Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

На правах рукописи

Фахрутдинов Адель Фердинандович

ПРОГРАММНЫЙ КОМПЛЕКС АНАЛИЗА ПОКАЗАТЕЛЕЙ ЖИЗНЕДЕЯТЕЛЬНОСТИ ЧЕЛОВЕКА С ИСПОЛЬЗОВАНИЕМ САМОАДАПТИРУЮЩЕЙСЯ МИКРОСЕРВИСНОЙ АРХИТЕКТУРЫ

Специальность: 2.3.5. – Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей

Диссертация
на соискание ученой степени кандидата технических наук

Научный руководитель:
доктор технических наук, доцент
Чикрин Дмитрий Евгеньевич

Казань – 2026 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ **5**

ГЛАВА 1. АНАЛИЗ ПРОГРАММНО-АППАРАТНЫХ КОМПЛЕКСОВ НЕИНВАЗИВНОГО МОНИТОРИНГА ПОКАЗАТЕЛЕЙ ЖИЗНЕДЕЯТЕЛЬНОСТИ ЧЕЛОВЕКА **16**

1.1 Портативные программно-аппаратные комплексы неинвазивного мониторинга показателей жизнедеятельности человека **16**

1.1.1 Классификация комплексов неинвазивного мониторинга состояния здоровья человека **17**

1.1.2 Технологии, применяемые в программно-аппаратных комплексах неинвазивного мониторинга показателей жизнедеятельности человека **18**

1.1.3 Определяемые программно-аппаратным комплексом параметры неинвазивного мониторинга **23**

1.1.4 Сравнение реализаций программно-аппаратных комплексов неинвазивной диагностики показателей жизнедеятельности человека **27**

1.2 Автоматизированный программно-аппаратный комплекс неинвазивного мониторинга состояния жизнедеятельности человека **30**

1.3 Инфокоммуникационный функционал **32**

1.4 Выводы по главе **34**

ГЛАВА 2. САМОАДАПТИРУЮЩАЯСЯ МИКРОСЕРВИСНАЯ АРХИТЕКТУРА С ИСПОЛЬЗОВАНИЕМ ПЕРСОНАЛИЗИРОВАННОГО ФЕДЕРАТИВНОГО ОБУЧЕНИЯ МОДЕЛЕЙ **36**

2.1 Архитектурная проблематика **36**

2.1.1 Базовая архитектура построения вычислительных комплексов **36**

2.1.2 Проблематика архитектуры программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека **38**

2.2 Микросервисная архитектура **41**

2.3 Федеративное обучение **46**

2.3.1 Основные особенности **46**

2.3.2 Федеративное обучение и микросервисы **51**

2.4 Представление архитектуры программного комплекса динамического неинвазивного мониторинга показателей жизнедеятельности человека 52

2.4.1 Архитектурная модель 52

2.4.2 Оркестровка 59

2.4.3 Модуль контроля дрейфа данных 66

2.4.4 Модуль контроля ресурсов 73

2.5 Выводы по главе 77

ГЛАВА 3. МЕТОДИКИ НЕИНВАЗИВНОГО ОПРЕДЕЛЕНИЯ ПОКАЗАТЕЛЕЙ ЖИЗНЕДЕЯТЕЛЬНОСТИ ЧЕЛОВЕКА ПО ДАННЫМ ВИЗУАЛЬНЫХ, КИНЕМАТИЧЕСКИХ И АКУСТИЧЕСКИХ СЕНСОРОВ 79

3.1 Программно-алгоритмическая реализация методик анализа видеоданных для определения показателей жизнедеятельности 82

3.1.1 Определение пульса 82

3.1.2 Определение тремора головы 88

3.2 Программно-алгоритмическая реализация методик анализа кинематических данных для определения показателей жизнедеятельности 96

3.2.1 Определение положения тела 96

3.2.2 Определение мелкой моторики 99

3.3 Программно-алгоритмическая реализация методик анализа аудиоданных для определения показателей жизнедеятельности 107

3.3.1 Определение кашля и чихания 107

3.3.2 Определение эмоций 116

3.4 Необходимость использования федеративного обучения 125

3.5 Выводы по главе 128

ГЛАВА 4. ПРОВЕДЕНИЕ ИСПЫТАНИЙ ПРОГРАММНОГО КОМПЛЕКСА 131

4.1 Экспериментальная установка и методика проведения испытаний 131

4.1.1 Экспериментальная установка и используемое оборудование . 131

4.1.2 Проверяемые показатели 133

4.2 Результаты проведения испытаний 137

4.2.1 Проверка качества работы детектора дрейфа данных модели машинного обучения определения кашля и чихания 137

4.2.2 Проверка качества работы детектора дрейфа данных модели машинного обучения определения эмоций 141

4.2.3 Изменение качества модели машинного обучения определения кашля и чихания 144

4.2.4 Изменение качества модели машинного обучения определения эмоций 147

4.2.5 Изменение потребления аккумулятора мобильного устройства при работающем приложении программного комплекса при использовании модуля контроля ресурсов 152

4.2.6 Проверка производительности и масштабируемости серверного узла при выполнении нагрузочных испытаний 153

4.3 Анализ результатов 156

4.4 Выводы по главе 157

ЗАКЛЮЧЕНИЕ 159

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 161

ПРИЛОЖЕНИЕ А 181

ПРИЛОЖЕНИЕ Б 183

ПРИЛОЖЕНИЕ В 184

ВВЕДЕНИЕ

Актуальность исследования

Прошедшая пандемия вируса COVID-19, которая обладала высокой степенью распространения и, соответственно, высокой степенью заразности, бросила вызов системам здравоохранения почти всего мира. В целом, количество зараженных сопоставимо с распространением обычной ОРВИ. Однако опасность возникновения таких пандемий отнюдь не в общем количестве больных, а в количестве заболевших в короткий промежуток времени, что показали волны распространения [1] COVID-19. Именно в случае лавинообразного роста количества больных система здравоохранения начинает испытывать большие сложности в работе, начиная от нехватки койко-мест и лекарственных препаратов, заканчивая нехваткой медицинских специалистов [2]. Понимая, что в конечном счете переболеет абсолютное большинство, одной из тактик преодоления такого рода пандемий является увеличение временного интервала роста числа заболевших для предполагаемого количества больных, тем самым не перегружая систему. Стоит добавить, что существует риск возникновения новой пандемии, и поэтому человечеству следует извлечь уроки из пройденного. При этом цифровые решения должны учитывать индивидуальные особенности человека, где это возможно.

Все эти аспекты важны для мониторинга состояния здоровья людей, в частности, на рабочих местах, где важно вовремя обнаружить предпосылки к возникновению различных отклонений и расстройств. Можно резюмировать, что существует необходимость в разработке технических систем мониторинга состояния здоровья человека, и, следовательно, методик их определения, которая может привести к внедрению концепции цифровой диагностики, позволяющая снизить нагрузку на медицинских специалистов и сохранить человеческие жизни.

Техническая проблема состоит в обеспечении эффективного функционирования распределённого программного комплекса неинвазивного

мониторинга показателей жизнедеятельности человека в реальных условиях эксплуатации. Источник неэффективности – противоречие между требованиями к качеству формируемого информационного статуса контролируемых показателей при непрерывном мониторинге и ресурсными ограничениями мобильных вычислительных узлов. Оно проявляется в трёх взаимосвязанных аспектах. Во-первых, точность моделей машинного обучения деградирует со временем вследствие дрейфа входных данных, обусловленного сменой пользователей, условий записи и характеристик устройств, что требует поддержания актуальности моделей без обращения к исходным данным. Во-вторых, непрерывная обработка акустических, кинематических и визуальных потоков данных моделями машинного обучения приводит к нерациональному расходованию заряда аккумулятора, загрузки процессора и оперативной памяти мобильного устройства, ограничивая практическую применимость комплекса. В-третьих, персональный характер собираемых данных исключает прямое централизованное переобучение моделей, что требует распределённой схемы обновления. Известные решения либо не обеспечивают совместного учёта перечисленных аспектов, либо требуют передачи исходных данных на серверную сторону, что нарушает конфиденциальность.

Выбор совокупности предложенных решений обусловлен необходимостью устранения указанных источников неэффективности. Применение федеративного обучения позволяет поддерживать актуальность моделей машинного обучения без передачи исходных данных на серверную сторону, что сохраняет конфиденциальность персональных физиологических показателей и одновременно использует обучающий опыт всех мобильных устройств, участвующих в мониторинге. Применение методики обнаружения дрейфа данных с верификацией качества моделей обеспечивает своевременное выявление деградации моделей и инициацию их распределённого обновления. Применение алгоритма адаптивного управления режимами работы сенсорных и вычислительных модулей

позволяет согласовать качество формируемого информационного статуса контролируемых показателей с ресурсными ограничениями мобильного вычислительного узла, продлевая время автономной работы. Совместно перечисленные решения образуют единый контур самоадаптации: обнаружение отклонений качества инициирует федеративное обновление моделей, а управление ресурсами поддерживает непрерывность мониторинга вплоть до момента обновления, что обеспечивает эффективное функционирование комплекса в реальных условиях эксплуатации.

Степень изученности темы

В проанализированной совокупности научных работ выделяются следующие направления. Первое направление – телемедицинский мониторинг и мобильная оценка показателей состояния человека: результаты Федык Л.А., Чандрасекарана В. и платформенные решения коллектива Маврогиоргу А. продемонстрировали техническую достижимость и социально-экономический потенциал дистанционного наблюдения средствами мобильных устройств. Второе направление – неинвазивное определение физиологических показателей по гетерогенным сигналам смартфона: группа Хименеса Д., Парнанди А. и Гутьерреса-Осуны С. показала возможность извлечения частоты сердечных сокращений и частоты дыхания по фотоплетизмографическому и кинематическому сигналу с ошибкой в пределах долей процента. Третье направление – микросервисная архитектура программных комплексов: Клоков В.Н. и систематизация промышленных практик Ньюмана С. обосновали преимущества микросервисного декомпозирования крупных систем. Четвёртое направление – адаптация моделей машинного обучения к изменяющемуся распределению входных данных: исходная постановка проблемы дрейфа концепций принадлежит Видмеру Г. и Кубату М., а систематизация методов обнаружения дрейфа [3–10] выполнена в работе Гамы Ж. и соавторов; развитие федеративного обучения моделей опирается на протокол FedAvg МакМахена Б. с соавторами

и его применение в исследованиях Ляпунцовой Е.В. Пятое направление – рациональное использование ресурсов мобильных вычислительных устройств: Алшурафа Н. предложил методы управления энергопотреблением смартфона в процессе мониторинга. Показанные направления развиваются преимущественно обособленно, что обосновывает постановку задачи объединения соответствующих механизмов в едином контуре самоадаптации программного комплекса [11–14].

В рамках данного исследования был проведен анализ научной литературы, посвященной методологии оценки показателей жизнедеятельности человека, архитектуре крупных программно-аппаратных комплексов, телемедицинским технологиям и применению алгоритмов машинного обучения: Федык Л.А., обосновавшей высокую актуальность и потенциал телемедицины [3], Чандрасекарана В., разработавшего методику сбора базовых медицинских метрик с использованием смартфона [4], Клокова В.Н., который показал преимущества микросервисного подхода при разработке программных комплексов [8], Ляпунцовой Е.В., которая применила федеративное обучение моделей для решения задачи совершенствования качества модели машинного обучения обнаружения аномалий сетевого трафика [13], Маврогиоргу А., предложившего платформу обработки медицинских данных [5] beHEALTHIER, основанную на использовании микросервисной архитектуры, Альшурафа Н., предложившего метод оптимизации потребления заряда смартфона путем отслеживания его состояния [14]

Проведённый анализ показывает, что рассматриваемые направления разрабатываются в основном независимо: результаты в области телемедицины и оценки показателей жизнедеятельности человека, микросервисной архитектуры программных комплексов, федеративного обучения моделей и энергопотребления мобильных устройств получены [3-5, 8, 13, 15-18] разными исследовательскими коллективами и не образуют единого подхода. Критический анализ этих работ позволяет выделить ограничения, не

преодолённые в известных решениях: отсутствие формальной модели самоадаптации программного комплекса к изменению распределения входных данных; отсутствие механизмов верификации качества моделей машинного обучения в реальной эксплуатации; отсутствие совместного учёта дрейфа данных и ресурсного состояния мобильного устройства при принятии управляющих решений. Указанные ограничения определяют актуальность разработки самоадаптирующегося программного комплекса, сочетающего обнаружение дрейфа данных, федеративное обновление моделей и адаптивное управление ресурсами в едином контуре.

Объектом исследования являются программные комплексы неинвазивного мониторинга показателей жизнедеятельности человека, функционирующие в условиях распределенной обработки данных, неоднородности пользовательских устройств и ресурсных ограничений мобильных вычислительных узлов.

Предметом исследования являются модели, методы и алгоритмы построения, адаптации, верификации и экспериментальной оценки самоадаптирующейся микросервисной архитектуры программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека.

Цель работы – Повышение эффективности функционирования программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека за счет разработки самоадаптирующейся микросервисной архитектуры, механизмов обнаружения дрейфа данных и федеративного обучения моделей машинного обучения. [1]

Задачи:

1. Разработать архитектурную модель самоадаптирующегося программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека на основе микросервисного взаимодействия и

[1] Под эффективностью в работе понимается способность программного комплекса поддерживать качество моделей при изменении распределения входных данных, инициировать адаптацию без централизованного сбора исходных пользовательских данных и рационально использовать ресурсы мобильного вычислительного устройства.

персонализированного федеративного обучения моделей машинного обучения.

2. Разработать алгоритм многопараметрической обработки гетерогенных сенсорных данных мобильного устройства, включающих акустические, кинематические и визуальные данные, для формирования информационного статуса контролируемых показателей жизнедеятельности человека.

3. Разработать методику обнаружения дрейфа данных и верификации моделей машинного обучения в составе распределенного программного комплекса с последующим инициированием процедуры федеративного дообучения.

4. Разработать алгоритм адаптивного управления режимами работы сенсорных и вычислительных компонентов мобильного приложения, направленный на рациональное использование ресурсов мобильного устройства при сохранении качества определения контролируемых показателей.

5. Провести экспериментальную оценку эффективности разработанных архитектурных, алгоритмических и программных решений по сравнению с базовой конфигурацией комплекса без механизмов самоадаптации, федеративного дообучения и адаптивного управления режимами сенсоров.

Функционирование разрабатываемого комплекса рассматривается как управляемый процесс, связывающий поставленные задачи единой причинно-следственной цепочкой: событие мониторинга – измеряемый критерий – управляющее воздействие – изменение конфигурации – ожидаемый эффект. Так, обнаруженный дрейф данных и состояние ресурсов мобильного устройства выступают измеряемыми критериями, инициирующими управляющие воздействия в виде федеративного дообучения моделей и адаптивного изменения режимов работы компонентов, что приводит к изменению конфигурации комплекса и ожидаемому эффекту повышения

качества моделей машинного обучения и рационального использования ресурсов мобильного устройства.

Методы исследования. Для решения поставленных в работе задач использовались методы теоретического моделирования, системного моделирования, алгоритмической разработки, машинного обучения, цифровой обработки сигналов.

Научная новизна

1. Научная новизна разработанной архитектурной модели самоадаптирующегося программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека заключается в интеграции микросервисного взаимодействия, механизма обнаружения дрейфа данных и персонализированного федеративного обучения моделей машинного обучения. В отличие от монолитных и статически конфигурируемых решений, предложенная архитектурная модель обеспечивает изменение параметров работы программных компонентов и обновление моделей машинного обучения без централизации исходных пользовательских данных. Предложенная архитектурная модель задает формальный контур самоадаптации «событие мониторинга – измеряемый критерий – управляющее воздействие – изменение конфигурации – ожидаемый эффект», связывающий наблюдаемые критерии дрейфа данных и ресурсного состояния с управляющими воздействиями по переконфигурации комплекса, что отличает её от статической интеграции известных технологий.

2. Научная новизна разработанного алгоритма многопараметрической обработки гетерогенных сенсорных данных мобильного устройства заключается в совместном использовании акустической, кинематической и визуальной сенсорики для формирования информационного статуса контролируемых показателей жизнедеятельности человека. В отличие от решений, ориентированных на обработку одного типа сигнала или ограниченного набора показателей, предложенный алгоритм

обеспечивает анализ расширенного набора показателей на базе пользовательского мобильного устройства.

3. Научная новизна разработанной методики обнаружения дрейфа данных и верификации моделей машинного обучения в составе программного комплекса заключается в совместном использовании статистических критериев, оценки расхождения распределений и метрик качества моделей машинного обучения для принятия решения о необходимости дообучения. В отличие от автономных процедур контроля качества моделей, предложенная методика встроена в контур функционирования программного комплекса и инициирует федеративное обновление моделей при выявлении дрейфа данных.

4. Научная новизна разработанного алгоритма адаптивного управления режимами работы сенсорных и вычислительных компонентов мобильного приложения заключается в учете вектора состояния мобильного устройства и требований к качеству определения контролируемых показателей. В отличие от простого контроля уровня заряда аккумулятора, предложенный алгоритм изменяет режимы работы программных компонентов, частоту обращения к сенсорам и параметры обработки данных с учетом компромисса между энергопотреблением и качеством анализа.

Теоретическая значимость диссертационного исследования заключается в развитии моделей и методов построения самоадаптирующихся программных комплексов, функционирующих в условиях гетерогенных сенсорных данных, дрейфа распределения входных признаков и ограниченных ресурсов мобильных вычислительных устройств. Предложенные архитектурные, алгоритмические и методические решения развивают подходы к проектированию программных систем с распределенной обработкой данных, встроенной верификацией моделей машинного обучения и адаптацией параметров функционирования компонентов программного комплекса.

