load_audio_models_b5()

# Загрузка весов модели для анализа аудиоинформации
url_OPE = _b5.weights_for_big5['audio']['fi']['b5']['openness']['sberdisk']
url_CON = _b5.weights_for_big5['audio']['fi']['b5']['conscientiousness']['sberdisk']
url_EXT = _b5.weights_for_big5['audio']['fi']['b5']['extraversion']['sberdisk']
url_AGR = _b5.weights_for_big5['audio']['fi']['b5']['agreeableness']['sberdisk']
url_NNEU = _b5.weights_for_big5['audio']['fi']['b5']['non_neuroticism']['sberdisk']

res_load_models_weights_b5 = _b5.load_audio_models_weights_b5(
    url_openness = url_OPE, # Открытость опыту
    url_conscientiousness = url_CON, # Добросовестность
    url_extraversion = url_EXT, # Экстраверсия
    url_agreeableness = url_AGR, # Доброжелательность
    url_non_neuroticism = url_NNEU, # Эмоциональная стабильность
)

PATH_TO_DIR = './video_FI/'
```



```

PATH_SAVE_VIDEO = './video_FI/test/'

_b5.path_to_save_ = PATH_SAVE_VIDEO

# Загрузка 10 тестовых мультимодальных файлов из корня First Impressions V2
# URL: https://chalearnlap.cvc.uab.cat/dataset/24/description/
domain = 'https://download.sberdisk.ru/download/file/'
tets_name_files = [
    '429713680?token=FqHdMLSSh7zYSZt&filename=_plk5k7PBEg.003.mp4',
    '429713681?token=Hz9b4lQkrLfic33&filename=be0DQawtVKE.002.mp4',
    '429713683?token=EgUXS9Xs8xHm5gz&filename=2d6btbaNdfo.000.mp4',
    '429713684?token=1U26753kmPYdIgt&filename=300gK3CnzW0.003.mp4',
    '429713685?token=LyigAWLTzDNwKJO&filename=300gK3CnzW0.001.mp4',
    '429713686?token=EpFRbCKHyuc4HPu&filename=cLaZxEf1nE4.004.mp4',
    '429713687?token=FNTkwqBr4jOS95l&filename=g24JGYuT74A.004.mp4',
    '429713688?token=qDT95nz7hfm2Nki&filename=JZNMxa3OKHY.000.mp4',
    '429713689?token=noLguEGXDpbckhg&filename=nvlqJbHk_Lc.003.mp4',
    '429713679?token=9L7RQ0hgdJlcek6&filename=4vdJGgZpj4k.003.mp4'
]

for curr_files in tets_name_files:
    _b5.download_file_from_url(url = domain + curr_files, out = False)

# Получение прогнозов
_b5.path_to_dataset_ = PATH_TO_DIR # Директория набора данных
_b5.ext_ = ['.mp4'] # Расширения искоемых файлов

# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk']

_b5.get_audio_union_predictions(url_accuracy = url_accuracy)

```

Результаты оценивания ПКЛЧ по аудиоинформации представлены на рис. 16 и 17.

	Path	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism
Person ID						
1	2d6btbaNdfo.000.mp4	0.545981	0.655463	0.412282	0.591633	0.566958
2	300gK3CnzW0.001.mp4	0.482044	0.461589	0.372908	0.528315	0.459843
3	300gK3CnzW0.003.mp4	0.447993	0.474451	0.337972	0.497484	0.413
4	4vdJGgZpj4k.003.mp4	0.59249	0.617516	0.539873	0.611519	0.582145
5	be0DQawtVKE.002.mp4	0.579003	0.567649	0.496231	0.608233	0.571018
6	cLaZxEf1nE4.004.mp4	0.653647	0.563365	0.565574	0.582277	0.590217
7	g24JGYuT74A.004.mp4	0.53703	0.329784	0.437795	0.512874	0.471008
8	JZNMxa3OKHY.000.mp4	0.634548	0.567335	0.571362	0.587481	0.584822
9	nvlqJbHk_Lc.003.mp4	0.553215	0.503285	0.428273	0.49168	0.475411
10	_plk5k7PBEg.003.mp4	0.555685	0.555988	0.463666	0.601584	0.54942

Рисунок 16 – Предсказанные оценки ПКЛЧ по аудиоинформации

Рис. 16 показывает следующую информацию: Person ID – порядковый номер человека; Path – название мультимодального файла; Openness («Открытость опыту»), Conscientiousness («Добросовестность»), Extraversion («Экстраверсия»), Agreeableness («Доброжелательность») и Non-Neuroticism («Эмоциональная стабильность») – пять ПКЛЧ,

оцениваемые компонентом OCEAN-AI, и соответствующие им гипотезы предсказаний для каждой человека.

	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism	Mean
<b>Metrics</b>						
<b>MAE</b>	0.0522	0.0595	0.0825	0.0722	0.0554	0.0644
<b>Accuracy</b>	0.9478	0.9405	0.9175	0.9278	0.9446	0.9356

Рисунок 17 – Показатели оценивания алгоритма анализа аудиоинформации ААИ для 10 тестовых мультимодальных файлов

Рис. 17 показывает следующую информацию: Metrics – показатели оценивания алгоритма ААИ; MAE – средняя абсолютная ошибка; Accuracy – 1-MAE. Показатели оценивания вычисляются независимо для каждого из пяти ПКЛЧ. Mean – среднее значение по всем показателям эффективности.

### 5.3.2. Структура (пайплайн) для алгоритма анализа видеоинформации АВИ

Структура (пайплайн) компонента OCEAN-AI для алгоритма АВИ для оценивания ПКЛЧ по лицевым характеристикам представлена на рис. 18.

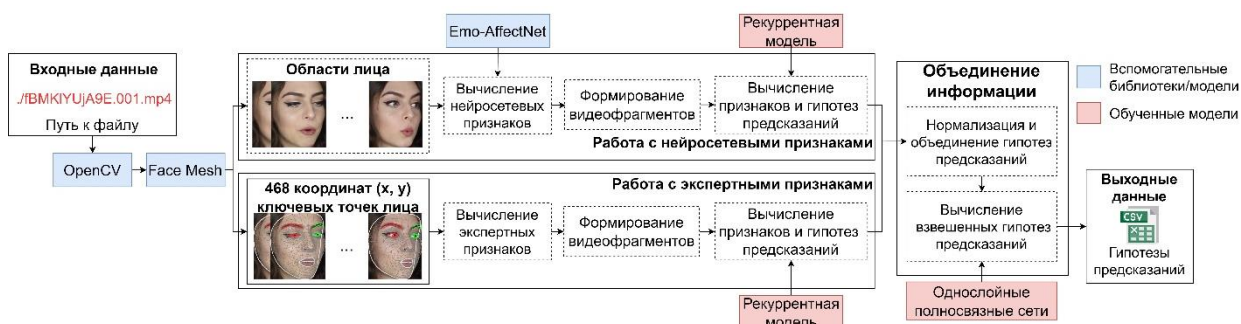


Рисунок 18 – Структура (пайплайн) компонента OCEAN-AI для алгоритма анализа видеоинформации АВИ для оценивания ПКЛЧ по лицевым характеристикам

Для получения оценок ПКЛЧ по видеоинформации согласно рис. 18 для 10 тестовых мультимодальных файлов из корпуса First Impressions V2 необходимо выполнить последовательность действий:

```
# Импорт необходимых инструментов
import os
import pandas as pd

# Импорт модуля
from oceanai.modules.lab.build import Run

# Создание экземпляра класса
_b5 = Run()

# Настройка ядра
```

```
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000      # Размер загрузки файла из сети за 1 шаг

# Формирование видеомоделей
res_load_model_hc = _b5.load_video_model_hc(lang='en')
res_load_model_deep_fe = _b5.load_video_model_deep_fe()
res_load_model_nn = _b5.load_video_model_nn()

# Загрузка весов видеомоделей
url = _b5.weights_for_big5_['video']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_video_model_weights_hc(url = url)

url = _b5.weights_for_big5_['video']['fi']['fe']['sberdisk']
res_load_model_weights_deep_fe = _b5.load_video_model_weights_deep_fe(url = url)

url = _b5.weights_for_big5_['video']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_video_model_weights_nn(url = url)

# Формирование модели для анализа видеoinформации
res_load_model_b5 = _b5.load_video_models_b5()

# Загрузка весов модели для анализа видеoinформации
url_OPE = _b5.weights_for_big5_['video']['fi']['b5']['openness']['sberdisk']
url_CON = _b5.weights_for_big5_['video']['fi']['b5']['conscientiousness']['sberdisk']
url_EXT = _b5.weights_for_big5_['video']['fi']['b5']['extraversion']['sberdisk']
url_AGR = _b5.weights_for_big5_['video']['fi']['b5']['agreeableness']['sberdisk']
url_NNEU = _b5.weights_for_big5_['video']['fi']['b5']['non_neuroticism']['sberdisk']

res_load_models_weights_b5 = _b5.load_video_models_weights_b5(
    url_openness = url_OPE,      # Открытость опыту
    url_conscientiousness = url_CON, # Добросовестность
    url_extraversion = url_EXT,   # Экстраверсия
    url_agreeableness = url_AGR,  # Доброжелательность
    url_non_neuroticism = url_NNEU, # Эмоциональная стабильность
)

PATH_TO_DIR = './video_FI/'
PATH_SAVE_VIDEO = './video_FI/test/'

_b5.path_to_save_ = PATH_SAVE_VIDEO

# Загрузка 10 тестовых мультимодальных файлов из корпуса First Impressions V2
# URL: https://chalearnlap.cvc.uab.cat/dataset/24/description/
domain = 'https://download.sberdisk.ru/download/file/'
tets_name_files = [
    '429713680?token=FqHdMLSSh7zYSZt&filename=_plk5k7PBEg.003.mp4',
    '429713681?token=Hz9b4lQkrLfic33&filename=be0DQawtVKE.002.mp4',
    '429713683?token=EgUXS9Xs8xHm5gz&filename=2d6btbaNdfo.000.mp4',
    '429713684?token=1U26753kmPYdIgt&filename=300gK3CnzW0.003.mp4',
    '429713685?token=LyigAWLTzDNwKJO&filename=300gK3CnzW0.001.mp4',
    '429713686?token=EpFRbCKHyuc4HPu&filename=cLaZxEf1nE4.004.mp4',
    '429713687?token=FNTkwqBr4jOS95l&filename=g24JGYuT74A.004.mp4',
    '429713688?token=qDT95nz7hfm2Nki&filename=JZNMxa3OKHY.000.mp4',
    '429713689?token=noLguEGXDpbckHg&filename=nvlqJbHk_Lc.003.mp4',
    '429713679?token=9L7RQ0hgdlJlcek6&filename=4vdJGgZpj4k.003.mp4'
]

for curr_files in tets_name_files:
    _b5.download_file_from_url(url = domain + curr_files, out = False)

# Получение прогнозов
_b5.path_to_dataset_ = PATH_TO_DIR # Директория набора данных
_b5.ext_ = ['.mp4']                # Расширения искоемых файлов

# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk']
```

`_b5.get_video_union_predictions(url_accuracy = url_accuracy, lang='en')`

Результаты оценивания ПКЛЧ по видеоинформации представлены на рис. 19 и 20.

	Path	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism
<b>Person ID</b>						
1	2d6btbaNdfo.000.mp4	0.459653	0.601691	0.345645	0.555024	0.464603
2	300gK3CnzW0.001.mp4	0.491161	0.461013	0.402908	0.53071	0.477383
3	300gK3CnzW0.003.mp4	0.505455	0.45183	0.39026	0.52886	0.46101
4	4vdJGgZpj4k.003.mp4	0.607575	0.588699	0.500772	0.583191	0.577082
5	be0DQawtVkE.002.mp4	0.660873	0.536593	0.592555	0.668345	0.640274
6	cLaZxEf1nE4.004.mp4	0.663511	0.579378	0.594304	0.589859	0.600188
7	g24JGYuT74A.004.mp4	0.580344	0.396475	0.412041	0.510385	0.445533
8	JZNMxa3OKHY.000.mp4	0.639624	0.565898	0.598395	0.581595	0.597473
9	nvlqJbHk_Lc.003.mp4	0.528223	0.460129	0.418835	0.518816	0.429008
10	_plk5k7PBEg.003.mp4	0.604554	0.657346	0.517441	0.654853	0.593637

Рисунок 19 – Предсказанные оценки ПКЛЧ по видеоинформации

	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism	Mean
<b>Metrics</b>						
<b>MAE</b>	0.0755	0.0567	0.1003	0.0797	0.0741	0.0773
<b>Accuracy</b>	0.9245	0.9433	0.8997	0.9203	0.9259	0.9227

Рисунок 20 – Показатели оценивания алгоритма анализа видеоинформации АВИ для 10 тестовых мультимодальных файлов

### 5.3.3. Структура (пайплайн) для алгоритма анализа текстовой информации АТИ

Структура (пайплайн) компонента OCEAN-AI для алгоритма АТИ для оценивания ПКЛЧ по текстовым признакам разговорной речи человека представлена на рис. 21.

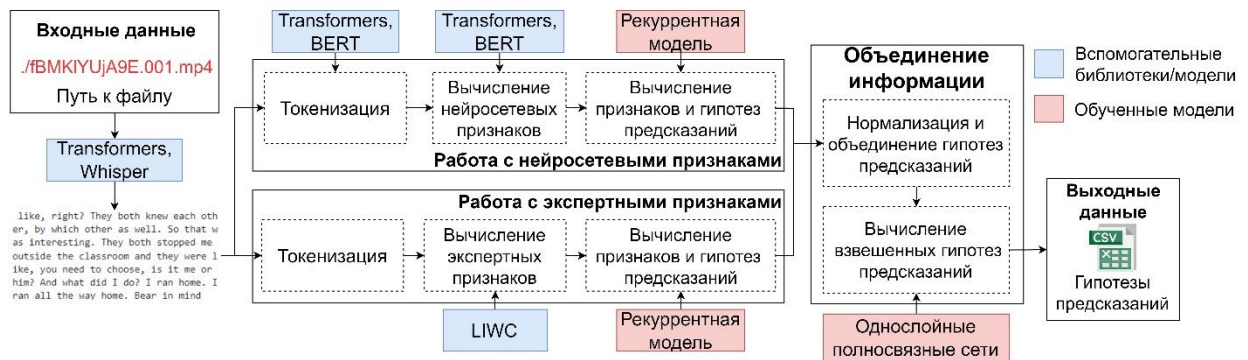


Рисунок 21 – Структура (пайплайн) компонента OCEAN-AI для алгоритма анализа видеоинформации АТИ для оценивания ПКЛЧ по признакам разговорной речи человека

Для получения оценок ПКЛЧ по текстовой информации согласно рис. 21 для 10 тестовых мультимодальных файлов из корпуса First Impressions V2 необходимо выполнить последовательность действий:

```
# Импорт необходимых инструментов
import os
import pandas as pd

# Импорт модуля
from oceanai.modules.lab.build import Run

# Создание экземпляра класса
_b5 = Run()

# Настройка ядра
_b5.path_to_save_ = './models' # Директория для сохранения файла
_b5.chunk_size_ = 2000000 # Размер загрузки файла из сети за 1 шаг

# Формирование вспомогательных моделей для обработки текстовой информации
_b5.load_text_features()
_b5.setup_translation_model()
_b5.setup_bert_encoder()

# Формирование текстовых моделей
res_load_model_hc = _b5.load_text_model_hc(corpus='fi')
res_load_model_nn = _b5.load_text_model_nn(corpus='fi')

# Загрузка весов текстовых моделей
url = _b5.weights_for_big5['text']['fi']['hc']['sberdisk']
res_load_model_weights_hc = _b5.load_text_model_weights_hc(url = url)

url = _b5.weights_for_big5['text']['fi']['nn']['sberdisk']
res_load_model_weights_nn = _b5.load_text_model_weights_nn(url = url)

# Формирование модели для анализа текстовой информации
res_load_model_b5 = _b5.load_text_model_b5()

# Загрузка весов модели для анализа текстовой информации
url = _b5.weights_for_big5['text']['fi']['b5']['sberdisk']
res_load_model_weights_b5 = _b5.load_text_model_weights_b5(url = url)

PATH_TO_DIR = './video_FI/'
PATH_SAVE_VIDEO = './video_FI/test/'

_b5.path_to_save_ = PATH_SAVE_VIDEO
```

```
# Загрузка 10 тестовых мультимодальных файлов из корня First Impressions V2
# URL: https://chalearnlap.cvc.uab.cat/dataset/24/description/
domain = 'https://download.sberdisk.ru/download/file/'
tets_name_files = [
    '429713680?token=FqHdMLSSh7zYSZt&filename=_plk5k7PBEg.003.mp4',
    '429713681?token=Hz9b4lQkrLfic33&filename=be0DQawtVKE.002.mp4',
    '429713683?token=EgUXS9Xs8xHm5gz&filename=2d6btbaNdfo.000.mp4',
    '429713684?token=1U26753kmPYdIgt&filename=300gK3CnzW0.003.mp4',
    '429713685?token=LyigAWLTzDNwKJO&filename=300gK3CnzW0.001.mp4',
    '429713686?token=EpFRbCKHyuc4HPu&filename=cLaZxEf1nE4.004.mp4',
    '429713687?token=FNTkwqBr4jOS95l&filename=g24JGYuT74A.004.mp4',
    '429713688?token=qDT95nz7hfm2Nki&filename=JZNMxa3OKHY.000.mp4',
    '429713689?token=noLguEGXDpbckHg&filename=nvlqJbHk_Lc.003.mp4',
    '429713679?token=9L7RQ0hgdJlcek6&filename=4vdJGgZpj4k.003.mp4'
]

for curr_files in tets_name_files:
    _b5.download_file_from_url(url = domain + curr_files, out = False)

# Получение прогнозов
_b5.path_to_dataset_ = PATH_TO_DIR # Директория набора данных
_b5.ext_ = ['.mp4'] # Расширения искоемых файлов