Практическая значимость работы заключается в разработке программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека, реализующего многопараметрическую обработку акустических, кинематических и визуальных данных мобильного устройства, обнаружение дрейфа данных, федеративное дообучение моделей машинного обучения и адаптивное управление режимами работы мобильного приложения.

Разработанная архитектурная модель может быть использована как основа для создания масштабируемых программных комплексов мониторинга функционального состояния человека в задачах безопасности жизнедеятельности, производственного контроля и удаленного наблюдения за контролируемыми показателями.

Разработанная методика обнаружения дрейфа данных и федеративного обновления моделей позволяет поддерживать актуальность моделей машинного обучения без передачи исходных пользовательских данных на серверный узел. Разработанный алгоритм управления режимами работы компонентов мобильного приложения позволяет увеличить время работы мобильного устройства при ограниченном снижении качества анализа сенсорных данных.

Научные положения, выносимые на защиту

На защиту выносятся следующие научные положения, выдвигаемые на основе полученных в диссертационной работе результатов:

1) Архитектурная модель самоадаптирующегося программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека, основанная на микросервисном взаимодействии компонентов серверной части, модульной организации клиентской части, обнаружении дрейфа данных и федеративном обновлении моделей машинного обучения. (соотв. п. 3 специальности 2.3.5).

2) Алгоритм многопараметрической обработки акустических, кинематических и визуальных сенсорных данных мобильного устройства для

формирования информационного статуса контролируемых показателей жизнедеятельности человека. (соотв. п. 7 специальности 2.3.5).

3) Методика обнаружения дрейфа данных и верификации качества моделей машинного обучения в составе распределенного программного комплекса, использующая статистические критерии расхождения распределений и метрики качества моделей и обеспечивающая принятие решения о запуске процедуры федеративного дообучения. (соотв. п. 4 специальности 2.3.5).

4) Алгоритм адаптивного управления режимами работы сенсорных и вычислительных модулей мобильного приложения, обеспечивающий рациональное использование ресурсов мобильного устройства при ограниченном снижении качества анализа. (соотв. п. 1 специальности 2.3.5).

5) Методика экспериментальной оценки эффективности самоадаптации программного комплекса на основе интегрального критерия, применение которой подтверждает повышение устойчивости к дрейфу данных и качества моделей машинного обучения, а также увеличение времени работы мобильного устройства по сравнению с базовой конфигурацией комплекса без механизмов самоадаптации. (соотв п.п. 10).

Апробация работы

Основные положения и результаты диссертационного исследования были изложены и обсуждены на итоговой научно-практической конференции профессорско-преподавательского состава Института физики и Химического института имени А.М. Бутлерова Казанского федерального университета (г. Казань, 2023); Всероссийской научно-технической конференции с международным участием имени профессора О.Н. Пьявченко «Компьютерные и информационные технологии в науке, инженерии и управлении» (г. Таганрог, 2023–2024); Одиннадцатой всероссийской научно-практической конференции по имитационному моделированию и его применению в науке и промышленности (г. Казань, 2023); 34-й международной научно-технической конференции «Экстремальная

робототехника» (г. Санкт-Петербург, 2023); Международной молодежной научной конференции «Технологии ИИ в науке и образовании», МГУ им. М. В. Ломоносова (г. Москва, 2023).

Публикации

Основные научные и практические результаты работы представлены в 4 статьях в изданиях из списка ВАК по специальности, результаты интеллектуальной деятельности защищены 1 свидетельством о государственной регистрации программы для ЭВМ. Имеется единоличная публикация. Личный вклад соискателя в работах, выполненных в соавторстве, состоит в разработке архитектуры программного комплекса, реализации алгоритмов обработки сенсорных данных и методов машинного обучения, а также в проведении экспериментальных исследований и анализе их результатов.

ГЛАВА 1. АНАЛИЗ ПРОГРАММНО-АППАРАТНЫХ КОМПЛЕКСОВ НЕИНВАЗИВНОГО МОНИТОРИНГА ПОКАЗАТЕЛЕЙ ЖИЗНЕДЕЯТЕЛЬНОСТИ ЧЕЛОВЕКА

1.1 Портативные программно-аппаратные комплексы неинвазивного мониторинга показателей жизнедеятельности человека

Программно-аппаратный комплекс неинвазивного мониторинга показателей здоровья человека – автоматизированное решение, которое выполняет анализ биометрических данных, получаемых с некоторого набора сенсорики, формируя предварительную оценку определяемых комплексом параметров жизнедеятельности человека. Сегодня в основе работы таких комплексов лежит алгоритмическая обработка данных с использованием теории обработки сигналов в синтезе с базой медицинских знаний, позволяющей выявлять возможные отклонения по анализируемым параметрам жизнедеятельности от норм и, при необходимости, формировать предварительные рекомендации.

Одной из немаловажных задач, решаемых такого рода комплексами, является повышение медицинской грамотности пользователей и разгрузка системы здравоохранения путем отсеивания неэкстренных случаев. Стоит отметить, что такие комплексы не являются полной заменой медицинской диагностики, проводимой профессиональными специалистами в области медицины, так как существует ряд существенных ограничений. К таким ограничениям можно отнести как точность определения показателей алгоритмами, индивидуальные особенности людей, которые на момент обследования не могли быть учтены алгоритмами, так и отсутствие физического осмотра. Используемые в работе термины «шок», «гипоксия», «дыхательная недостаточность», «гипертензия» и подобные им обозначают предварительные информационные статусы и индикаторы риска, формируемые комплексом по данным мониторинга, и не являются медицинскими диагнозами.

1.1.1 Классификация комплексов неинвазивного мониторинга состояния здоровья человека

Сегодня такие программно-аппаратные комплексы по способу реализации можно классифицировать как:

- Решения, основанные на использовании мобильных устройств связи [15];
- Располагаемые непосредственно на теле человека самостоятельные носимые изделия;
- Комбинированные решения, которые сочетают в себе, как и использование мобильных устройств, так и аппаратных модулей;

Такие комплексы проводят обработку сигналов, которые могут иметь разную природу происхождения. В связи с этим их принято классифицировать по типу:

- вибрационные;
- визуальные;
- тепловые;
- газодинамические;
- акустические.

По уровню автоматизации:

- экспертные;
- автоматические;
- смешанные.

По варианту размещения диагностического оборудования:

- бортовые:
  - встроенные;
  - встраиваемые;
- стендовые:
  - стационарные;
  - переносимые;
- комбинированные.

По способу технической реализации:

- программные;
- аппаратные;
- программно-аппаратные.

По широте решаемых задач:

- комплексные;
- узкоспециализированные.

Среди разнообразных классов систем предварительной оценки состояния здоровья особую значимость имеют портативные решения. Их ключевое преимущество – возможность длительного непрерывного мониторинга биометрических показателей, что минимизирует риск пропуска критических изменений состояния. В отличие от стационарных комплексов, фиксирующих данные дискретно в заданных локациях, мобильные системы обеспечивают контроль параметров в условиях естественной жизнедеятельности пользователя, где особое внимание стоит уделить рабочему месту пользователя. Деятельность работника может быть сопряжена с рисками, причем несвоевременно обнаруженные проблемы со здоровьем могут стать причиной большого очага заражения заболеваниями или в худшем случае стать причиной летального исхода как самого работника, так и его коллег.

1.1.2 Технологии, применяемые в программно-аппаратных комплексах неинвазивного мониторинга показателей жизнедеятельности человека

Современные мониторинговые комплексы должны обладать принципиально бинарной операционной адаптивностью, обеспечивая функциональную полноту как в условиях клинической инфраструктуры (стационары, диагностические центры), так и во внеучрежденческой среде (домашний мониторинг, удалённые территории). Эта концепция предполагает не просто интеграцию с медицинскими информационными системами, но и реализацию алгоритмов скрининговой оценки в децентрализованных

условиях, что детерминирует применение дистанционных диагностических технологий. Подобная дуальная операционная модель направлена на обеспечение сквозной диагностической преемственности на всех этапах оказания медицинской помощи.

В целом, основными направлениями современных систем диагностики состояния здоровья человека являются:

1. Внедрение в медицинскую инфраструктуру:

- адаптация решений для использования в лечебно-диагностических учреждениях (стационары, поликлиники, лаборатории);
- разработка систем для индивидуального применения (персональные устройства мониторинга);
- внедрение в корпоративные программы здоровья (профилактика на рабочих местах).

2. Технологические и методологические подходы:

- телемедицинские платформы для дистанционной диагностики и консультирования;
- искусственный интеллект (ИИ), включая машинное обучение для анализа медицинских данных и прогностической аналитики;
- биомаркеры и геномные исследования (например, секвенирование ДНК, эпигенетические тесты);
- микробиомный анализ (изучение микрофлоры организма для диагностики дисбиозов и связанных патологий);
- спектроскопические методы (оптическая, инфракрасная, рамановская спектроскопия) для неинвазивного исследования тканей;
- электроимпедансная томография (визуализация распределения электропроводности в тканях).

В условиях активной цифровой трансформации здравоохранения системы телемедицины, реализуемые средствами вычислительной техники, приобретают ключевое значение [19]. Функционал таких комплексов заключается в организации удаленного наблюдения за больными и

предоставлении консультационных услуг, что позволяет осуществлять первичный мониторинг здоровья и проводить предварительное диагностическое обследование. Технологической основой для реализации подобных комплексов служит специализированное программное обеспечение, в том числе платформы для видеосвязи, мобильные приложения, электронные системы ведения истории болезни и прочие элементы информационной инфраструктуры. Инженерно-техническая разработка телемедицинских комплексов нацелена на сокращение временных расходов при оказании медицинской помощи и уменьшение стоимости услуг благодаря автоматизации рабочих процессов и организации удаленного контакта между участниками системы. Однако, с точки зрения разработки и возможностей ПО, важно учитывать, что телемедицина обладает рядом технологических ограничений и не способна полностью заменить очные визиты к врачам.

Исследование подтверждает необходимость внедрения телемедицинских платформ для организации консультативной поддержки в периоды эпидемий. Авторы также подчеркивают, что подобные системы расширяют доступ к высококвалифицированным специалистам, чье физическое присутствие может быть затруднено [3].

С точки зрения проектирования ПО, преимущества телемедицинских систем включают: [20]

- расширение географической доступности медицинских услуг;
- оптимизацию временных ресурсов пациентов и персонала;
- обеспечение комфортного взаимодействия с пациентом в удаленных условиях;
- снижение рисков перекрестного инфицирования за счет минимизации контактов.

Таким образом, современные телемедицинские комплексы предоставляют технологическую основу для эффективного дистанционного мониторинга состояния здоровья на доврачебном этапе.

Алгоритмы машинного обучения (МО) формируют более современный инструментарий в сфере диагностики состояния человека, предоставляя вычислительные методики для анализа сложных биомедицинских данных [21]. Разрабатываемые программные комплексы на основе МО предназначены для обработки обширных гетерогенных массивов информации, включающих сигналы сенсоров, электронные истории болезни и результаты инструментальных обследований, с целью генерации структурированной оценки физиологических параметров. В контексте проектирования программных решений для медицинской визуализации:

- в работе [19] демонстрируется реализация МО-модели, решающей задачу автоматизированного обнаружения патологий предстательной железы на медицинских снимках;
- исследователи [22] применяют методы машинного обучения как технологический этап предварительной обработки для оптимизации качества рентгенографических изображений перед дальнейшим анализом;
- авторы [23] используют архитектуры глубокого обучения (Deep Learning) для разработки программных модулей, интерпретирующих сигналы фотоплетизмограммы (PPG) с целью неинвазивной оценки показателей артериального давления;
- авторы [24] предлагают алгоритмический подход, основанный на сверточных нейронных сетях (CNN), для автоматизации анализа видеопотока в капсульной эндоскопии.

С позиции проектирования информационных систем, ключевым приоритетом является разработка программно-аппаратного комплекса, ориентированного на два сценария развертывания:

- персональное использование конечными потребителями;
- интеграция в корпоративные информационные среды предприятий, включая объекты с особыми требованиями к безопасности (в первую очередь, опасные производства с повышенными экологическими и профессиональными рисками).

Актуальность внедрения такого комплекса на производственных площадках обусловлена его архитектурой, обеспечивающей:

- беспрерывный сбор и агрегацию физиологических показателей сотрудников через интегрированные аппаратные сенсоры;
- автоматизированный анализ поступающих данных в режиме реального времени;
- формирование предупредительных уведомлений при обнаружении статистически значимых отклонений от установленных биометрических норм.

Данный функционал, реализуемый ядром программно-аппаратного комплекса, направлен на:

- раннюю идентификацию потенциальных проблем со здоровьем;
- инициацию процедур углубленной инструментальной диагностики в медицинских учреждениях;
- реализацию превентивных мер по сокращению профпатологий, минимизации риска острых состояний и снижению финансовых рисков, связанных с операционными простоями из-за временной нетрудоспособности персонала.

Важным техническим ограничением, учитываемым при разработке пользовательских интерфейсов и алгоритмов обработки, является:

- исключительная зависимость от неинвазивных методик сбора данных;
- трактовка результатов мониторинга исключительно как скрининговых данных. Комплекс формирует индикаторы потенциальных аномалий, требующих верификации с помощью специализированного клинического оборудования медицинскими работниками. Архитектура ПО должна четко определять границы ответственности системы и медицинского персонала.

С точки зрения проектирования и реализации программно-аппаратного комплекса для скрининга физиологических показателей, ключевые архитектурные и функциональные требования включают:

- обязательна реализация безопасного централизованного хранилища данных, соответствующего концепциям телемедицины. Доступ к данным должен регулироваться механизмами аутентификации и ролевого контроля доступа, предоставляя авторизованным медицинским специалистам удаленный API-доступ для мониторинга и анализа;
- программный клиент должен обеспечивать полностью самостоятельное проведение измерений пользователем через интуитивно понятный UX/UI дизайн с минимальной интерактивностью, включая автоматическую калибровку и валидацию данных;
- архитектура комплекса должна исключительно опираться на ПО для обработки сигналов, получаемых от неинвазивной сенсорики. Алгоритмы предварительной обработки и специализированные вычислительные модели (например, для оценки параметров по косвенным сигналам) являются критическим компонентом ядра системы;
- шифрование и соответствие профильным стандартам являются обязательными нефункциональными требованиями.

1.1.3 Определяемые программно-аппаратным комплексом параметры неинвазивного мониторинга

Динамика физиологических процессов организма человека сопровождается детектируемыми изменениями показателей. Для портативных диагностических комплексов критически важна неинвазивная фиксация базовых параметров жизнедеятельности. В качестве основных можно выделить следующие классы показателей, которые можно определить неинвазивным путем:

- базовые показатели жизненно важных функций;
- рефлекторные;
- неврологические показатели;

- травматологические воздействия.

К базовым параметрам относится частота сердечных сокращений, которая является одним из самых простых и важных показателей состояния здоровья человека, так как проблемы с сердцем могут повлечь за собой фатальные последствия для пациента. Данный показатель позволяет определить такие отклонения, как брадикардию и тахикардию, которые в определенных обстоятельствах могут свидетельствовать о предобморочном состоянии, возможной аритмии, которая может повлечь за собой такие осложнения, как инфаркт миокарда, острую сердечную недостаточность, ишемический инсульт. Поэтому мониторинг такого параметра является важным, чтобы вовремя обратиться к специалисту, для предотвращения или минимизации негативных последствий. Другим базовым параметром является насыщение крови кислородом (сатурация, оксигенация) – процент насыщенного кислородом гемоглобина в крови пациента относительно общего гемоглобина в крови. Нормальным показателем оксигенации взрослого человека являются значения в диапазоне 95%-100%. Значения ниже свидетельствуют о различных степенях дыхательной недостаточности, вплоть до гипоксической комы. Артериальное давление также является одним из самых важных параметров для человека, показывает давление крови на стенки кровеносных сосудов. Оно определяется объемом крови, перекачиваемым сердцем в единицу времени и сопротивлением сосудистого русла, поскольку кровь движется под влиянием градиента давления в сосудах, которое создает сердце. Другим базовым параметром является частота дыхания – количество дыхательных движений, которые совершает человек за определённую единицу времени (обычно за минуту). Этот параметр может быть использован для оценки общего состояния организма, так как изменение частоты дыхания может указывать на наличие различных отклонений или состояний, таких как стресс, усталость, отклонения дыхательной системы и т.д. Вместе с частотой также определяется поверхностность (поверхность) дыхания – характеристика дыхания, которая отражает степень участия верхних дыхательных путей (носа

и гортани) в процессе дыхания. Если дыхание поверхностное, то воздух преимущественно проходит через верхние дыхательные пути, а нижние (бронхи и легкие) участвуют в дыхании меньше. Поверхностное дыхание может быть связано с различными отклонениями дыхательной системы, такими как бронхиальная астма, хроническая обструктивная болезнь легких (ХОБЛ) и др.

В перечень ключевых неврологических маркеров входят нарушения мелкой моторики, проявляющиеся дисфункцией координации движений пальцев кисти. Данный дефицит возникает вследствие неврологических патологий (инсульт, нейродегенеративные отклонения), травматических поражений ЦНС или иных этиологических факторов. Количественная оценка моторного дефицита осуществляется посредством стандартизированных проб, фиксирующих точность и скорость выполнения мелкоамплитудных движений. Параллельно значимым индикатором выступает тремор – непроизвольные осцилляции конечностей с частотным диапазоном 3–15 Гц [25]. Его генез связан с дисбалансом экстрапирамидной системы, психоэмоциональным напряжением или органической патологией. Объективизация тремора достигается методами электромиографии, регистрирующими биоэлектрическую активность мышечных волокон.

Травматологически значимые события включают высокоэнергетические механические воздействия (удары) и неконтролируемые изменения позы (падения). Если первые обусловлены экзогенными факторами, то вторые нередко связаны с синкопальными (обморочными) состояниями. Мониторинг данных параметров критически важен на производствах с повышенным риском травматизма.

К защитно-рефлекторным феноменам относятся:

- кашель – реакция на раздражение респираторного эпителия, выступающая маркером респираторных патологий (ОРВИ, пневмония, бронхообструктивные состояния). Характеризуется

продуктивностью/непродуктивностью и длительностью. Физиологическая роль – эвакуация патогенов и секрета;

- чихание – непроизвольный форсированный выдох при раздражении назальной слизистой, обеспечивающий элиминацию аэрозольных агентов. Триггерами выступают аллергены, инфекции.

Комбинации параметров позволяют стать основой для выявления следующих отклонений: бронхиальная астма (при наличии сухого кашля, высокой частоты дыхания, поверхностного дыхания), хроническая обструктивная болезнь легких (при наличии кашля, частота которого увеличивается с течением времени, высокой частоты дыхания, наличия поверхностного дыхания), острая респираторная вирусная инфекция (при наличии частого сухого кашля, чихания), паническая атака (при снижении уровня оксигенации, увеличении частоты дыхания, наличии психоэмоционального стресса), болезнь Паркинсона (при наличии тремора рук, тремора головы), сердечная недостаточность (при наличии высокой частоты дыхания, высокой частоты сердечных сокращений, сухого кашля). Таким образом, разрабатываемый программный комплекс было принято решение ориентировать на отслеживание следующих ключевых параметров с использованием кинематических, акустических и визуальных сенсоров:

1. Обнаружение кашля и типа кашля.
2. Обнаружение чихания.
3. Определение частоты дыхания.
4. Определение уровня оксигенации.
5. Определение поверхностности дыхания.
6. Определение частоты сердечных сокращений.
7. Определение систолического и диастолического давлений (артериальное давление).
8. Определение состояния шока.
9. Определение наличия психоэмоционального стресса.
10. Определение позы человека.

11. Определение сильных ударов.
12. Определение падений.
13. Определение степени мускульной усталости.
14. Определение наличия тремора рук.
15. Определение наличия тремора головы.
16. Определение наличия дефектов мелкой моторики.
17. Определение дефектов моторики, связанных с потерей общей координации движений.

Заявленный перечень из 17 контролируемых показателей жизнедеятельности имеет различную степень алгоритмической и экспериментальной поддержки, что обусловлено приоритизацией задач и ограничениями неинвазивного мониторинга. По результатам выполненной работы показатели разделены на две группы. К первой группе относятся реализованные и прошедшие экспериментальную верификацию показатели: определение кашля и его типа, определение чихания, определение психоэмоционального стресса (модели распознавания эмоций). Ко второй группе относятся показатели, интегрированные в программный комплекс с полноценной алгоритмической реализацией и ограниченной экспериментальной оценкой: определение частоты сердечных сокращений (средняя ошибка измерений в пределах 2%) и не рассматриваемые глубоко в тексте диссертации.

1.1.4 Сравнение реализаций программно-аппаратных комплексов неинвазивной диагностики показателей жизнедеятельности человека

На глобальном рынке медицинских технологий наблюдается рост предложений комплексов анализа показателей жизнедеятельности. Ведущие зарубежные производители предлагают гибридные программно-аппаратные платформы, ориентированные на самостоятельный мониторинг показателей здоровья. Для проведения анализа был рассмотрен ряд подходов[20, 26–28], сравнение которых представлено в таблице 1.1. Для уточнения части

характеристик некоторых решений были написаны обращения к создателям с целью уточнения данных.

Таблица 1.1 – Сравнение реализаций программно-аппаратных комплексов неинвазивной диагностики показателей жизнедеятельности человека

Название	СберЗдоровье	Ornament	Монитор сердечного ритма	VisualDX
Архитектура	Многомодульная чистая архитектура[29]	Монолитная/Трехуровневая [30]	Монолитная/Трехуровневая	Не подтверждено производителем
Тип исполнения	Мобильное ПО	Мобильное ПО	Мобильное ПО	Мобильное ПО
Кол-во измеряемых параметров / Анализируемые данные	9 (уровень стресса, вариабельность сердечного ритма, уровень усталости, средний уровень глюкозы, риск нарушения обмена глюкозы, давление, холестерин, жёсткость артериальной стенки, частоту сердечных сокращений)	Все анализы из LabCorp и других систем; 20+	1 (Частота сердечных сокращений)	Производится анализ фото участка кожи с целью выявления возможного заболевания
Используемая сенсорика	Визуальные (камера)	Не используются	Визуальные (камера)	Визуальные (камера)

Продолжение таблицы 1.1

Название	СберЗдоровье	Ornament	Монитор сердечного ритма	VisualDX
Устойчивость к дрейфу данных	Не подтверждено производителем	Отсутствует.	Отсутствует	Данные попадают в базу обучающей выборки для проведения процедуры дообучения
Управление энергопотреблением	Нет	Нет	Нет	Нет

Сравнительный анализ современных программно-аппаратных комплексов неинвазивного мониторинга (СберЗдоровье, Ornament, Монитор сердечного ритма, VisualDX) выявляет системные ограничения, принципиально препятствующие созданию адаптивных систем. Существенно ограничена и функциональная полнота – большинство систем опираются исключительно на визуальную сенсорику и имеют довольно небольшой набор определяемых параметров за исключением Ornament, который не предоставляет функционала прямого определения показателей, а обеспечивает анализ данных путем загрузки и обработки файла анализов, проведенных в профильных клиниках. Персонализация реализована фрагментарно через ручную калибровку или обновления ПО, без механизмов автоматической адаптации моделей машинного обучения к индивидуальным биометрическим паттернам. Наиболее значимый пробел – повсеместное отсутствие устойчивости к дрейфу данных: ни один программный комплекс не содержит инструментов верификации отказоустойчивости или адаптации алгоритмов к долговременным изменениям сигналов. Эти ограничения подтверждают необходимость разработки новых архитектурных подходов реализации программного комплекса, сочетающих комплексность измерений, а также количество определяемых параметров. В результате рассмотренных решений можно сделать вывод о необходимости использования архитектуры,

которая позволит обеспечить самоадаптацию – изменение поведения программного комплекса в ответ на изменения в системе или ее окружении для обеспечения некоторого уровня качества.

1.2 Автоматизированный программно-аппаратный комплекс неинвазивного мониторинга состояния жизнедеятельности человека

Архитектура программно-аппаратного комплекса неинвазивного мониторинга показателей жизнедеятельности человека базируется на автоматизированной обработке ряда показателей жизнедеятельности, что позволяет формировать предварительные диагностические заключения.

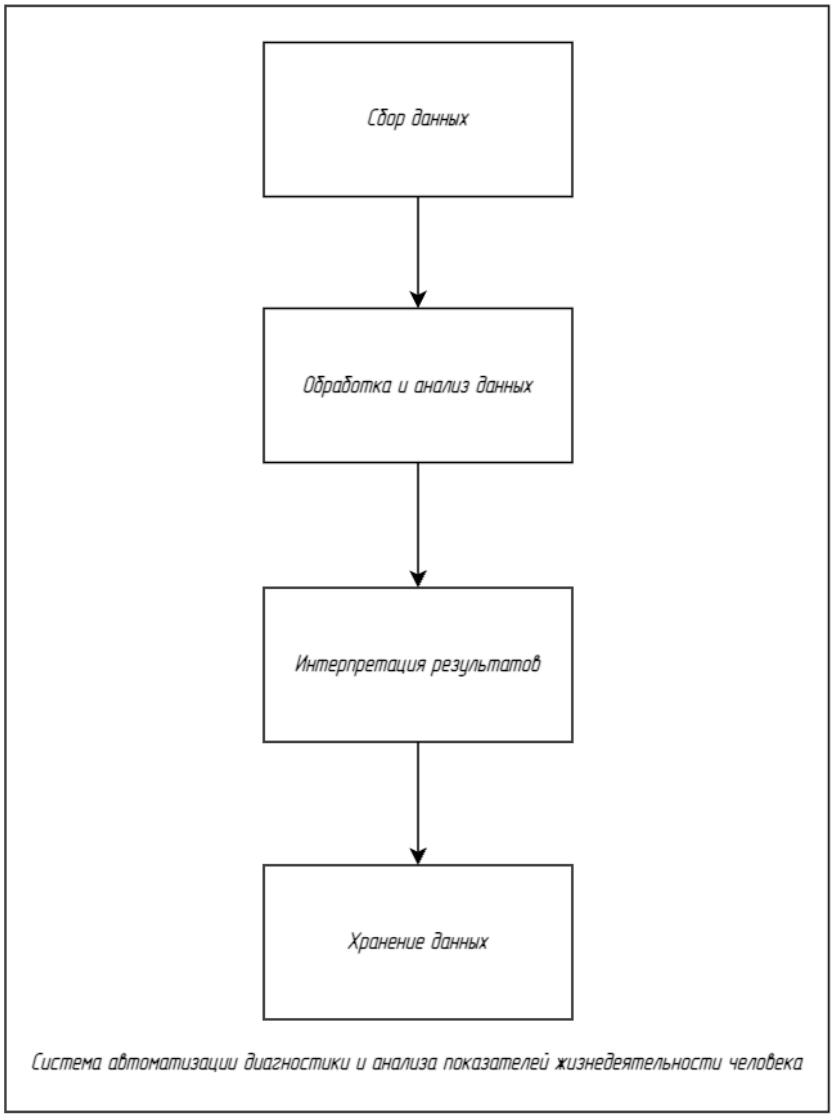


Рисунок 1.1 – Типовая структурная схема комплекса автоматизации комплекса анализа показателей жизнедеятельности

Типовая структура таких комплексов состоит из следующих компонентов (рисунок 1.1):

1. Блок сбора данных осуществляет фиксацию первичных биометрических показателей, используя унифицированные интерфейсы подключения гетерогенной сенсорики, включая: кинематические [31-32], визуальные [33-34], акустические [35-36], гибкие носимые [37] и иные сенсорные устройства [38].

2. Модуль обработки и анализа данных реализует алгоритмы предварительной обработки и исследования необработанных сенсорных показаний, применяя вычислительные методы цифровой обработки сигналов и статистического моделирования.

3. Блок интерпретации результатов трансформирует выходные данные аналитики в предварительные диагностические заключения. Процесс интерпретации может осуществляться автоматически или при участии врача-эксперта, осуществляющего мониторинг состояния пациентов. На основе полученных данных формируется рекомендация о необходимости проведения дополнительных, в том числе инвазивных, обследований.

4. Модуль хранения данных отвечает за сохранение информации о пользователях и результатов мониторинга (оценки информационного статуса контролируемых показателей) в структурированном формате, например, в виде баз данных на серверных платформах.

Согласно исследованию [39] современные программно-аппаратные комплексы должны быть оснащены следующим функционалом:

1. Предоставление медицинских услуг населению в рамках информационной системы для оказания первичной медико-санитарной помощи.

2. Интеграция в единую централизованную платформу, обеспечивающую мониторинг состояния пациентов в реальном времени.

3. Формирование единого реестра учетных данных в распределенной информационной архитектуре.

4. Обеспечение аналитической поддержки для принятия управленческих решений в сфере здравоохранения.

1.3 Инфокоммуникационный функционал

Согласно определению инфокоммуникационных систем, ключевыми функциями таких комплексов являются обработка информации, организация хранения и обеспечение передачи данных. Функционал системы включает следующие аспекты:

- сбор информации посредством сенсоров технических устройств;
- мониторинг физиологических параметров пациента через анализ полученных измерений: определение частоты пульса, артериального давления, температуры тела, уровня насыщения крови кислородом, выявление тремора, кашля, чихания и других показателей;
- оценка поведенческой и двигательной активности путем отслеживания физических нагрузок, позволяющая выявлять недостатки питания, дефицит физической активности, которые могут привести к развитию сопутствующих отклонений, а также регистрировать случаи получения физического вреда;
- получение обратной связи и рекомендаций от аналитических алгоритмов системы и высококвалифицированных специалистов (формирующих вместе экспертную систему) на основе собранных данных измерений пациента;
- хранение информации о пациенте, включая персональные данные, базовые антропометрические показатели, результаты оценки жизненных параметров и заключения экспертной системы;
- обеспечение доступа к данным в рамках единой инфокоммуникационной медицинской сети.

Задача регистрации данных с помощью сенсорных устройств решается путем применения аппаратных технических средств. Сенсоры можно классифицировать на два основных типа:

- генераторного типа – способны под воздействием измеряемого сигнала непосредственно генерировать электрическое напряжение или ток, к данной категории относятся пьезоэлектрические, термоэлектрические, индукционные и фотоэлектрические сенсоры;
- параметрического типа – отличаются тем, что под воздействием измеряемой величины изменяют свои электрические параметры и требуют внешнего источника питания, к этому типу относятся: емкостные, индуктивные, реостатные сенсоры.

В зависимости от энергии, являющейся носителем информации сенсоры можно классифицировать как:

- визуальные;
- температурные;
- акустические;
- механические.

Мониторинг и анализ показателей жизнедеятельности человека реализуется с применением специализированных алгоритмов, осуществляющих трансформацию исходных данных от сенсоров в конкретные показатели жизнедеятельности, а также обеспечивающих последующую интерпретацию результатов врачами-экспертами. Специальные алгоритмы обработки могут включать преобразования на основе физических законов и математических теорем, а также использовать методы машинного обучения [40].

Обеспечение хранения данных реализуется посредством систем управления базами данных (СУБД). Доступ к сохраненной информации осуществляется через формирование запросов к базе данных, как с использованием специализированных протоколов, так и решений, основанных на отраслевых стандартах.

1.4 Выводы по главе

1. Таким образом, в рамках главы проведено исследование архитектурных принципов построения программных комплексов мониторинга физиологического состояния. Проанализированы существующие рыночные решения, обоснована практическая ценность разработки. Основные решения, как правило, основываются на определении небольшого количества параметров, большинство не обладают устойчивостью к дрейфу данных и к тому же не имеют многокритериального подхода к обеспечению экономии заряда аккумулятора. В связи с этим сделан вывод о необходимости разработки и применения такой архитектуры программного комплекса неинвазивного мониторинга показателей здоровья человека, которая позволит обеспечить самоадаптацию – изменение поведения программного комплекса в ответ на изменения в системе или ее окружении для обеспечения некоторого уровня качества.

2. Аналоги программного комплекса в основном имеют небольшое количество определяемых параметров, тогда как комбинации параметров позволяют выдвинуть предположение о наличии недугов и опасных отклонений, в число которых входят: бронхиальная астма (при наличии сухого кашля, высокой частоты дыхания, поверхностного дыхания), хроническая обструктивная болезнь легких (при наличии кашля, частота которого увеличивается с течением времени, высокой частоты дыхания, поверхностного дыхания), острая респираторная вирусная инфекция (при наличии частого сухого кашля, чихания), паническая атака (при снижении уровня оксигенации, увеличении частоты дыхания, наличии психоэмоционального стресса), болезнь Паркинсона (при наличии тремора рук, тремора головы), сердечная недостаточность (при наличии высокой частоты дыхания, высокой частоты сердечных сокращений, сухого кашля). Таким образом, определен набор параметров для определения программным комплексом.

3. Ключевые инженерные требования к комплексу включают: обеспечение портативности (работа на ресурсоограниченных устройствах); поддержку пользовательских смартфонов и носимых сенсоров; интеграцию с телемедицинскими платформами с контролем доступа для медперсонала; реализацию полностью автономного UX для самостоятельного сбора биометрических данных; использование исключительно неинвазивных методов измерения.

4. Были описаны основные компоненты системы автоматизации комплекса анализа показателей жизнедеятельности человека, функционал инфокоммуникационных систем и способы снятия данных для анализа.

5. Таким образом, было принято решение по разработке эффективного комплекса неинвазивного мониторинга показателей жизнедеятельности человека с использованием самоадаптирующейся микросервисной архитектуры и федеративного обучения моделей.

ГЛАВА 2. САМОАДАПТИРУЮЩАЯСЯ МИКРОСЕРВИСНАЯ АРХИТЕКТУРА С ИСПОЛЬЗОВАНИЕМ ПЕРСОНАЛИЗИРОВАННОГО ФЕДЕРАТИВНОГО ОБУЧЕНИЯ МОДЕЛЕЙ

2.1 Архитектурная проблематика

2.1.1 Базовая архитектура построения вычислительных комплексов

Сегодня для реализации портативных программных комплексов мониторинга показателей здоровья человека популярны 3 типа архитектур: локальная, клиент-серверная и трехуровневая. Локальная архитектура является самой независимой, так как весь функционал решений, построенных на данной архитектуре, сконцентрирован в пределах одного устройства. Это же и является главным недостатком такого подхода, так как ограничивает масштабируемость разрабатываемого комплекса. Решает эту проблему клиент-серверная архитектура, позволяя разгрузить нагрузку на устройстве, где сервер, используя систему управления базами данных с хранимыми процедурами, позволяющими взять на себя небольшую часть бизнес-логики вычислительного комплекса. Однако все равно хранимые процедуры являются ограниченным решением для комплексов, которые требуют реализовать обширный функционал, в результате чего применяется трехуровневая архитектура (рисунок 2.1).