# Полный путь к файлу с верными предсказаниями для подсчета точности
url_accuracy = _b5.true_traits_['fi']['sberdisk']

_b5.get_text_union_predictions(url_accuracy = url_accuracy, lang='en')
```

Результаты оценивания ПКЛЧ по текстовой информации представлены на рис. 22 и 23.

	Path	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism
<b>Person ID</b>						
1	2d6btbaNdfo.000.mp4	0.568848	0.556875	0.459379	0.577024	0.54314
2	300gK3CnzW0.001.mp4	0.664632	0.626621	0.583274	0.635476	0.64077
3	300gK3CnzW0.003.mp4	0.592807	0.531285	0.505754	0.572473	0.54919
4	4vdJGgZpj4k.003.mp4	0.513681	0.529299	0.42787	0.530636	0.488311
5	be0DQawtVKE.002.mp4	0.587146	0.554922	0.488549	0.571464	0.552106
6	cLaZxEf1nE4.004.mp4	0.521515	0.456004	0.442764	0.504145	0.45683
7	g24JGYuT74A.004.mp4	0.59935	0.514481	0.509759	0.571018	0.548252
8	JZNMxa3OKHY.000.mp4	0.562864	0.473718	0.477902	0.528514	0.494847
9	nvlqJbHk_Lc.003.mp4	0.570894	0.580698	0.466853	0.577114	0.545557
10	_plk5k7PBEg.003.mp4	0.594539	0.643821	0.494306	0.613988	0.594732

Рисунок 22 – Предсказанные оценки ПКЛЧ по текстовой информации



	Openness	Conscientiousness	Extraversion	Agreeableness	Non-Neuroticism	Mean
Metrics						
MAE	0.1123	0.1024	0.1619	0.1247	0.1175	0.1238
Accuracy	0.8877	0.8976	0.8381	0.8753	0.8825	0.8762

Рисунок 23 – Показатели оценивания алгоритма анализа текстовой информации АТИ для 10 тестовых мультимодальных файлов

5.3.4. Структура (пайплайн) для алгоритма мультимодального объединения информации МОИ

Структура (пайплайн) компонента OCEAN-AI для алгоритма МОИ для оценивания ПКЛЧ по мультимодальным характеристикам представлена на рис. 24.

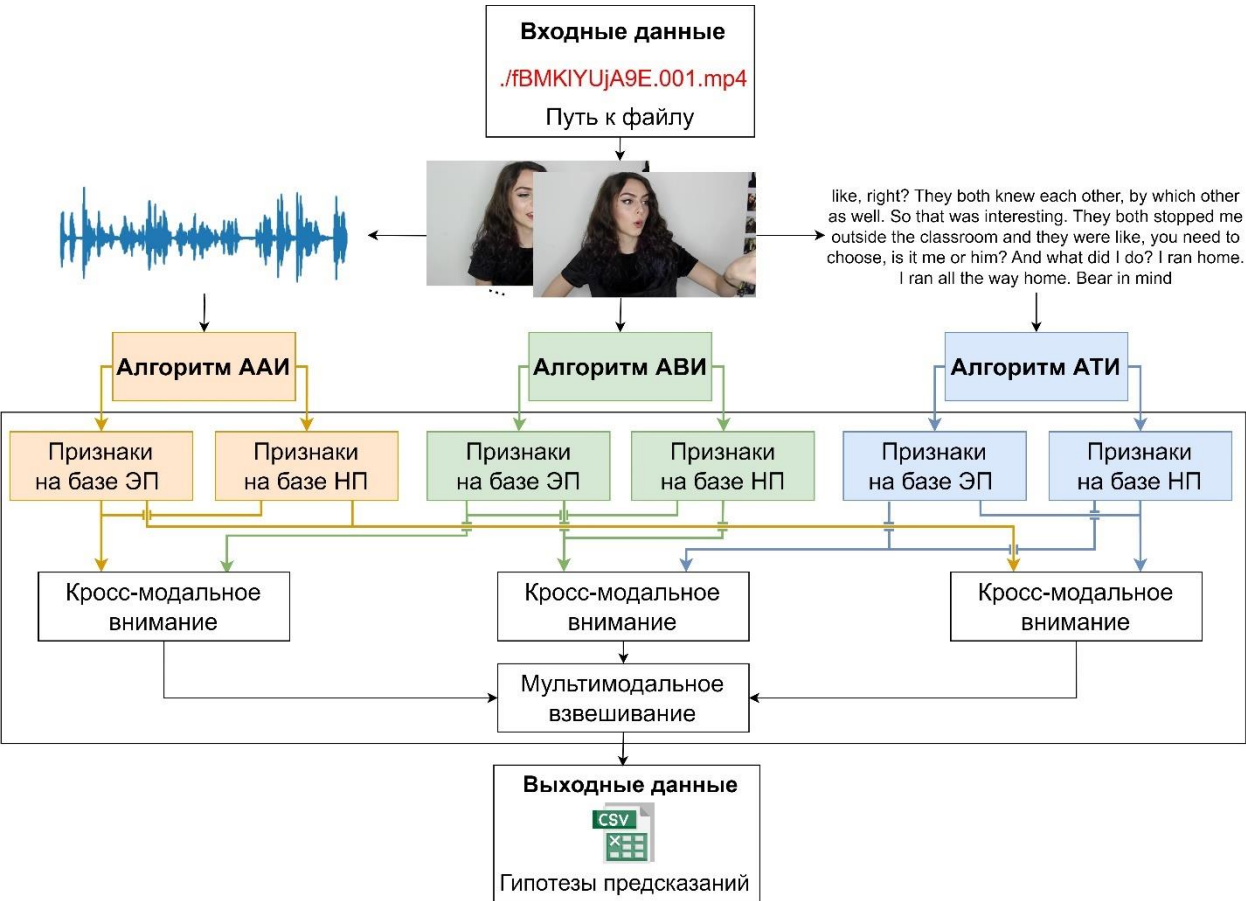


Рисунок 24 – Структура (пайплайн) компонента OCEAN-AI для алгоритма мультимодального объединения информации МОИ для оценивания ПКЛЧ по мультимодальным характеристикам

Для получения мультимодальных оценок ПКЛЧ согласно рис. 24 для 10 тестовых мультимодальных файлов из корпуса First Impressions V2 необходимо выполнить последовательность действий представленный в разделе 3.2.1.

#### **5.4. Интеграция библиотеки в платформу проекта**

Библиотека OCEAN-AI может использоваться как непосредственно, так и в рамках инструментальной облачной платформы для проектирования, быстрой разработки и обучения прикладных систем ИИ RU.СНАБ-00845 (DataMall 2.0)

Для интеграции OCEAN-AI в платформе создается контейнер. Для сборки образа контейнера необходимо дополнительно подготовить файл Dockerfile с инструкциями, подробнее - <https://docs.docker.com/engine/reference/builder/>.

Образ должен соответствовать следующим требованиям:

- а) должен быть установлен jupyter notebook или jupyter lab;
- б) Entrypoint должен быть пустым;
- в) CMD должен запускать jupyter;
- г) пользователь, от имени которого запускается jupyter должен быть `jovyan`;
- д) вся функциональность модуля должна быть доступна для пользователя `jovyan`;
- е) должны быть добавлены / развернуты требуемое окружение и соответствующие программные библиотеки для функционирования основной библиотеки;
- ж) при необходимости добавляются ноутбуки с примерами реализации функциональных возможностей.

### **6. ПРОВЕРКА ПРОГРАММЫ**

#### **6.1. Модульные и интеграционные тесты**

Компонент OCEAN-AI включает три модульных и пять интеграционных теста. Модульные тесты (`test_get_acoustic_features()`, `test_get_visual_features()`, `test_get_text_features()`, размещенные по адресу <https://gitlab.actcognitive.org/itmo-sai-code/oceanai/-/tree/main/test> в файлах `unit_tests_fi.py` и `unit_tests_mupta.py`) проверяют размерность выходных акустических, визуальных и текстовых признаков и работают с методами для:

1. Извлечения акустических низкоуровневых признаков для модульного теста `test_get_acoustic_features()`:

- `get_acoustic_features()`.



2. Извлечения визуальных низкоуровневых признаков для модульного теста `test_get_visual_features()`:

- `get_visual_features()`;
- `load_video_model_deep_fe()`;
- `load_video_model_weights_deep_fe()`;

3. Извлечения текстовых признаков для модульного теста `test_get_text_features()`:

- `load_text_features()`;
- `setup_translation_model()`;
- `setup_bert_encoder()`.

4. Общий метод для загрузки тестовых записей с облачного хранилища для всех модульных тестов:

- `download_file_from_url()`.

Интеграционные тесты (`test_get_audio_union_predictions()`, `test_get_video_union_predictions()`, `test_get_text_union_predictions()`, `test_get_av_union_predictions()`, `test_get_avt_predictions()`, размещенные по адресу <https://gitlab.actcognitive.org/itmo-sai-code/oceanai/-/tree/main/test> в файлах `unit_tests_fi.py` и `unit_tests_mupta.py`) проверяют показатели оценивания ПКЛЧ для алгоритмов ААИ, АВИ, АТИ и МОИ и работают с методами для:

1. Предобработки аудиоинформации, вычисления акустических признаков и выдача по ним гипотез предсказаний оценок ПКЛЧ для интеграционного теста `test_get_audio_union_predictions()`:

- `load_audio_model_hc()`;
- `load_audio_model_nn()`;
- `load_audio_models_b5()`;
- `load_audio_model_weights_hc()`;
- `load_audio_model_weights_nn()`;
- `load_audio_models_weights_b5()`;
- `get_audio_union_predictions()`.

2. Предобработки видеоинформации, вычисления визуальных признаков и выдача по ним гипотез предсказаний оценок ПКЛЧ для интеграционного теста `test_get_video_union_predictions()`:

- `load_video_model_deep_fe()`;
- `load_video_model_hc()`;

- load\_video\_model\_nn();
- load\_video\_models\_b5();
- load\_video\_model\_weights\_deep\_fe();
- load\_video\_model\_weights\_hc();
- load\_video\_model\_weights\_nn();
- load\_video\_models\_weights\_b5();
- get\_video\_union\_predictions().

3. Предобработки текстовой информации, вычисления текстовых признаков и выдача по ним гипотез предсказаний оценок ПКЛЧ для интеграционного теста test\_get\_text\_union\_predictions():

- load\_text\_features();
- setup\_translation\_model();
- setup\_bert\_encoder();
- load\_text\_model\_hc();
- load\_text\_model\_nn();
- load\_text\_model\_b5();
- load\_text\_model\_weights\_hc();
- load\_text\_model\_weights\_nn();
- load\_text\_model\_weights\_b5();
- get\_text\_union\_predictions().

4. Выполнения мультимодального объединения среднеуровневых признаков, полученных от аудио- и видеoinформации, и получения мультимодальных гипотез предсказаний оценок ПКЛЧ для интеграционного теста test\_get\_av\_union\_predictions():

- load\_audio\_model\_hc();
- load\_audio\_model\_nn();
- load\_audio\_model\_weights\_hc();
- load\_audio\_model\_weights\_nn();
- load\_video\_model\_deep\_fe();
- load\_video\_model\_hc();
- load\_video\_model\_nn();
- load\_video\_model\_weights\_deep\_fe();
- load\_video\_model\_weights\_hc();
- load\_video\_model\_weights\_nn();

- `load_av_models_b5();`
- `load_av_models_weights_b5();`
- `get_av_union_predictions().`

5. Выполнения мультимодального объединения среднеуровневых признаков, полученных от аудио-, видео- и текстовой информации, и получения мультимодальных гипотез предсказаний оценок ПКЛЧ для интеграционного теста `test_get_avt_predictions()`:

- `load_audio_model_hc();`
- `load_audio_model_nn();`
- `load_audio_model_weights_hc();`
- `load_audio_model_weights_nn();`
- `load_video_model_deep_fe();`
- `load_video_model_hc();`
- `load_video_model_nn();`
- `load_video_model_weights_deep_fe();`
- `load_video_model_weights_hc();`
- `load_video_model_weights_nn();`
- `load_text_features();`
- `setup_translation_model();`
- `setup_bert_encoder();`
- `load_text_model_hc();`
- `load_text_model_nn();`
- `load_text_model_weights_hc();`
- `load_text_model_weights_nn();`
- `load_avt_model_b5();`
- `load_avt_model_weights_b5();`
- `get_avt_predictions().`

6. Общий метод для загрузки тестовых записей с облачного хранилища для всех интеграционных тестов:

- `download_file_from_url().`

Интегральные и модульные тесты размещены по адресу <https://gitlab.actcognitive.org/itmo-sai-code/oceanai/-/tree/main/test> в двух файлах `unit_tests_fi.py` и `unit_tests_mufta.py`. Это сделано для того, чтобы проверить работу

реализованных алгоритмов на двух корпусах (First Impressions V2 и MuPTA). В общей сложности на два корпуса представлено 13 интегральных и модульных текста.

Пример выполнения модульных и интеграционных тестов представлен на рис. 25.

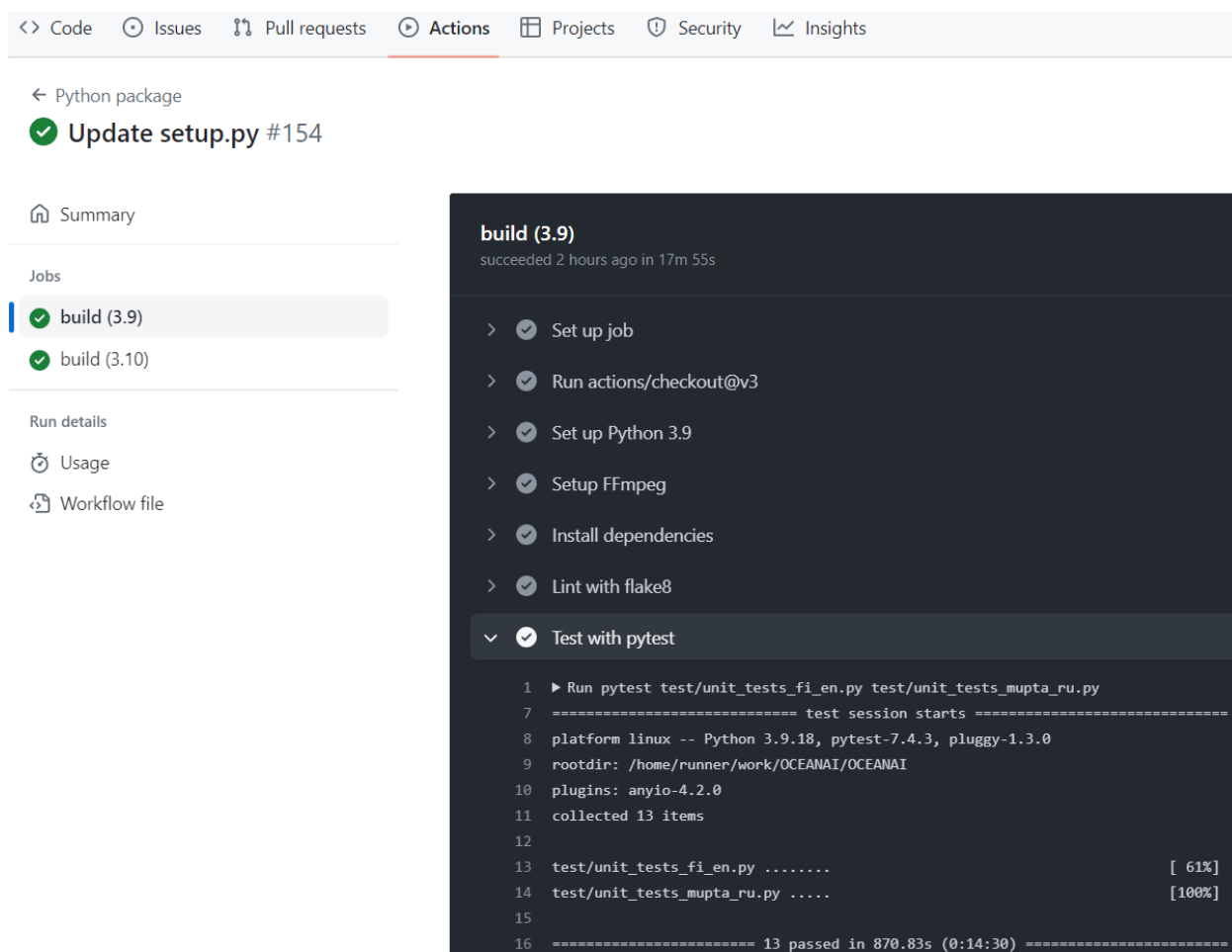


Рисунок 25 – Пример выполнения модульных и интеграционных тестов

Рис. 25 демонстрирует, что успешно выполнены 13 тестов из 13 возможных на двух версиях Python: 3.9 и 3.10.

## 6.2. Контрольные примеры

Контрольные примеры рассмотрены для:

- модульного теста извлечения акустических признаков – `test_get_acoustic_features()`;
- интеграционного теста оценивания ПКЛЧ по видеoinформации – `test_get_video_union_predictions()`.

В качестве входных данных для выполнения контрольного примера работы метода `get_acoustic_features()`, тестируемого модульным тестом `test_get_acoustic_features()`,

выступает путь к аудиозаписи glgfB3vFewc.004.mp4, размещенной по адресу <https://hci.nw.ru/ru/pages/ocean-ai>. Код для выполнения контрольного примера в среде Jupyter Lab:

```
from oceanai.modules.lab.build import Run

_b5 = Run()

PATH_SAVE_VIDEO = './video_FI/test/'
FILENAME = 'glgfB3vFewc.004.mp4'

_b5.path_to_save_ = PATH_SAVE_VIDEO
domain = 'https://download.sberdisk.ru/download/file/'

_b5.download_file_from_url(url = domain + '425715491?token=EyJnTnsfmJIIsb0&filename=' + FILENAME, out = False)

hc_features, melspectrogram_features = _b5.get_acoustic_features(path = PATH_SAVE_VIDEO + FILENAME, out = True)
```

Выходными данными метода `get_acoustic_features()` являются два массива с извлеченными признаками. Выходные сведения массивов признаков представлены на рис. 26.

```
[2023-01-12 17:34:48] Извлечение признаков (экспертных и лог мел-спектрограмм) из акустического сигнала ...
[2023-01-12 17:34:49] Статистика извлеченных признаков из акустического сигнала:
Общее количество сегментов с:
1. экспертными признаками: 12
2. лог мел-спектрограммами: 12
Размерность матрицы экспертных признаков одного сегмента: 196 × 25
Размерность тензора с лог мел-спектрограммами одного сегмента: 224 × 224 × 3
--- Время выполнения: 1.084 сек. ---
```

Рисунок 26 – Выходные сведения массивов признаков, полученных методом `get_acoustic_features()`

В качестве входных данных для выполнения контрольного примера работы метода `get_video_union_predictions()`, тестируемого интеграционным тестом `test_get_video_union_predictions()`, выступают пути к двум видеозаписям (glgfB3vFewc.004.mp4 и 6V807Mf\_gHM.003.mp4), размещенным по адресу <https://hci.nw.ru/ru/pages/ocean-ai>.

Код для выполнения контрольного примера в среде Jupyter Lab:

```
from oceanai.modules.lab.build import Run

_b5 = Run()
_b5.ext_ = ['.mp4']
CORPUS = 'fi'
LANG = 'en'
URL_ACCURACY = _b5.true_traits_[CORPUS]['sberdisk']
PATH_TO_DIR = './video_FI/'
PATH_SAVE_VIDEO = './video_FI/test/'
PATH_SAVE_MODELS = './models'

_b5.path_to_save_ = PATH_SAVE_VIDEO
domain = 'https://download.sberdisk.ru/download/file/'
```

```
tets_name_files = ['425715491?token=EyJnTnsfmJIlsb0&filename=glgfB3vFewc.004.mp4',
                  '425715497?token=00tfwHAaMeMRZMi&filename=6V807Mf_gHM.003.mp4']

for curr_files in tets_name_files:
    _b5.download_file_from_url(url = domain + curr_files, out = False)

_b5.path_to_save_ = PATH_SAVE_MODELS
_b5.path_to_dataset_ = PATH_TO_DIR

_b5.load_video_model_hc(lang = LANG, out = False)
url = _b5.weights_for_big5_['video'][CORPUS]['hc']['sberdisk']
_b5.load_video_model_weights_hc(url = url, out = False)

_b5.load_video_model_deep_fe(out = False)
url = _b5.weights_for_big5_['video'][CORPUS]['fe']['sberdisk']
_b5.load_video_model_weights_deep_fe(url = url, out = False)

_b5.load_video_model_nn(out = False)
url = _b5.weights_for_big5_['video'][CORPUS]['nn']['sberdisk']
_b5.load_video_model_weights_nn(url = url, out = False)

_b5.load_video_models_b5(out = False)
url_OPE = _b5.weights_for_big5_['video'][CORPUS]['b5']['openness']['sberdisk']
url_CON = _b5.weights_for_big5_['video'][CORPUS]['b5']['conscientiousness']['sberdisk']
url_EXT = _b5.weights_for_big5_['video'][CORPUS]['b5']['extraversion']['sberdisk']
url_AGR = _b5.weights_for_big5_['video'][CORPUS]['b5']['agreeableness']['sberdisk']
url_NNEU = _b5.weights_for_big5_['video'][CORPUS]['b5']['non_neuroticism']['sberdisk']
_b5.load_video_models_weights_b5(
    url_openness = url_OPE,
    url_conscientiousness = url_CON,
    url_extraversion = url_EXT,
    url_agreeableness = url_AGR,
    url_non_neuroticism = url_NNEU,
    out = False
)

_b5.get_video_union_predictions(url_accuracy = URL_ACCURACY, out = True, lang = LANG)
```

Выходными данными метода `get_video_union_predictions()` являются два файла с гипотезами предсказаний оценок ПКЛЧ и показателями качества алгоритма АВИ. Выходные данных проиллюстрированы на рис. 27.

[2023-01-12 17:45:54] Получение прогнозов и вычисление точности (видео модальность) ...

2 из 2 (100.0%) ... Users\Elena Ryumina\Desktop\work\OCEANA\video\_FI\test\glgfB3vFewc.004.mp4 ...

ID	Path	Openness	Conscientiousness	Extraversion	Agreeableness	Neuroticism
1	C:\Users\Elena Ryumina\Desktop\work\OCEANA\vi...	0.686578	0.663079	0.626678	0.643983	0.655583
2	C:\Users\Elena Ryumina\Desktop\work\OCEANA\vi...	0.668936	0.790461	0.579965	0.678129	0.686302

[2023-01-12 17:45:54] Точность по отдельным персональным качествам личности человека ...

	Openness	Conscientiousness	Extraversion	Agreeableness	Neuroticism	Mean
Metrics						
MAE	0.1056	0.0977	0.0514	0.0731	0.156	0.0968
Accuracy	0.8944	0.9023	0.9486	0.9269	0.844	0.9032

[2023-01-12 17:45:54] Средняя средних абсолютных ошибок: **0.0968**, средняя точность: **0.9032** ...

Лог файлы успешно сохранены ...

--- Время выполнения: 7.794 сек. ---

Рисунок 27 – Выходные данные, полученные методом `get_video_union_predictions()`

Рис. 27 показывает, что для двух тестовых видеозаписей достигается значение Accuracy = 90,32% для алгоритма АВИ. В зависимости от операционной системы (Windows/Linux/MacOS) гипотезы предсказаний оценок ПКЛЧ, представленные в первой таблице, изменяются на тысячные, что влияет на показатели качества алгоритма АВИ. Гипотезы предсказанных оценок ПКЛЧ могут быть применены для решения практических задач, рассмотренных в разделах 3.2.2, 3.2.3 и 3.2.4.

## 7. ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ

### 7.1. Входные данные

В табл. 3 представлены состав и структура входных данных.

Таблица 3 – Состав и структура входных данных

Наименование данных	Обозначение	Структура данных	Способ ввода данных	Ограничения
Аудио-данные	$b_i^A$ , $i = 1, \dots, n$	Переменная-массив audio класса oceanai.modules.la b.audio.Audio	Файл с аудиоданными	Формат mp4, avi, mov. Продолжительность сигнала до 170 сек.
Видео-данные	$b_i^V$ , $i = 1, \dots, n$	Переменная-массив frame класса oceanai.modules.la b.audio.Video	Файл с видеоданными	Формат mp4, avi. Продолжительность сигнала от 3 до 15 сек.
Текстовые данные	$b_i^T$ , $i = 1, \dots, n$	Переменная-массив text класса oceanai.modules.la b.text.Text	Файл с текстовыми данными	Формат txt, mp4, avi, mov. Продолжительность сигнала для форматов mp4, avi, mov до 170 сек. Длина текстового файла не более 2000 символов.

Чтобы обеспечить библиотеку данными, необходимо указать папку размещения соответствующих мультимодальных файлов через свойство path\_to\_dataset\_ класса oceanai.modules.core.settings.Settings, расширение к файлам устанавливается через свойство ext\_ класса oceanai.modules.core.settings.Settings, глубина иерархии устанавливается аргументом depth метода методов get\_av\_union\_predictions(), get\_avt\_union\_predictions(), get\_audio\_union\_predictions(), get\_video\_union\_predictions() и get\_text\_union\_predictions().