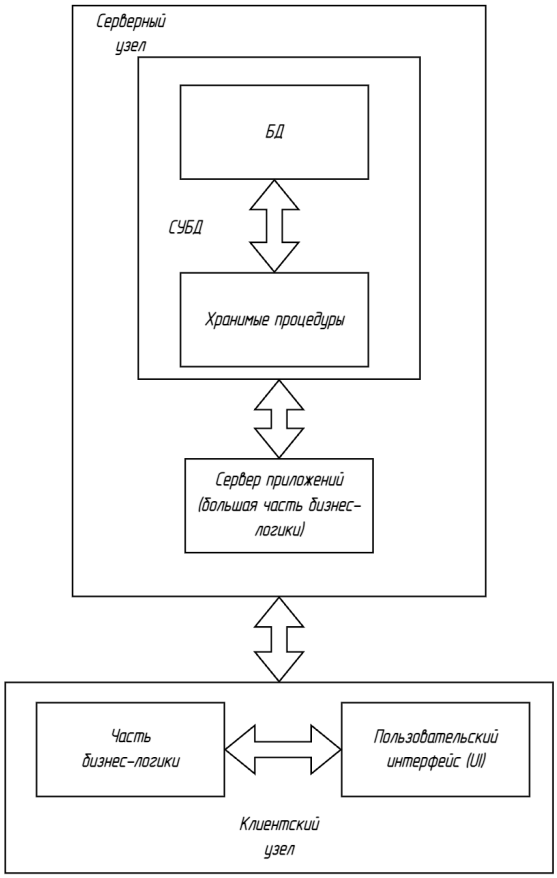


Рисунок 2.1 – Структура трехуровневой архитектуры

В дополнение к клиент-серверной архитектуре тут добавляется так называемый сервер приложений, основное назначение которого заключается в выполнении еще большей части бизнес-логики программного комплекса, снижая вычислительную нагрузку на клиентский узел. При создании портативного программно-аппаратного комплекса для неинвазивного контроля показателей жизнедеятельности человека оптимальным решением выступает трехуровневая архитектура. Обработка информации требует реализации множества вычислительных алгоритмов для каждого пользователя с мобильным устройством. Ключевые аспекты комплекса используют анализ акустических сигналов и обработку визуальных данных, которые занимают центральное место в диагностике физиологических параметров и могут требовать ресурсоемких алгоритмов машинного обучения. Данные методы, ставшие стандартом в задачах распознавания паттернов и классификации биометрических данных, обладают высокой вычислительной сложностью, что исключает их реализацию через хранимые процедуры в СУБД. Кроме того,

выполнение таких вычислений на клиентских узлах приводит к критической нагрузке на их аппаратные ресурсы, снижая производительность и стабильность системы. Стоит отметить, что в данном случае также возможно использовать кэширование часто запрашиваемых данных для снижения нагрузки на БД.

2.1.2 Проблематика архитектуры программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека

Ввиду необходимости реализовать определение большого количества показателей здоровья человека простого использования трехуровневой архитектуры недостаточно. Современные вычислительные комплексы неинвазивного мониторинга показателей жизнедеятельности человека имеют определенный недостаток, он связан тем, что такие комплексы требуют адаптации к новым данным и устройствам (если речь идет о реализации узлов на смартфонах). Можно выделить ряд проблем касаемо этого недостатка:

1. Проблема устаревания моделей.

Так как современные решения все больше опираются на использование моделей машинного обучения все чаще встает вопрос об актуальности моделей машинного обучения, так как модели, обученные на исторических данных, теряют точность со временем. Причиной этого являются: появление новых версий измерительных сенсоров (с точки зрения технических характеристик этих сенсоров), изменение клинических проявлений отклонений со временем или методов диагностики. Возникает так называемый дрейф данных (Data drift) – сдвиг в распределении входных признаков между обучающей выборкой и текущими получаемыми признаками. Таким образом, программный комплекс, реализованный на смартфонах более поздних версий, может стать причиной снижения качества работы моделей машинного обучения ввиду того, что модели становятся устаревшими. В статье авторы обращают внимание, что в решениях, применяемых в медицине с использованием моделей машинного обучения сдвиг данных, играет немаловажную роль, поэтому применение процедур, [11, 41] направленных на

выявление необходимости переобучения, перекалибровки моделей является необходимым для поддержания качества модели машинного обучения со временем.

Авторы [42] также поднимают вопрос мониторинга сдвига данных моделей, в качестве примера они приводят ситуацию, которая произошла в больнице университета Мичиган, где применялась модель, оповещающая о возникновении сепсиса. Со временем данная модель начала выдавать множество ложных сработок, в результате чего данная модель машинного обучения была отключена. Основной причиной стало то обстоятельство, что демографические характеристики пациентов изменились из-за пандемии коронавируса COVID-19. Сдвиг набора данных изменил взаимосвязь между лихорадкой и сепсисом. Так же в данной статье авторами представлены общие причины сдвига набора данных: технологические, изменения в условиях и населении, поведенческие изменения. Резюмируется, что необходимо проводить технический надзор над моделями машинного обучения.

Сами сдвиги данных поддаются мониторингу. В статье [43] описывается применение моделей машинного обучения в рамках методологии MLOps (Machine Learning Operations), суть которой заключается в объединении процесса разработки моделей машинного обучения и процесса эксплуатации в рамках одной единой системы. Ключевая проблема, решаемая данной методологией – разрыв между производственными решениями и экспериментальными моделями. В данной статье предлагается применение детектора дрейфа данных, цель которого является определение ситуации, при которой распределение запросов развернутой модели машинного обучения начинает сильно расходиться с обучающей выборкой, тем самым определяя моменты, когда модель становится ненадежной. На рисунке 2.2 представлена примерная структура системы асинхронного мониторинга модели машинного обучения используя технологии KF-Serving и KNative.

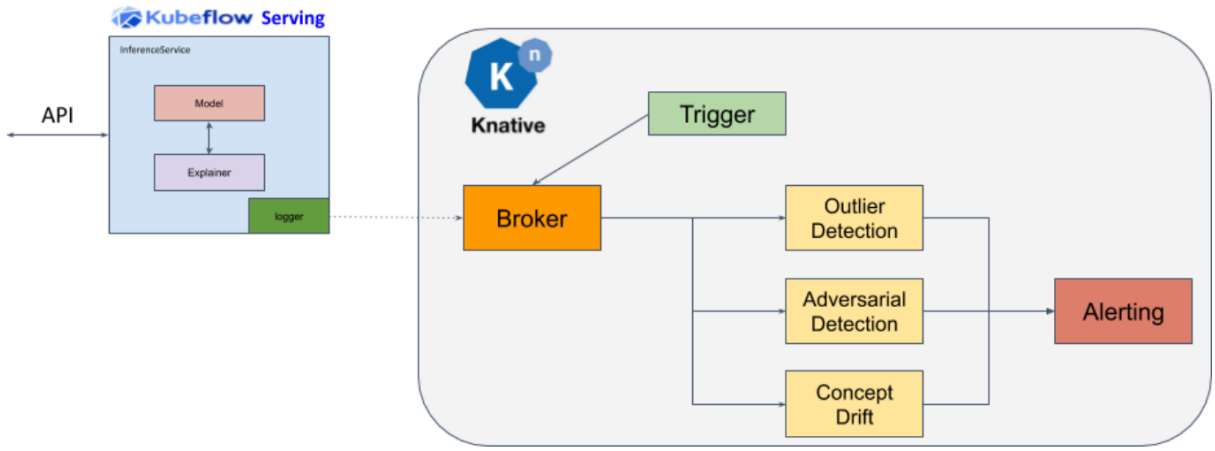


Рисунок 2.2 – Система асинхронного мониторинга моделей машинного обучения [43]

Базовым элементом является KF-Serving, который еще называют черным ящиком. Он состоит из 2 основных компонентов: непосредственно сама модель и так называемый объяснитель (Explainer). В то время как модель выдает предсказание, объяснитель в качестве результата выдает причину (объяснение) почему было сделано такое предсказание, которое необходимо для определения степени доверия к данной модели. KNative благодаря логированию выполняет задачу определения сдвига данных и сигнализирует о возникшем событии.

2. Проблема негибкой архитектуры.

Многие существующие медицинские комплексы построены в виде монолита и имеют жестко связанные компоненты. В монолитной системе изменения в одной компоненте требуют полной перекомпиляции вычислительного комплекса, повторного тестирования и развертывания всего комплекса в целом. Тем самым имеет место временные издержки, которые откладывают ввод в эксплуатацию обновленного программного комплекса. Монолитная архитектура не позволяет масштабировать отдельные компоненты независимо. Если нагрузка возрастет на один компонент, необходимо масштабировать весь серверный кластер, что приводит к перерасходу ресурсов и средств. В конечно счете стоит отметить, что даже если программный комплекс построен на базе трехуровневой архитектуры,

его узловые компоненты зачастую являются монолитами, что тянет за собой все проблемы монолитности.

Таким образом, предлагается использовать самоадаптирующуюся микросервисную архитектуру с использованием персонализированного федеративного обучения моделей. Основная идея данной архитектуры заключается в применении микросервисов использующих механизм оркестровки, в контроле ресурсов, а также в использовании методики обучения моделей машинного обучения персонализированного типа, известного как Федеративное обучение. При этом программный комплекс, построенный на базе самоадаптирующейся архитектуры выполняет мониторинг и адаптирует свое поведение и структуру для сохранения или улучшения качества работы программного комплекса.

2.2 Микросервисная архитектура

Для решения проблемы негибкой архитектуры предлагается переход на микросервисы в рамках трехуровневой архитектуры. Таким подходом в области медиа воспользовалась компания Netflix, применение которой описано в статье. Микросервисная архитектура представляет собой подход к разработке программного обеспечения, при котором приложение состоит из набора небольших, автономных сервисов. Каждый из этих сервисов разрабатывается, тестируется и разворачивается независимо, что позволяет командам разработчиков работать параллельно и эффективно. Этот подход был впервые описан в 2011 году на конференции Software Engineering Professionals, где отмечалось его отличие от монолитной архитектуры. Компании, такие как Netflix и Amazon, стали пионерами в применении микросервисной архитектуры, что обеспечило им высокую гибкость и масштабируемость систем, удовлетворяя изменяющиеся потребности пользователей. Как подчеркивают Клоков и Вечерская, «чтобы разбить бизнес-логику и уровни доступа к данным на независимые сервисы. Каждая служба имеет собственную базу данных и ориентирована на определенные бизнес-возможности». Этот принцип не только улучшает [8, 9, 45] управление

данными, но и способствует повышению скорости разработки и внедрения новых функций. Описанная ранее трехуровневая архитектура в большей степени представляет собой монолит. На рисунке 2.3 показана структура трехуровневой архитектуры с микросервисами.

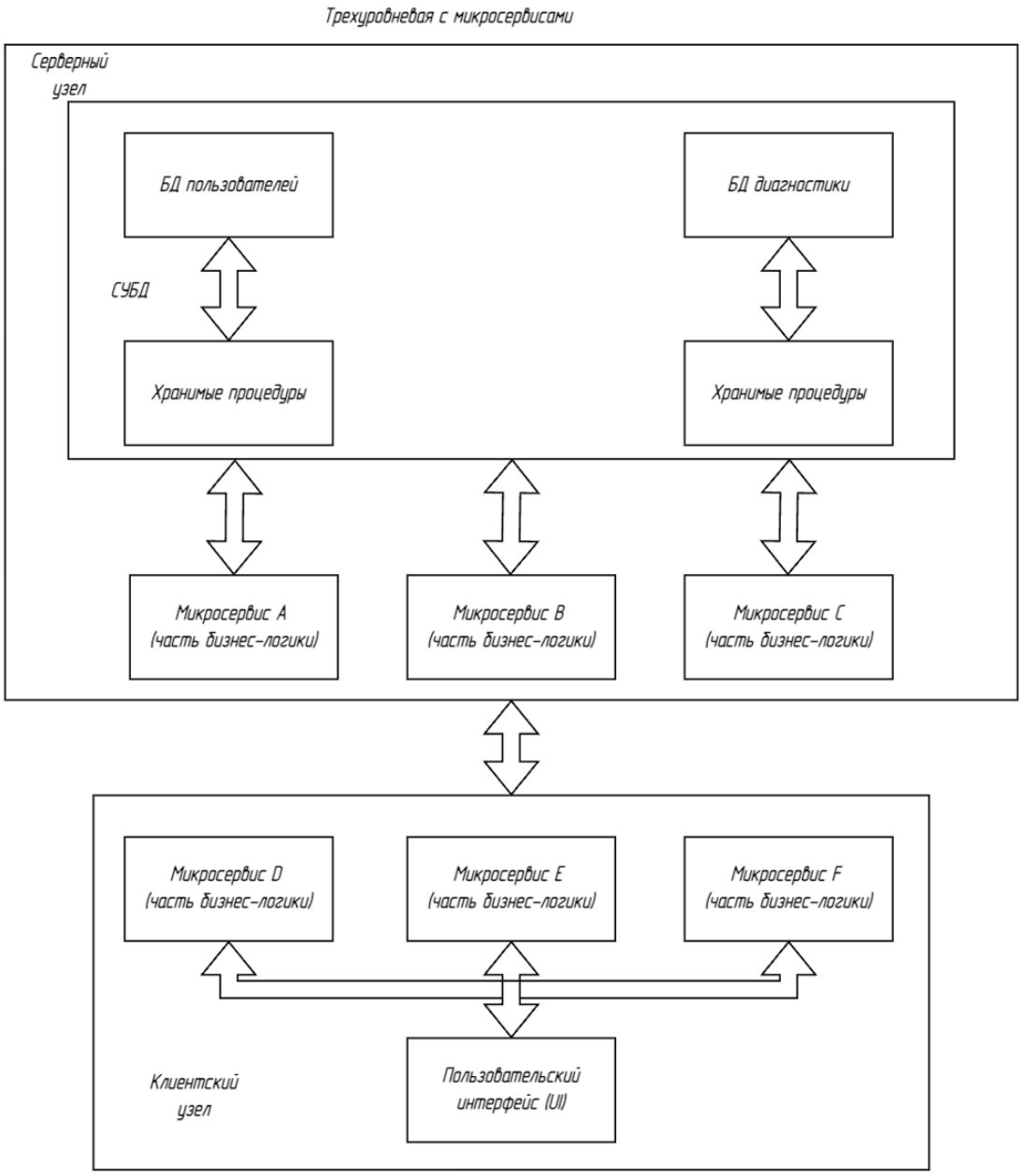


Рисунок 2.3 – Трехуровневая архитектура с микросервисами

Критерии размещения микросервисов на Клиенте либо Сервере определяются следующими факторами: доступностью сенсорных данных (акустическая, кинематическая и визуальная сенсорика доступны только на мобильном устройстве, поэтому микросервисы сбора и первичной обработки сенсорных данных размещаются на Клиенте); вычислительной сложностью

процедур (ресурсоемкие операции, такие как обучение и верификация моделей машинного обучения, выполняются на Сервере); требованиями к задержке и автономности (операции, критичные к времени отклика и не требующие постоянного сетевого соединения, размещаются на Клиенте); а также требованиями к конфиденциальности данных пользователя и минимизации объема передаваемого трафика (обновление моделей осуществляется агрегированно на Сервере).

Микросервисная архитектура представляет собой эволюционный подход к проектированию сложных программных систем, характеризующийся декомпозицией единого приложения на набор слабосвязанных, функционально специализированных и независимо развертываемых сервисов, каждый из которых реализует уникальную бизнес-возможность в рамках четко определенных границ контекста (Bounded Context) и взаимодействует с другими сервисами посредством легковесных, обычно асинхронных механизмов коммуникации (таких как RESTful API [46], gRPC [47] или обмен сообщениями через брокеры событий). Ключевые принципы включают: автономность сервисов (независимость жизненного цикла, стека технологий и баз данных), децентрализацию управления данными (каждый сервис владеет своей моделью данных и схемой БД), отказоустойчивость (реализуемая через паттерны типа Circuit Breaker), непрерывную поставку/развертывание (CI/CD) за счет независимости сервисов и горизонтальную масштабируемость (возможность масштабировать отдельные сервисы под нагрузку). Эта архитектурная парадигма контрастирует с монолитными системами, устраняя их фундаментальные ограничения – такие как сложность модификации кодовой базы, длительные циклы развертывания, неэффективное использование ресурсов и технологическую инерцию – но вводит новые вызовы: повышенную сложность оркестровки (управление через Kubernetes, Docker Swarm), необходимость реализации распределенных транзакций (Saga-паттерн [48]), обеспечение сквозной наблюдаемости (observability через трассировку, логи,

метрики) и проектирование устойчивых сетевых взаимодействий в условиях частичных отказов. Теоретической основой микросервисов являются концепции доменно-ориентированного проектирования (DDD) и принципа единой ответственности (SRP), перенесенные на уровень системной архитектуры, что позволяет организациям достигать большей гибкости, скорости разработки и технологической адаптивности в условиях динамично меняющихся требований и гетерогенных сред исполнения (гибридные облака, edge-устройства).

Эффективное управление распределенными программными комплексами требует применения современных подходов, позволяющих снизить сложность и повысить надежность их работы. Одним из ключевых методов является автоматизация процессов управления. Согласно отчету компании Gartner за 2021 год, внедрение автоматизированных систем управления распределенными системами позволяет сократить время на настройку и обслуживание до 40%. Это достигается за счет минимизации ручного вмешательства, что уменьшает вероятность ошибок и ускоряет выполнение операций. Автоматизация также способствует улучшению мониторинга и реагирования на изменения в системе, что критически важно для обеспечения её стабильной работы. Согласно Никитину и Гриценко [16], «при необходимости изменить какой-то параметр, можно достаточно легко найти зависимости всех параметров между собой». Вместе с тем, они указывают на уязвимость такого подхода, отмечая, что «единой точкой отказа всей системы» является одна точка входа приложения. Это подчеркивает необходимость комплексного подхода к управлению, который учитывает не только автоматизацию, но и архитектурные решения, способствующие повышению эффективности системы.

Паттерн Circuit Breaker (Автоматический Выключатель) [17, 49] представляет собой критически важный механизм повышения отказоустойчивости распределенных систем, реализующий превентивную логику изоляции сбоев для предотвращения каскадных отказов при

деградации зависимых сервисов или сетевых ресурсов. Концептуально он моделирует поведение электротехнического предохранителя: при превышении порога ошибок (error threshold) входящих запросов к целевому сервису, выключатель детектирует аномальное состояние и переходит в состояние «разомкнутой цепи» (Open State), мгновенно блокируя дальнейшие вызовы на протяжении заданного таймаута (reset timeout). В этом состоянии система генерирует исключение или возвращает запасное значение (fallback response) без реального взаимодействия с неисправным компонентом, минимизируя нагрузку на деградировавший ресурс и снижая латентность обработки запросов клиента. После истечения таймаута выключатель переходит в состояние «полуразомкнутой цепи» (Half-Open State), пропуская ограниченное количество пробных запросов для верификации восстановления сервиса. При успешном выполнении этих запросов цепь автоматически замыкается (Closed State), возобновляя нормальную работу; в противном случае таймаут сбрасывается, и цикл повторяется (рисунок 2.4).

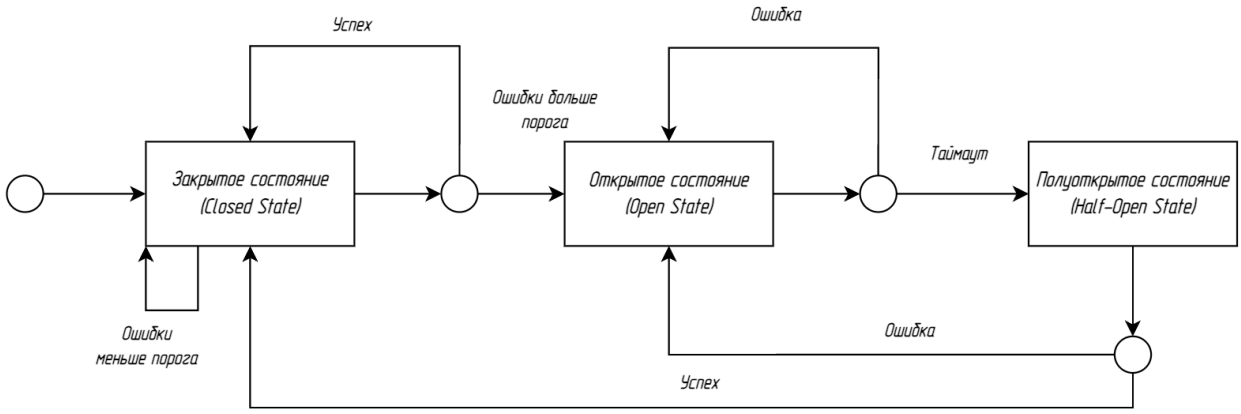


Рисунок 2.4 – Паттерн Автоматический Выключатель

Математически модель описывается конечным автоматом с тремя состояниями, где переходы между ними управляются счетчиком ошибок (например, скользящее окно за последние T секунд) и пороговой функцией ($failure_rate > \theta$). Ключевые параметры конфигурации включают:

1. Порог срабатывания (*failureThreshold*) – минимальный процент ошибок для активации Open State.

2. Длительность таймаута (*resetTimeout*) – время блокировки вызовов в Open State.

3. Объем тестовой выборки (*sampleSize*) – число запросов в Half-Open State для верификации.

Таким образом, микросервисная архитектура представляет собой современный подход к проектированию приложений, который обеспечивает их гибкость, масштабируемость и адаптивность к изменениям. Данный подход, разделяющий приложение на независимые сервисы, позволяет ускорить разработку, повысить производительность комплекса и упростить внедрение новых функций. Однако внедрение микросервисов сопровождается сложностью управления распределенными комплексами. Для успешной реализации микросервисного подхода критически важно учитывать как технические, так и организационные аспекты.

2.3 Федеративное обучение

2.3.1 Основные особенности

Федеративное обучение (FL) представляет собой распределённую парадигму машинного обучения, при которой децентрализованные клиентские устройства (смартфоны, IoT-сенсоры, локальные серверы) совместно обучают общую глобальную модель, не передавая исходные приватные данные в централизованное хранилище, тем самым обеспечивая соблюдение принципов конфиденциальности и соответствие регуляторным требованиям (GDPR, HIPAA, 152-ФЗ[50]). Технически FL реализуется через итеративный процесс:

1) Центральный сервер рассылает актуальную глобальную модель Θ_tклиентам.

2) Каждый клиент k локально обучает модель на своих данных, вычисляя градиенты.$D_k \nabla L(\Theta_t; D_k)$

3) Только обновления модели (например, градиенты или веса) агрегируются на сервере с использованием алгоритмов типа Federated Averaging (FedAvg):

$$\Theta_{t+1} = \sum_{k=1}^{K} \frac{|\mathrm{D}_k|}{N} \Theta_t^{(k)}, N = \sum_{k=1}^{K} |\mathrm{D}_k|, \quad (2.1)$$

где: $\Theta_t^{(k)}$ – локальная модель клиента k, $|\mathrm{D}_k|$ – объем его данных.

Помимо конфиденциальности такой вид обучения позволяет избегать передачу данных на центральный серверный узел, тем самым снижая затраты на передачу данных, а также обеспечивается персонализация данных, не только для одного пользователя, но и для группы пользователей [18]. При этом алгоритм FedAvg сам по себе не является механизмом персонализации: он обеспечивает построение общей глобальной модели, тогда как персонализация достигается последующей локальной адаптацией полученной модели под данные конкретного пользователя, в том числе путем дообучения отдельных слоев модели. В работе под персонализированным федеративным обучением понимается связка «агрегация обновлений по FedAvg + локальное дообучение на пользовательском устройстве», обеспечивающая индивидуальную настройку модели без потери преимуществ коллективного обучения.

Термин «федеративное» подчеркивает, что обучение организовано как объединение (федерация) независимых клиентских узлов: каждый участник сохраняет владение своими данными и выполняет локальное обучение на собственном устройстве, а серверный узел лишь координирует процесс и агрегирует параметры моделей. В отличие от классического распределенного обучения (distributed learning), при котором единый владелец данных распределяет обучение между узлами кластера для ускорения вычислений и при этом располагает всеми исходными данными, федеративное обучение решает задачу обучения модели в условиях, когда сбор исходных данных в едином центре невозможен или запрещен: данные физически не покидают устройства пользователей. Для разрабатываемого программного комплекса именно это свойство определяет выбор FL: показатели жизнедеятельности относятся к персональным данным, их передача на серверный узел в исходном виде ограничена требованиями законодательства (152-ФЗ), а локальное

обучение с передачей и агрегацией только параметров моделей обеспечивает соблюдение требований конфиденциальности при сохранении качества моделей.

Федеративное машинное обучение можно классифицировать на 2 вида:

1) Горизонтальное федеративное обучение – имеет место разный набор объектов и одинаковый набор признаков.

2) Вертикальное федеративное обучение – имеет место одинаковый набор объектов и разный набор признаков.

Упрощенная схема архитектуры федеративного обучения представлена на рисунке 2.5.

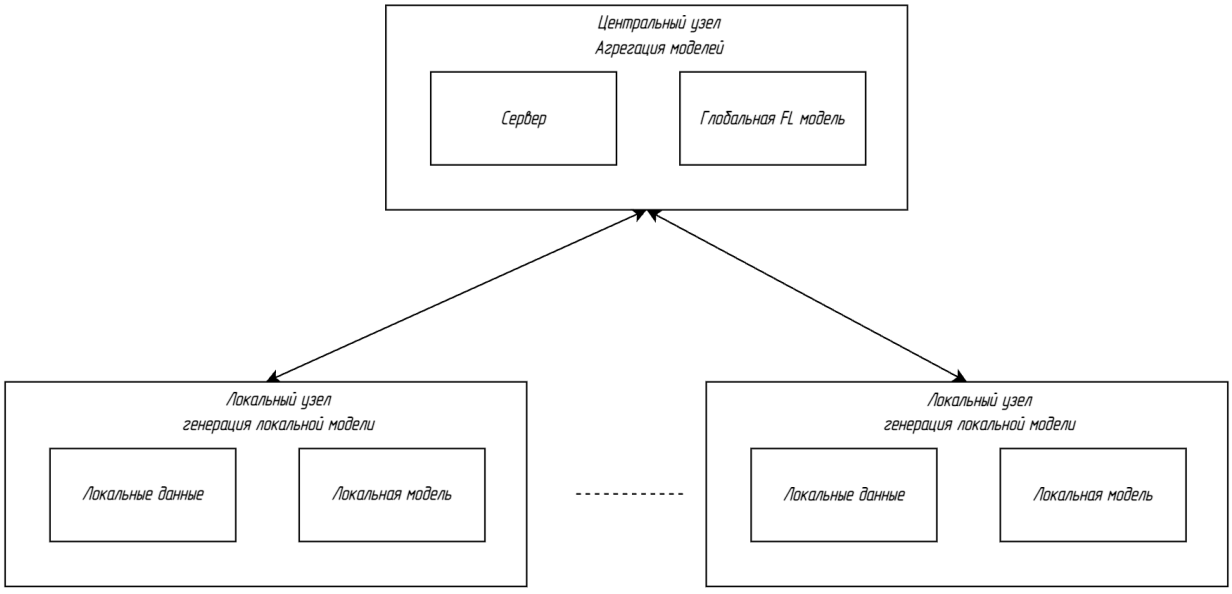


Рисунок 2.5 – Упрощенная архитектура Федеративного Обучения (FL)

Архитектурные компоненты системы федеративного обучения включают три ключевые роли:

1. Координатор (центральный сервер): детерминирует глобальный цикл обучения, инициируя процесс рассылкой инициализированной модели участникам, агрегируя локальные обновления посредством алгоритмических методов, и распределяя результирующую глобальную модель для следующей итерации.

2. Участники (клиентские узлы): сущности, владеющие распределенными приватными наборами данных. Выполняют

локализованную оптимизацию модели, применяя стохастический градиентный спуск к своим данным, и передают параметрические обновления (градиенты или веса) координатору без экспорта исходных данных.

3. Криптографический посредник (арбитр): факультативный доверенный компонент в вертикальных схемах FL, обеспечивающий безопасное согласование пространств признаков между гетерогенными участниками. Здесь используется гомоморфное шифрование – это криптографический метод, позволяющий выполнять арифметические операции на зашифрованных данных без их предварительного расшифрования, обеспечивая корректность результата в зашифрованном виде. Формально, для операций $\oplus$ и$\odot$, функция шифрования E удовлетворяет свойству:

$$E(a) \oplus E(b) = E(a \odot b), \tag{2.2}$$

где: a, b – открытые данные.

Авторы [13] успешно применяют нейронные сети на основе федеративного обучения для решения задачи выявления аномалий в сетевом трафике для обнаружения потенциальных хакерских атак. Авторы используют федеративную графовую нейронную сеть и проводят его сравнение со стандартной централизованной сетью. Особенность представленной модели машинного обучения заключается в объединении подхода графовых нейронных сетей и федеративного обучения. Основная конвейерная блок-схема методологии представлена на рисунке 2.6.

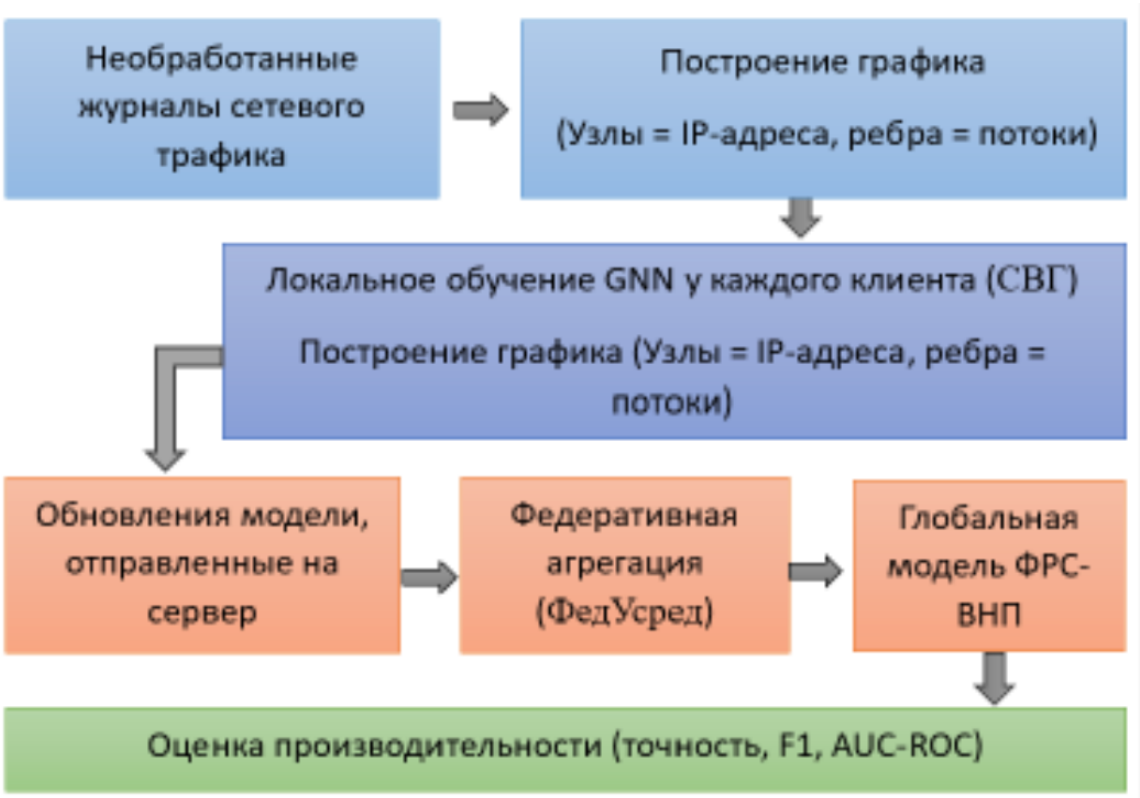


Рисунок 2.6 – Конвейерная блок-схема методологии [13]

Результаты проведенных испытаний авторами показали, что использование федеративного обучения практически не уступает стандартному подходу: 92% против 93,2% по метрике точности.

В работе [51] авторы представили классификацию подходов и методов федеративного обучения по следующим типам архитектуры: централизованная, децентрализованная и гибридная. Авторы рассмотрели ряд технологий, в число которых входит Knowledge Distillation, блокчейн, Hybrid-IoT, DHA-FL и другие. Особый интерес представляют централизованная и гибридная модели. Авторы заключили, что централизованная модель обеспечивает высокую точность и сходимость за счет центрального сервера, который может использовать все данные со всех устройств, тем самым позволяя оптимизировать глобальную модель, к тому же она проста в реализации, так как сервер контролирует этапы агрегации и обучения моделей машинного обучения. Гибридная же модель позволяет выполнять обучение кластеров в локальных серверах, которые впоследствии можно использовать для формирования глобальной модели. Кластеры снижают нагрузку на обмен

данными (что является узким местом централизованной модели) и обеспечивают отказоустойчивость.

2.3.2 Федеративное обучение и микросервисы

Микросервисная архитектура, благодаря своей гибкости и масштабируемости, является идеальной платформой для реализации федеративного обучения. Она позволяет распределять вычислительные нагрузки между различными компонентами системы, что в свою очередь способствует улучшению производительности и надежности. Компании, такие как Netflix, успешно применяют микросервисы для обработки больших объемов данных. Современные облачные платформы, включая AWS и Google Cloud, предоставляют инструменты, которые упрощают внедрение микросервисной архитектуры и её интеграцию с федеративным обучением. Это создает возможности для разработки эффективных систем мониторинга жизнедеятельности человека, обеспечивая высокий уровень конфиденциальности данных при сохранении высокой производительности. Таким образом, микросервисная архитектура не только поддерживает распределение вычислительных процессов, но и способствует оптимизации работы алгоритмов, используемых в федеративном обучении. Интеграция федеративного обучения с микросервисной архитектурой предоставляет значительные преимущества в разработке и поддержке сложных систем. Микросервисы позволяют независимо обновлять или изменять каждую составляющую системы, что не нарушает работу других компонентов. Это обеспечивает целостность и непрерывность обучения моделей, что особенно важно в условиях распределенных данных и динамично изменяющейся информационной среды. Важно отметить, что «интеграция приложений – неотъемлемая часть для успешного функционирования любого IT проекта». Таким образом, сочетание федеративного обучения и микросервисов становится ключевым аспектом в создании эффективных и адаптивных систем.

2.4 Представление архитектуры программного комплекса динамического неинвазивного мониторинга показателей жизнедеятельности человека

2.4.1 Архитектурная модель

В первую очередь сначала была определена архитектура программного комплекса без использования микросервисной архитектуры, которая представлена на рисунке 2.7. Комплекс состоит из множества модулей (в том числе реализующие алгоритмы, определяющие параметры, которые были выбраны ранее), как в клиентской, так и в серверной части, следуя концепциям трёхуровневой архитектуры. Модули, которые использовали сложные большие модели машинного обучения и опирались на ряд показателей, определяемых другими модулями, были размещены на серверном узле.