```
_b5 = Run()  
_b5.path_to_dataset_ = '/Users/dl/GitHub/oceanai/oceanai/dataset/First Impressions'
```

```
_b5.ext_ = ['.mp4'] # Расширения искомых файлов
res_get_avt_union_predictions = _b5.get_avt_union_predictions(
    depth = 2, # Глубина иерархии для получения аудио и видеоданных
    # Другие аргументы данного метода не относятся к подготовке данных
    # ...
)
```

## 7.2. Выходные данные

Состав и структура выходных данных. Выходными данными компонента OCEAN-AI являются файлы в формате DataFrame библиотеки Pandas. Один DataFrame содержит информацию о полных путях к мультимодальным файлам и предсказанным оценкам ПКЛЧ в табличном виде. Второй DataFrame содержит информацию о оценочных результатах работы компонента OCEAN-AI в показателях оценивания MAE и Accuracy, рассчитанных для каждого качества и их средние значения. В табл. 4 представлены состав и структура выходных данных.

Таблица 4 – Состав и структура выходных данных.

Наименование данных	Обозначение	Структура данных	Способ вывода данных	Ограничения
Объединенные гипотезы предсказаний	<i>U</i>	Переменная-матрица (DataFrame) <code>_df_files</code> класса <code>oceanai.modules.lab.prediction.Prediction</code>	Вызов метода <code>get_av_union_predictions()</code> класса <code>oceanai.modules.lab.prediction.Prediction</code>	Значения в интервале [0,1]
Показатели оценивания методов	<i>Z</i>	Переменная-матрица (DataFrame) <code>_df_accuracy</code> класса <code>oceanai.modules.lab.prediction.Prediction</code>	Вызов метода <code>get_av_union_predictions()</code> класса <code>oceanai.modules.lab.prediction.Prediction</code>	Значения в интервале [0,1]

Подготовка выходных данных. Программист может задать порядок вывода ПКЛЧ обозначив их через свойство `keys_dataset_` класса `Settings`, также указать путь к сохранению выходных файлов через свойство `path_to_logs_` класса `Settings`, указать путь к экспертным оценкам ПКЛЧ для подсчёта точности в аргументе `url_accuracy` метода `get_avt_union_predictions()`, и установить аргумент `logs` на `True` в методе `get_avt_union_predictions()` для сохранения файлов. Язык для использования необходимых весов моделей устанавливается через аргумент `lang` на `'en'` или `'ru'` в методе `get_avt_union_predictions()`. Предусматривается использование не только «англоязычных моделей» для англоязычных входных данных либо «русскоязычных моделей» на



русскоязычных входных данных, но и «англоязычных моделей» на русскоязычных входных данных. Отображения файлов регулируется аргументом `out` метода `get_video_union_predictions()`. Выходные данные интерпретируются в формате «csv».