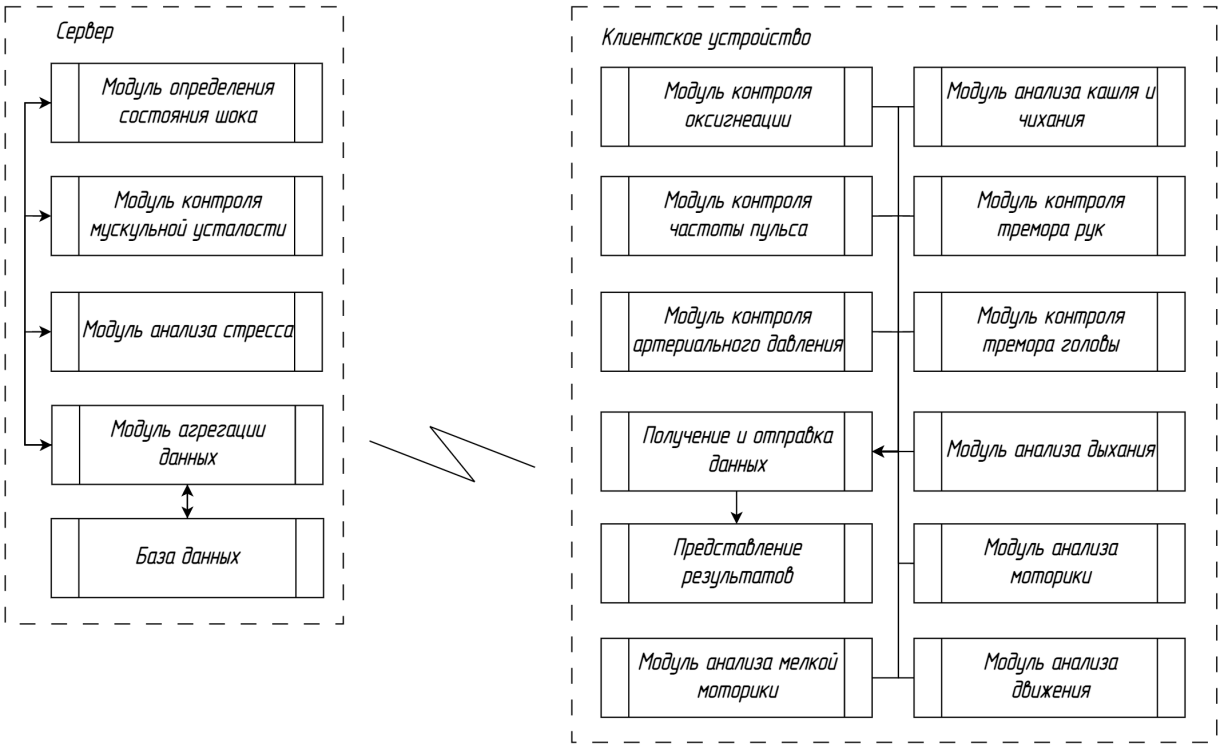


Рисунок 2.7 – Первоначальная архитектура программного комплекса неинвазивной диагностики показателей жизнедеятельности человека

Следующим этапом является переход заданной архитектуры в архитектуру микросервисов с оркестровкой. Для построения архитектуры программного комплекса были определены перечень микросервисов как для

работы серверного узла, так и для клиентских узлов. Было принято решение использовать для серверного узла следующие компоненты:

- распределительный шлюз оркестратор;
- микросервис определения состояния шока;
- микросервис контроля мускульной усталости;
- микросервис анализа стресса;
- микросервис авторизации;
- микросервис результатов;
- микросервис федеративного обучения;
- база данных авторизации;
- база данных результатов;
- база данных глобальных моделей федеративного машинного обучения.

Архитектуру микросервисов на клиентском устройстве не применяют в чистом виде ввиду того, что вся работа клиентского приложения реализуется в рамках одной сборки, и сами устройства имеют ограниченные ресурсы, однако существуют подходы, которые вдохновлены архитектурой микросервисов и позволяют реализовать некоторые его аспекты, в частности:

- модульность;
- изоляция ответственности;
- распределение ресурсов.

В данной работе реализация клиентского узла предназначена для устройств, работающих на операционной системе (ОС) Android. Таким образом, диаграмма развертывания комплекса после внедрения микросервисов представлена на рисунке 2.8. Упрощенная диаграмма классов (без GUI компонентов) клиентского модуля показана на рисунках 2.9, 2.10.

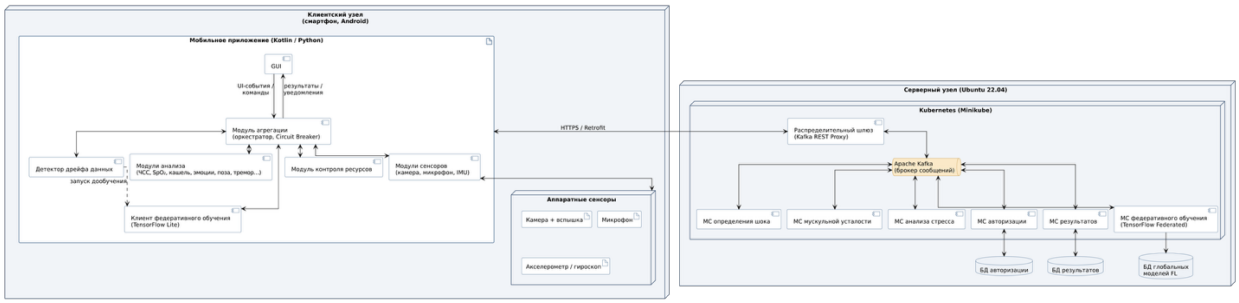


Рисунок 2.8 – Диаграмма развертывания программного комплекса неинвазивной мониторинга показателей жизнедеятельности человека

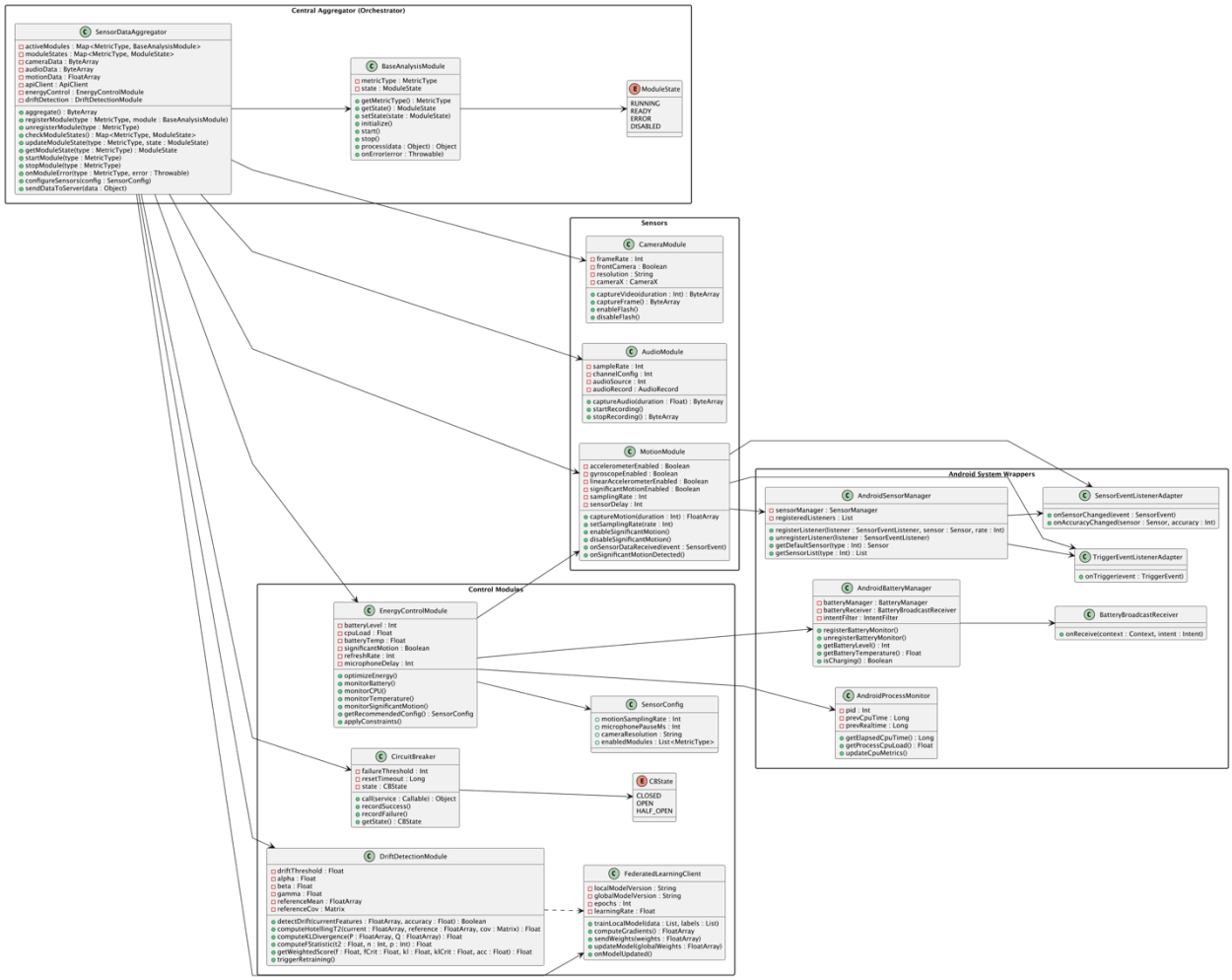


Рисунок 2.9 – Упрощенная диаграмма классов клиентского модуля программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека, часть 1

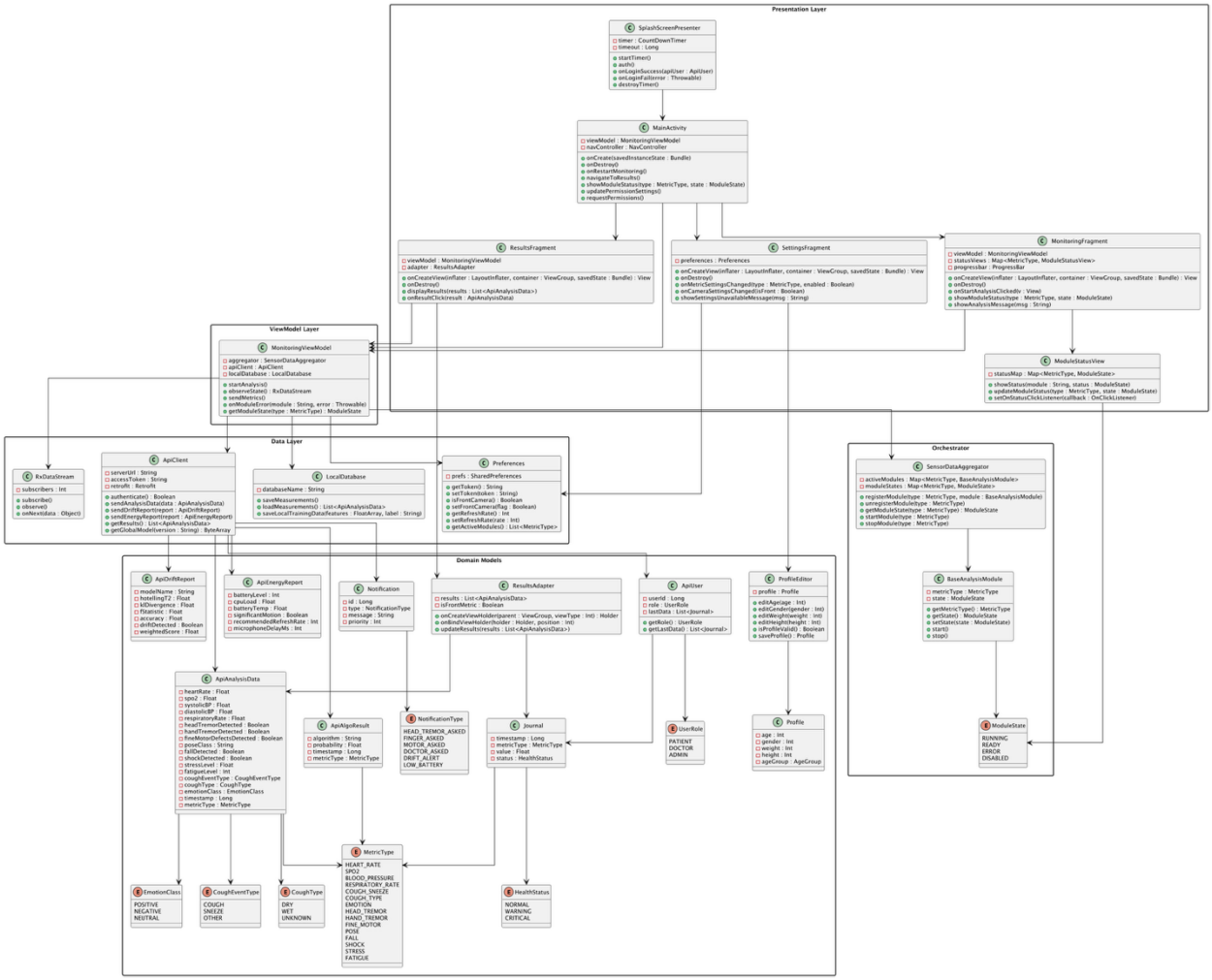


Рисунок 2.10 – Упрощенная диаграмма классов клиентского модуля программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека, часть 2

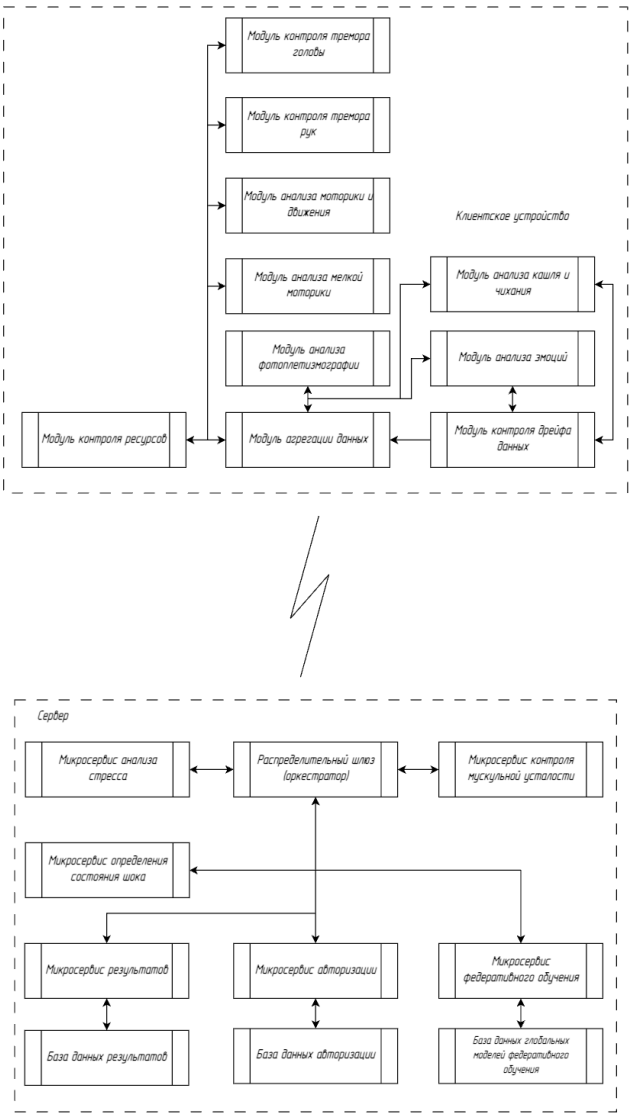


Рисунок 2.11 – Общая архитектура программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека

Самоадаптация программного комплекса реализуется за счет двух механизмов. Первый механизм связан с обнаружением дрейфа данных и запуском процедуры федеративного дообучения моделей машинного обучения. Второй механизм связан с адаптивным изменением режимов работы сенсорных и вычислительных компонентов мобильного приложения в зависимости от состояния устройства и требований к качеству анализа.

Как было указано ранее клиентские узлы построены на базе одного приложения для ОС Android написанный на языках Python и Kotlin. Серверная платформа функционирует под управлением ОС Ubuntu, а микросервисная архитектура реализована на языках программирования Python и Matlab.

Оркестровка микросервисов осуществляется средствами Kubernetes, а для управления базами данных применяется СУБД PostgreSQL. Структура баз данных представлена на рисунке 2.12.

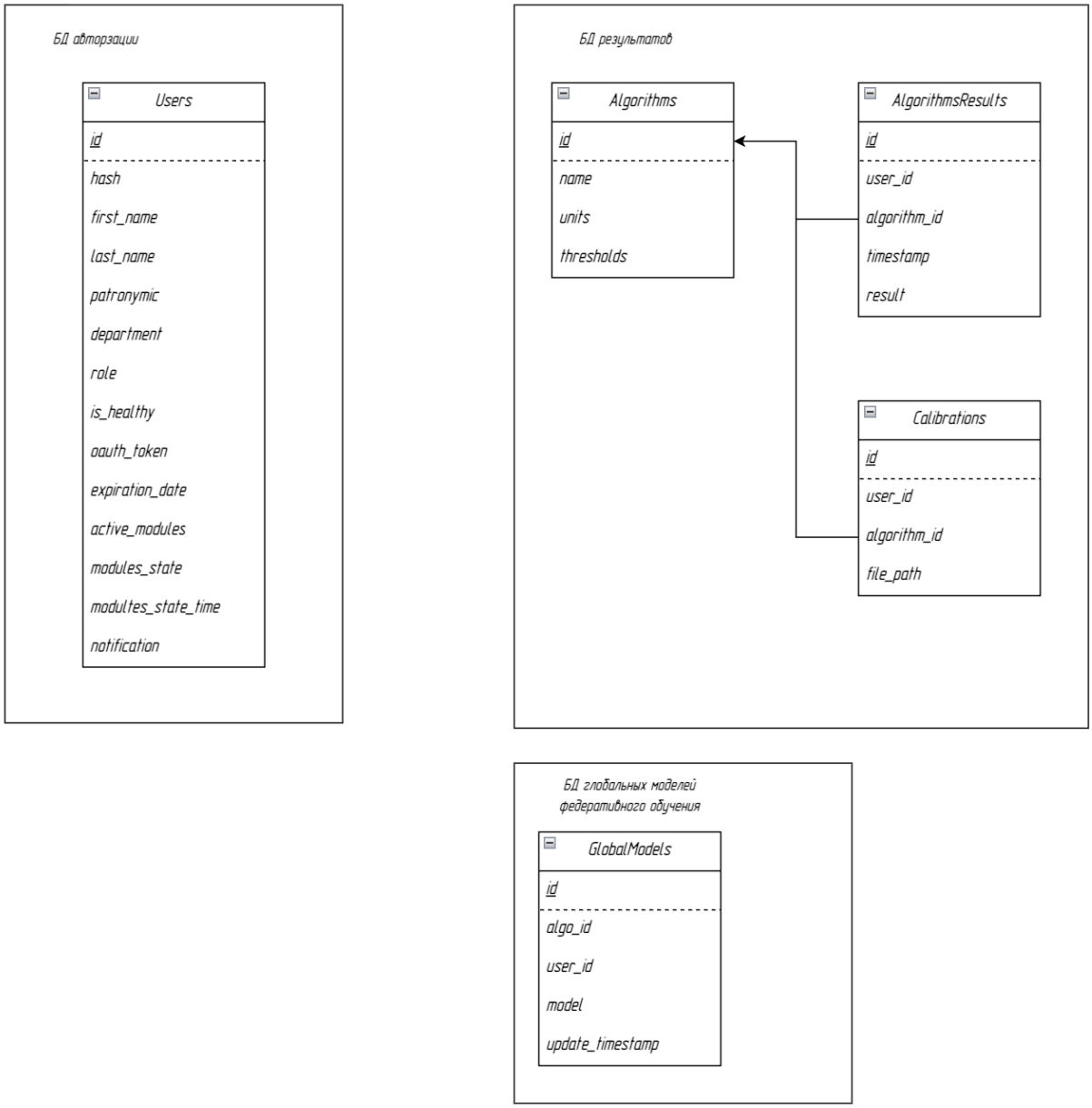


Рисунок 2.12 – Структура баз данных клиентского модуля

Отличительной особенностью разработанной архитектуры является изменяемость состава микросервисов в процессе функционирования системы. Однако разработанное решение не является прямым применением классической микросервисной модели, а представляет собой её модификацию,

ориентированную на условия функционирования мобильных устройств и задачи мониторинга показателей жизнедеятельности.