```
_b5 = Run()
_b5.keys_dataset_ = ['Path', 'Openness', 'Conscientiousness', 'Extraversion', 'Agreeableness',
'Non-Neuroticism']
_b5.path_to_logs_ = './logs' # Директория для сохранения LOG файлов
res_get_avt_union_predictions = _b5.get_avt_predictions(
    logs = True, # При необходимости формировать LOG файл
    out = True, # Отображение
    lang = 'en', # Выбор языка для моделей First Impressions V2 'en' для МПТА 'ru'
    # Другие аргументы данного метода не относятся к интерпретации выходных данных
    # ...
)
```

Примеры отображения «csv» файлов представлены на рис. 1, 2, 16, 17, 19, 20, 22, 23 и 27.

## 8. СООБЩЕНИЯ

Таблица 5 – Сообщения

Сообщение	Значение
[Дата Время] Не определен размер файла для загрузки	Не удалось загрузить файл
[Дата Время] Указан слишком маленький размер окна сегмента сигнала	Не удалось извлечь экспертные признаки
[Дата Время] Что-то пошло не так ... смотрите настройки ядра и цепочку выполнения действий ...	Не верные ключи в словаре
[Дата Время] Что-то пошло не так ... директория "... " не найдена ...	Файл по указанной директории не найден
[Дата Время] Что-то пошло не так ... нейросетевая архитектура модели для получения оценок по ... признакам не сформирована (... модальность) ...	Нейросетевая модель не загружена
[Дата Время] Что-то пошло не так ... Не определен размер файла для загрузки ...	Не удалось загрузить файл
[Дата Время] Что-то пошло не так ... не удалось обработать указанный URL ...	Указан не верный URL
[Дата Время] Что-то пошло не так ... не удалось загрузить веса нейросетевой модели ... Открытость опыту	Не удалось загрузить модель для ПКЛЧ «Открытость опыту»

[Дата Время] Что-то пошло не так ... конкатенация оценок по экспертным и нейросетевым признакам не произведена (... модальность) ...	Не удалось выполнить конкатенацию признаков
[Дата Время] Что-то пошло не так ... нейросетевая архитектура модели для получения нейросетевых признаков не сформирована (... модальность) ...	Не сформирована модель
[Дата Время] Неверные типы или значения аргументов в "..."	Ошибка в типе или значении аргумента
--- Время выполнения: ... сек. --	Время выполнения команды в секундах
<p>[Дата Время] Статистика извлеченных признаков из акустического сигнала:</p> <p>Общее количество сегментов с:</p> <ol style="list-style-type: none"> <li>1. экспертными признаками: ...</li> <li>2. лог-мел-спектрограммами: ...</li> </ol> <p>Размерность матрицы экспертных признаков одного сегмента: ... × ...</p> <p>Размерность тензора с лог-мел-спектрограммами одного сегмента: ... × ... × ...</p>	Вывод статистической информации о извлеченных признаках из акустического сигнала
<p>[2022-11-03 16:18:40] Статистика извлеченных признаков из визуального сигнала:</p> <p>Общее количество сегментов с:</p> <ol style="list-style-type: none"> <li>1. экспертными признаками: ...</li> <li>2. нейросетевыми признаками: ...</li> </ol> <p>Размерность матрицы экспертных признаков одного сегмента: ... × ...</p> <p>Размерность тензора с нейросетевыми признаками одного сегмента: ... × ...</p> <p>Понижение кадровой частоты: с ... до ...</p>	Вывод статистической информации о извлеченных признаках из визуального сигнала
[Дата Время] Выполнение заблокировано пользователем ...	блокировка выполнения команды пользователем

[Дата Время] Извлечение признаков (... и ...) из ... сигнала ...	Извлечение признаков
[Дата Время] Загрузка весов нейросетевой модели для получения оценок по ... признакам (... модальность) ...	Выполнение загрузки файла
[Дата Время] Загрузка весов нейросетевой модели для получения нейросетевых признаков (... модальность) ...	
[Дата Время] Загрузка весов нейросетевых моделей для получения результатов оценки персональных качеств (...модальность) ...	
[Дата Время] Загрузка весов нейросетевой модели	
[Дата Время] Загрузка файла "... " (100.0%) ...	
[Дата Время] Формирование нейросетевой архитектуры модели для получения оценок по ... признакам (...модальность) ...	Модель сформирована
[Дата Время] Формирование нейросетевой архитектуры для получения нейросетевых признаков (...модальность) ...	
[Дата Время] Формирование нейросетевых архитектур моделей для получения результатов оценки персональных качеств (...модальность) ...	
[Дата Время] Получение прогнозов и вычисление точности (...модальность) ...	Получение прогнозов и вычисление точности
[Дата Время] Точность по отдельным персональным качествам личности человека ...	Получение сведений о точности
[Дата Время] Средняя средних абсолютных ошибок: ..., средняя точность: ...	Получение сведений о точности
Лог файлы успешно сохранены ...	Успешное сохранение выходных данных
... из ... (...%) ... НАЗВАНИЕ ФАЙЛА ...	Процент обработанных записей

## ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

AI	–	Искусственный интеллект (Artificial Intelligence)
OCEAN	–	Персональные качества личности человека: Открытость опыту (Openness to experience), Добросовестность (Conscientiousness), Экстраверсия (Extraversion), Доброжелательность (Agreeableness), Эмоциональная стабильность (Non-Neuroticism)
OCEAN-AI	–	Библиотека алгоритмов интеллектуального анализа поведения человека на основе его мультимодальных данных, обеспечивающих оценивание уровня отдельных персональных качеств личности человека для выполнения профессиональных обязанностей
ААИ	–	Анализ аудиоинформации
АВИ	–	Анализ видеоинформации
АТИ	–	Анализ текстовой информации
МОИ	–	Мультимодальное объединение информации
ПКЛЧ	–	Персональные качества личности человека

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Hugo Jair Escalante et al. Modeling, recognizing, and explaining apparent personality from videos // *IEEE Transactions on Affective Computing*, 2022, vol. 13(2), pp. 894–911.
2. Ryumina E., Ryumin D., Markitantov M., Kaya H., Karpov A. Multimodal Personality Traits Assessment (MuPTA) Corpus: The Impact of Spontaneous and Read Speech // *Interspeech*, 2023, pp. 4049–4053.
3. Giritlioglu D. et al. Multimodal analysis of personality traits on videos of self-presentation and induced behavior. *Journal on Multimodal User Interfaces*, 2021, 15, 337–358. doi:10.1007/s12193-020-00347-7.
4. Aslan S. et al. Multimodal assessment of apparent personality using feature attention and error consistency constraint // *Image and Vision Computing*, 2021, vol. 110, 104163 P.
5. Sajjad H. et al. Personality and career choices // *African Journal of Business Management*, 2012, vol. 6 (6), pp. 2255-2260.
6. Alkhelil A. H. The relationship between personality traits and career choice: A case study of secondary school students // *International Journal of Academic Research in Progressive Education and Development*, 2016, vol. 5(2), pp. 2226-6348.
7. De Jong N. et al. Personality traits and career role enactment: Career role preferences as a mediator // *Frontiers in psychology*, 2019, vol. 10, pp. 1720.
8. Wehner C., de Grip A., Pfeifer H. Do recruiters select workers with different personality traits for different tasks? A discrete choice experiment // *Labour Economics*, 2022, vol. 78, pp. 102186.
9. O'Connor P. J. et al. What drives consumer automobile choice? Investigating personality trait predictors of vehicle preference factors // *Personality and individual differences*, 2022, vol. 184, pp. 111220.
10. Peltonen E., Sharmila P., Asare K. O., Visuri A., Lagerspetz E., Ferreira D. When phones get personal: Predicting Big Five personality traits from application usage // *Pervasive and Mobile Computing*, 2020, vol. 69, pp. 101269.
11. Kuroda S., Yamamoto I. Good boss, bad boss, workers' mental health and productivity: Evidence from Japan // *Japan & The World Economy*, 2018, vol. 48, pp. 106-118.
12. Velichko, A., Markitantov, M., Kaya, H., Karpov, A. Complex Paralinguistic Analysis of Speech: Predicting Gender, Emotions and Deception in a Hierarchical Framework // *Interspeech*, 2022, pp. 4735–4739.

13. Eyben F., Wöllmer M., Schuller B. OpenSMILE: The Munich Versatile and Fast Open-Source Audio Feature Extractor // 18th ACM International Conference on Multimedia, 2010, pp. 1459–1462.
14. Markitantov M., Ryumina E., Ryumin D et al. Biometric Russian Audio-Visual Extended MASKS (BRAVE-MASKS) Corpus: Multimodal Mask Type Recognition Task // Interspeech, 2022, pp. 1756–1760.
15. Ryumina E., Dresvyanskiy D., Karpov A. In Search of a Robust Facial Expressions Recognition Model: A Large-Scale Visual Cross-Corpus Study // Neurocomputing - 2022, Vol. 514, pp. 435–450.
16. Ryumina E., Markitantov M., Ryumin D., Karpov A. OCEAN-AI framework with EmoFormer cross-hemiface attention approach for personality traits assessment // Expert Systems with Applications, 2024, vol. 239, pp. 122441.
17. Ivanko, D., Ryumin, D., Kashevnik, A., Axyonov, A., Kitenko, A., Lashkov, I., Karpov, A. (2022) DAVIS: Driver's Audio-Visual Speech recognition // Interspeech, 2022, 1141–1142.
18. Ponce-López V. et al. Chalearn lap 2016: First round challenge on first impressions-dataset and results // European conference on computer vision, 2016, pp. 400-418.

[illegible]