В зависимости от контекста использования и состояния системы могут происходить следующие изменения:

- динамическое подключение или отключение отдельных микросервисов анализа;
- изменение параметров работы сервисов (частота обработки, используемые модели);
- добавление новых сервисов при расширении функциональности комплекса;
- перераспределение вычислительной нагрузки.

Данный механизм обеспечивает самоадаптацию программного комплекса к изменяющимся условиям эксплуатации и входным данным.

Формально модель самоадаптации может быть представлена следующим образом. Пусть C(t) – конфигурация программного комплекса в момент времени t, включающая число и состав активных модулей, параметры их функционирования (частоты дискретизации сенсоров, режимы обработки), версии моделей машинного обучения и параметры взаимодействия клиентского и серверного узлов; X(t) – вектор признаков контролируемых показателей; S(t) – вектор состояния мобильного устройства (уровень заряда АКБ, температура, нагрузка на ЦП, наличие движения). Вектор критериев формирования управляющего сигнала принимает вид:

$$K(t) = (K_{T2}(t), K_{KL}(t), K_{res}(t)), \quad (2.8)$$

где: $K_{T2}(t)$ – статистика Т-квадрат Хотеллинга; $K_{KL}(t)$ – KL-дивергенция распределений признаков; $K_{res}(t)$ – ресурсные критерии, формирующие управляющий сигнал.

Изменение конфигурации описывается отображением:

$$C(t+1) = F(C(t), u(t)), \quad (2.9)$$

где: *F* – функция, определяющая запуск федеративного дообучения, изменение режимов работы сенсоров и состава активных модулей. Верификация процедуры самоадаптации заключается в сопоставлении контролируемых критериев с пороговыми значениями и в последующей оценке качества моделей машинного обучения на размеченных испытательных данных до и после управляющего воздействия (глава 4).

Таблица 2.1 – Матрица «событие – детектор – управляющее воздействие – изменяемый компонент – измеримый эффект»

Событие	**Детектор**	**Управляющее воздействие**	**Изменяемый компонент**	**Измеримый эффект**
Дрейф данных (изменение распределения признаков моделей МО)	Модуль контроля дрейфа данных (T^2 Хотеллинга, KL-дивергенция, взвешенная оценка)	Запуск процедуры федеративного дообучения; временное отключение модулей определения показателей	Модели МО определения кашля и чихания, эмоций; конфигурация клиентского узла	Улучшение метрик (F1, точность, полнота) моделей после дообучения
Снижение заряда аккумулятора, рост нагрузки на ЦП	Модуль контроля ресурсов	Изменение режимов работы сенсоров и частоты обработки данных	Сенсорные и вычислительные компоненты клиентского приложения	Увеличение времени работы устройства
Изменение состава активных модулей по выбору пользователя	Модуль агрегации данных (оркестратор) с механизмом Dynamic Feature Modules	Динамическое подключение или отключение модулей определения показателей, загрузка необходимого функционала	Состав активных модулей клиентского устройства	Экономия памяти и вычислительных ресурсов клиентского устройства
Ошибка или отказ модуля	Модуль агрегации данных (оркестратор), шаблон «автоматический выключатель»	Проверка и обновление статусов модулей, перераспределение вычислительной нагрузки	Состав активных модулей комплекса	Отказоустойчивость комплекса и сохранение качества анализа

2.4.2 Оркестровка

Оркестровка отдельно реализуется как для клиентской, так и для серверной части программного комплекса. Модуль агрегации данных клиентского устройства на ОС Android играет роль оркестратора и выполняет следующие базовые задачи:

- определяет количество задействованных в программном комплексе модулей определения показателей жизнедеятельности человека;
- регулирует параметры чтения данных сенсорики и их обработки у клиентского устройства;
- следит за состоянием использующихся модулей;
- обеспечивает обмен сообщениями с серверным узлом программного комплекса.

Для реализации модулей в клиентском приложении используются две базовые технологии: инъекция зависимостей через библиотеку Koin [52] и механизм Dynamic Feature Modules [53], который позволяет предоставлять только необходимый функционал программного модуля в зависимости от потребностей владельца программного комплекса. Каждый модуль, взаимодействующий с узлом, исполняющим роль частичного оркестратора, имеет одно из четырёх состояний:

- запущен – модуль полноценно функционирует и выполняет свою задачу;
- готов к запуску – модуль готов к выполнению своей основной задачи при получении команды на выполнение;
- ошибка – внутри модуля возникло исключение, которое не позволяет полноценно функционировать модулю;
- выключен – модуль на данный момент не используется в программном комплексе.

Механизм коммуникации между модулями обеспечивается средствами реактивного программирования RxKotlin. BPMN диаграмма оркестровки клиентского узла представлена на рисунках 2.13-2.15.

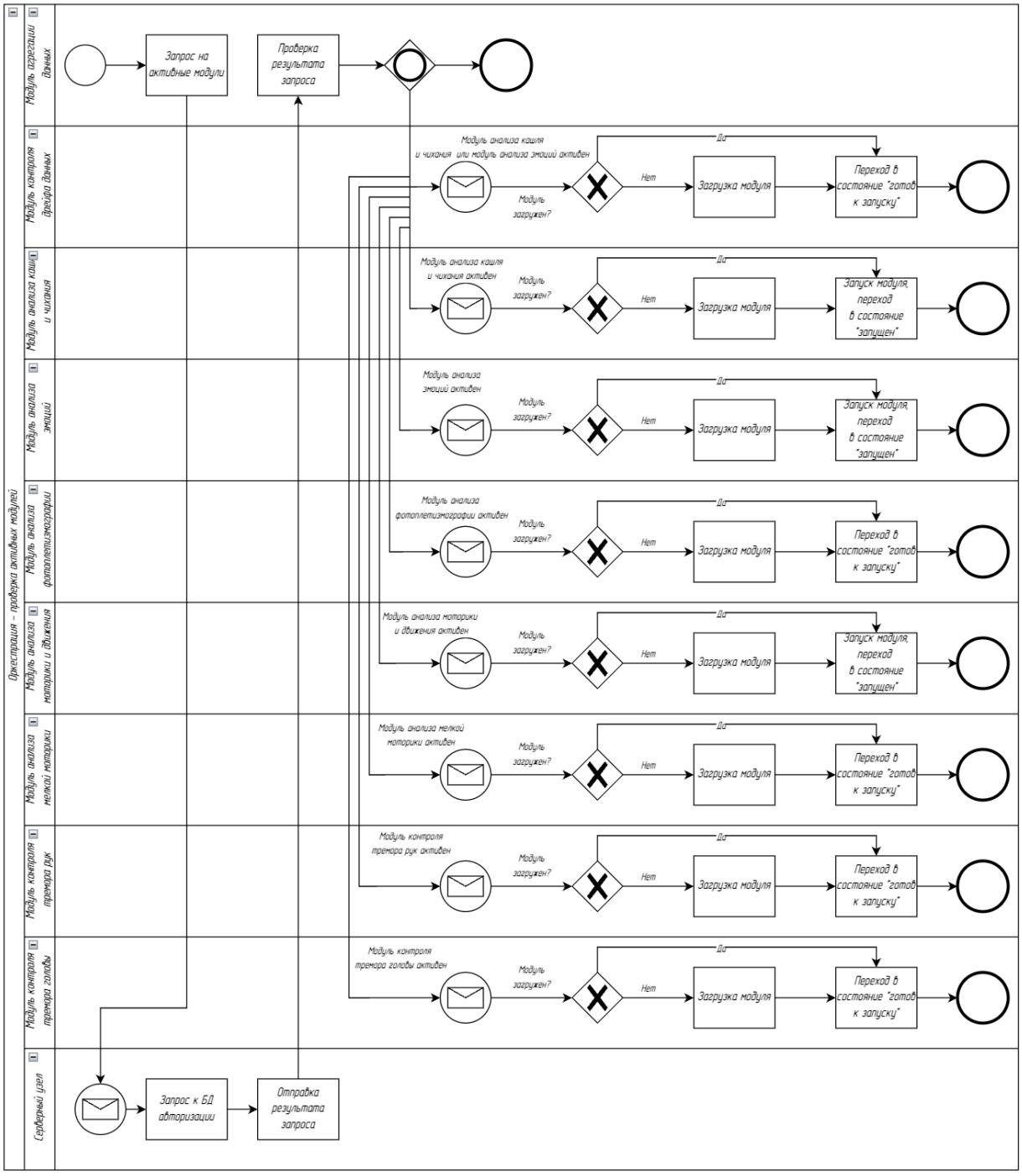


Рисунок 2.13 – BPMN диаграмма оркестровки клиентского узла программного комплекса для проверки активных модулей

Активные модули для пользователя позволяют сформировать гибкое решение, используя только необходимые алгоритмы для определения показателей жизнедеятельности человека и экономя память на клиентском устройстве. Если же модуль является активным для пользователя, то упомянутый ранее механизм Dynamic Feature Modules позволяет загрузить его.

Модули, которые не были выбраны активными, имеют свой внутренний статус “Выключен”.

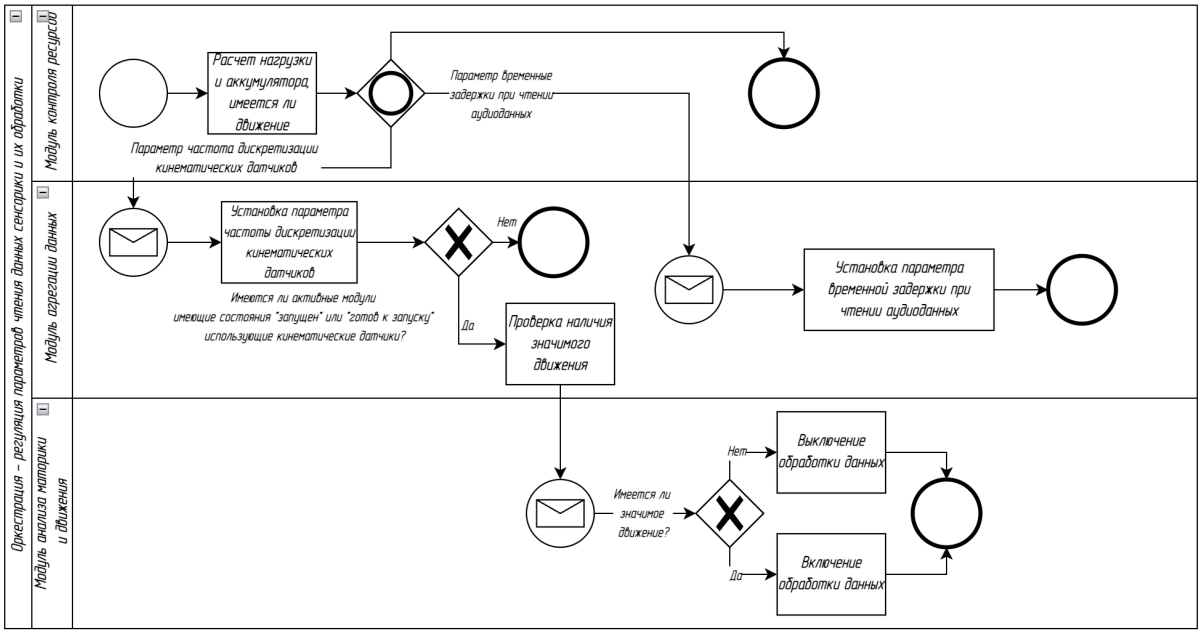


Рисунок 2.14 – BPMN диаграмма оркестровки клиентского узла программного комплекса для регуляции параметров чтения данных сенсорики и их обработки

Основным актором в процессе регуляции параметров чтения данных является модуль контроля ресурсов, который в полной мере описан далее в работе. На основе критериев, описанных в этом модуле, выполняется регуляция настроек работы с имеющейся сенсорикой для улучшения энергопотребления приложения клиентского устройства.

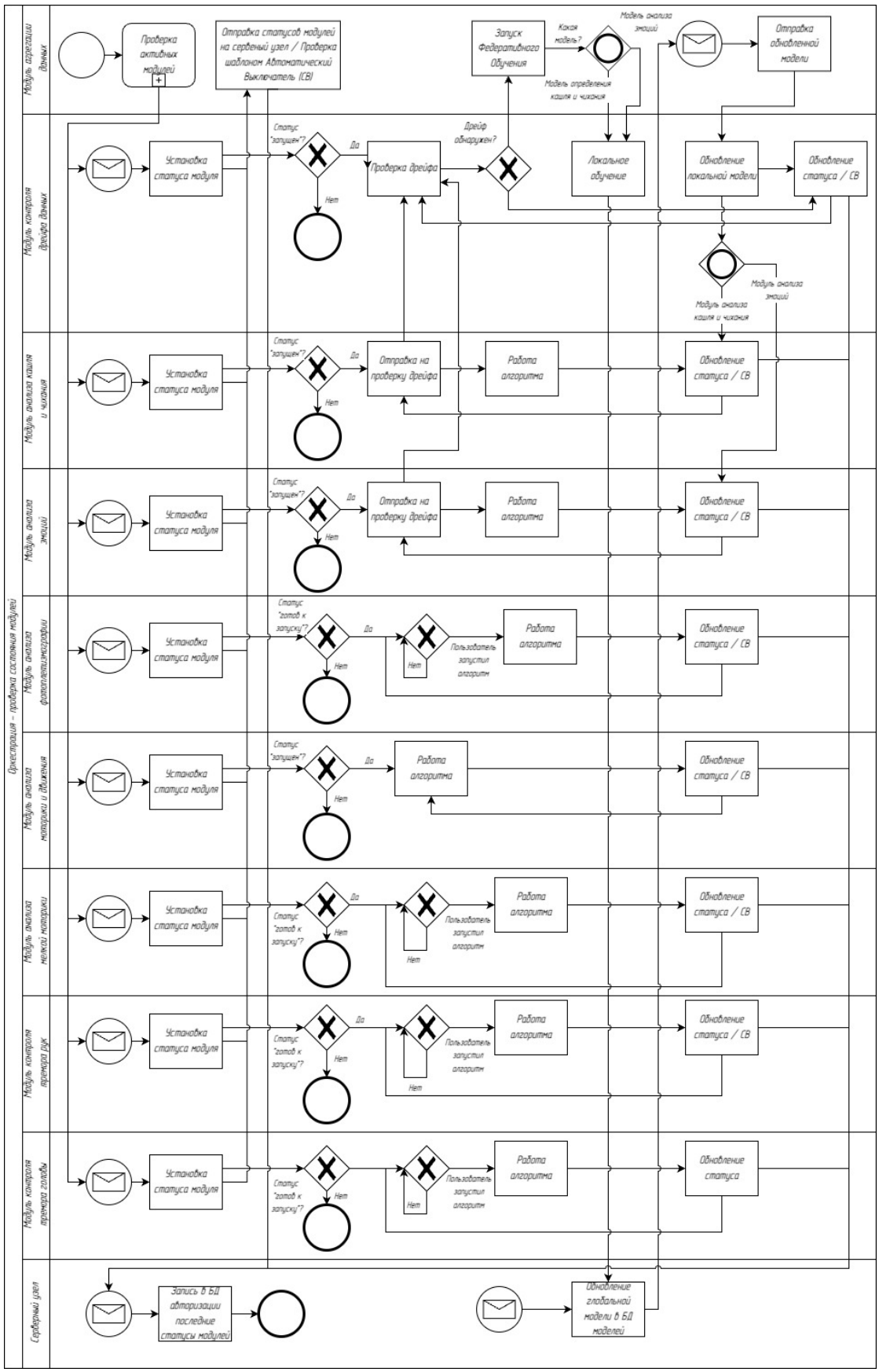


Рисунок 2.15 – BPMN диаграмма оркестровки клиентского узла программного комплекса для проверки состояния модулей и обмена сообщений

Так как каждый модуль имеет свой статус, модуль агрегации данных выполняет проверку и обновление этих статусов. При этом работа модулей следует шаблону автоматический выключатель.

Серверный узел, работающий на ОС Ubuntu, использует полноценные микросервисы Kubernetes версии Minikube 1.36.0. Для приема сообщений от клиентского узла (API Retrofit) используется обработка HTTP запросов с помощью Kafka REST Proxy. Каждый микросервис был реализован на языке программирования Python, а распределительный узел, играющий роль оркестратора с помощью средств асинхронного взаимодействия по технологии Kafka (доступный на Python). Объекты класса KafkaProducer формируют сообщения для отправки объектам класса KafkaConsumer. Так как Kafka является брокером сообщений, он также обладает возможностью сохранить сообщения в случае ошибки и выхода из строя какого-либо из микросервисов. Обработка запросов от клиентского узла показана на рисунке 2.16[54]

К основным сценариям отказа микросервисов серверного узла относятся: отказ отдельного микросервиса анализа, недоступность распределительного узла (оркестратора) и сбой микросервиса-потребителя сообщений. При отказе отдельного микросервиса вступает в действие механизм Circuit Breaker: на время таймаута вызовы к нему блокируются, а запросы обрабатываются с возвратом запасного значения, что исключает каскадное распространение отказа на остальные сервисы; после восстановления сервиса цепь возвращается в рабочее состояние через фазу пробных запросов (Half-Open). При сбое микросервиса-потребителя брокер Kafka сохраняет необработанные сообщения в топиках, что обеспечивает устойчивость к потере сообщений и позволяет возобновить обработку с сохранённого сообщения после восстановления сервиса.

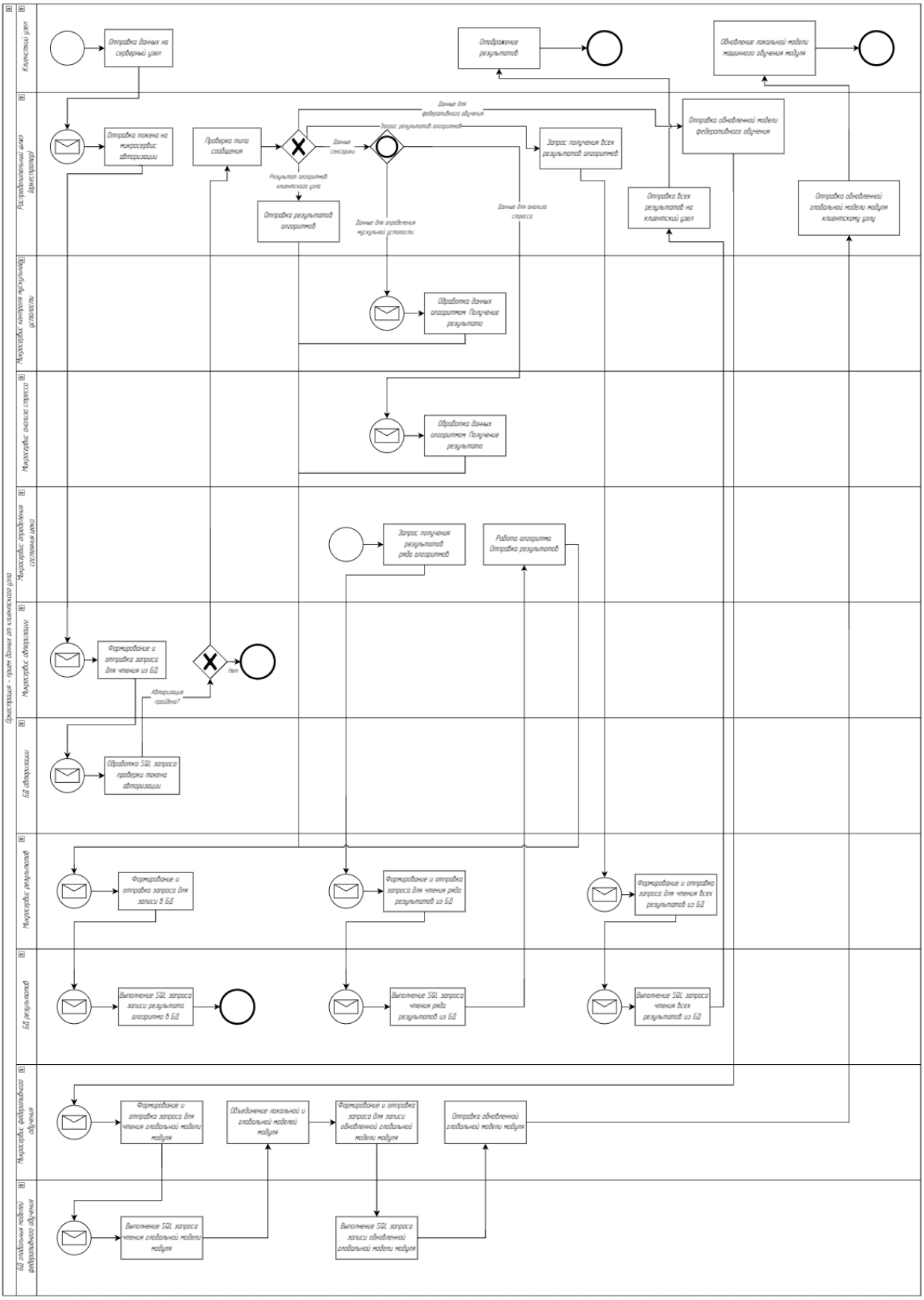


Рисунок 2.16 – BPMN диаграмма оркестровки серверного узла программного комплекса при обработке запросов клиентского модуля

2.4.3 Модуль контроля дрейфа данных

Данный модуль выполняет мониторинг двух модулей, использующих модели машинного обучения: модуль анализа кашля и чихания и модуль анализа эмоций. В условиях нестационарности данных векторов признаков моделей машинного обучения верификация моделей машинного обучения основывается на применении статистических критериев, направленных на выявление изменений распределений признаков и предсказаний моделей машинного обучения.

В качестве основного критерия наличия дрейфа было принято решение использовать многомерный критерий Т-квадрат Хотеллинга. Для учета нестационарности процесса и последовательного поступления данных критерий применяется в рамках скользящего окна, что позволяет отслеживать динамику изменения статистики во времени, тем самым обеспечивая локальную квазистационарность для глобально нестационарного процесса. Критерий выполняет сравнение эталонных признаков с текущими. Статистика Т-квадрат вычисляется по формуле (2.3) [55–56]:

$$T^2 = n\,(v_{curr} - v_{rev})^T\, S^{-1}\,(v_{curr} - v_{rev}), \tag{2.3}$$

где: v_{curr} – вектор средних текущих данных, v_{rev} – вектор средних эталона, n – размер выборки текущего набора данных, S – выборочная матрица ковариаций. Элементы ковариационной матрицы вычисляются по соотношению, показанному в формуле (2.4):

$$S = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T, \tag{2.4}$$

где: x_i – вектор-столбец признаков, $\bar{x}$ – вектор выборочных средних по каждому признаку.

Размер скользящего окна при расчёте статистики Т-квадрат составляет от 15 до 30 последних наблюдений: нижняя граница обеспечивает статистическую устойчивость оценок средних и ковариационной матрицы, верхняя граница ограничена требованием локальной квазистационарности процесса. Оценка ковариационной матрицы по наблюдениям окна (формула 2.4) является вырожденной, если размер выборки не превышает число

признаков p. Для обеспечения обратимости и устойчивости оценок применяется регуляризация: матрица ковариаций заменяется оценкой S* = S + ε·I, где I – единичная матрица, а ε = 10^{-6}·tr(S)/p – малая положительная константа, пропорциональная среднему значению диагональных элементов S. При n ≤ p матрица S сингулярна, поэтому для корректного расчёта статистики (2.3) используется псевдоинверсия Мура–Пенроуза, вычисляемая на основе сингулярного разложения (SVD). Условием применимости критерия Т-квадрат Хотеллинга является многомерная нормальность распределения признаков и выполнение соотношения n > p; в случае малых выборок регуляризация ковариационной матрицы и применение псевдоинверсии обеспечивают корректное вычисление статистики и сохраняют применимость критерия.

Следующим шагом является расчёт F статистики Фишера по формуле (2.5):

$$F = \frac{n-p}{p(n-1)} T^2, \tag{2.5}$$

где: p – количество признаков. Это значение используется для сравнения с критическим значением F статистики Фишера. Определяется из таблицы распределения Фишера [57], где число степеней свободы 1 определяется числом признаков p, число степеней свободы 2 – разностью n-p, а вероятность определяется уровнем значимости. Если вычисленное значение F превышает критическое значение Fкр, то считается, что дрейф обнаружен и инициируется процесс совершенствования модели машинного обучения методом федеративного обучения [57].

Дополнительно используется метрика KL-дивергенция (дивергенция Кульбака-Лейблера), определяемая по формуле:[58]

$$D_{KL}(P||Q) = \sum_x P(x) log \frac{P(x)}{Q(x)}, \tag{2.6}$$

где: P – начальное распределение, Q – текущее распределение. Для нормализованных признаков, если значение дивергенции больше 0,5 ($D_{KL\text{кр}}$) принимается решение о наличии дрейфа.

Для построения распределений P и Q используется гистограммная оценка функций плотности нормализованных признаков: распределение P строится по эталонному набору наблюдений, сформированному на этапе настройки детектора (исторические данные до начала мониторинга), распределение Q – по данным текущего скользящего окна, соответствующего окну расчёта статистики Т-квадрат. Для обеспечения сопоставимости распределения P и Q строятся на общей шкале: диапазон значений каждого признака разбивается на 10 интервалов одинаковой ширины, относительные частоты нормируются, благодаря чему сумма вероятностей каждого распределения равна 1. Для обработки нулевых вероятностей, возникающих при отсутствии наблюдений в отдельных интервалах, применяется аддитивное сглаживание: ко всем относительным частотам добавляется малая положительная константа $\varepsilon = 0{,}01$, после чего распределения повторно нормируются. В результате значения Pi и Qi строго положительны для всех интервалов, а значение KL-дивергенции остаётся конечным при любом объёме наблюдений в окне.

Для обеспечения учета качества модели в процедуре мониторинга дрейфа данных выполняется взвешенная оценка по формуле 2.7. Мониторинг дрейфа осуществляется 1 раз в день.

Использование интегрального критерия обусловлено неполнотой каждого из отдельных сигналов: статистика Т-квадрат фиксирует изменение распределения входных признаков, но не отражает непосредственное качество работы моделей машинного обучения, тогда как оценка точности (accuracy) отражает качество модели, однако реагирует на изменение распределения с запаздыванием и чувствительна к разметке. Объединение сигналов в виде взвешенной суммы с нормировкой относительно критических значений позволяет регистрировать дрейф на ранних стадиях по изменению распределения признаков и одновременно выявлять деградацию качества модели в отсутствие смещения распределения. Аддитивная форма выбрана ввиду наглядной интерпретируемости вклада каждого сигнала через весовой

коэффициент и исключения ложных срабатываний, характерных для минимаксных и мультипликативных комбинаций статистик.

$$D = \alpha \cdot \frac{F}{F_{\text{кр}}} + \beta \cdot \frac{D_{KL}}{D_{KL\text{кр}}} + \gamma \cdot (1 - accuracy), \quad (2.7)$$

где: α, β, γ – весовые коэффициенты, определяемые эмпирическим путем. Результат вычисления взвешенной суммы используется в качестве управляющего сигнала, инициирующего изменение конфигурации программного комплекса, включая запуск процедуры переобучения модели (локального обучения) и перераспределение вычислительных ресурсов. Блок-схема алгоритма определения дрейфа данных показан на рисунке 2.17. Следует отметить, что метрика корректности (accuracy) применяется в работе только при оценке качества моделей машинного обучения на размеченных испытательных данных (глава 4), где истинные метки известны. В реальной эксплуатации, где истинные метки отсутствуют, решение о наличии дрейфа принимается исключительно по статистическим критериям, не требующим разметки: многомерной статистике T^2 Хотеллинга и KL-дивергенции распределений признаков, объединяемым во взвешенной оценке. Таким образом, процедура дрейф-детекции не зависит от наличия истинных меток в потоке эксплуатационных данных.

Значения весовых коэффициентов в формуле (2.7) приняты равными: $\alpha = \beta = \gamma = 1/3$, что соответствует равнозначному учету всех трех составляющих интегрального критерия при отсутствии априорной информации о преимущественном вкладе какой-либо из них. Каждая составляющая нормируется относительно своего критического значения: статистика T-квадрат нормируется относительно критического значения Fкр распределения Фишера при уровне значимости $\alpha = 0{,}05$; KL-дивергенция нормируется относительно критического значения, принятого равным 0,5 (формула 2.6); составляющая (1 – accuracy) нормирована в естественном диапазоне [0; 1]. Итоговое правило принятия решения задано порогом 0,5: дрейф данных фиксируется при превышении взвешенной оценкой D значения 0,5, что соответствует преобладанию положительных показаний, составляющих

критерия для нормированных индикаторов. Статистическая обоснованность критерия обеспечивается уровнем значимости α = 0,05, определяющим критическое значение Fкр и вероятность ошибки первого рода при проверке гипотезы об отсутствии дрейфа по критерию Т-квадрат Хотеллинга.

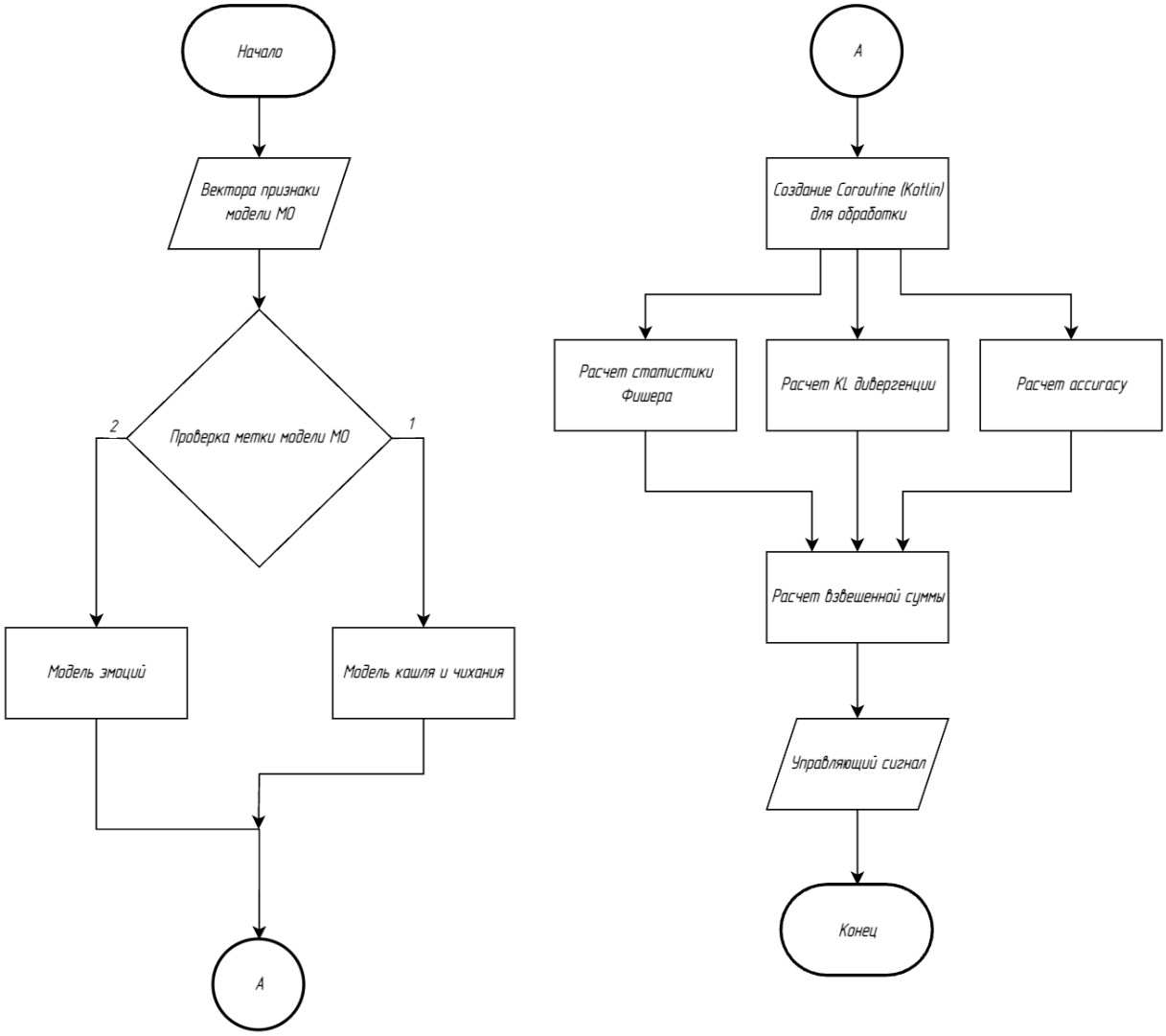


Рисунок 2.17 – Блок-схема алгоритма определения дрейфа данных

Для оценки устойчивости решений к заданным значениям весовых коэффициентов и порога выполнен анализ чувствительности интегрального критерия: значение каждого весового коэффициента варьировалось в диапазоне 0,2–0,8 с шагом 0,1 при фиксированных остальных параметрах, значение порога срабатывания – в диапазоне 0,3–0,7 с шагом 0,05. Для каждой конфигурации фиксировалось число обнаруженных дрейфов и число изменений решения относительно базовой конфигурации (равные веса, порог

0,5). По результатам анализа итоговое решение об обнаружении дрейфа не изменялось в 95% конфигураций (57 из 60 конфигураций), что подтверждает устойчивость детектора дрейфа к вариации весовых коэффициентов и порога в указанных пределах.

Итоговые параметры модели обнаружения дрейфа данных сведены в единую схему принятия решения. Эталонный набор наблюдений формируется на этапе настройки детектора по историческим данным, записанным до начала мониторинга в период корректной работы моделей; текущее распределение оценивается по данным скользящего окна размером от 15 до 30 последних наблюдений. Оценка распределений выполняется гистограммным методом на общей шкале из 10 интервалов равной ширины с нормировкой частот; нулевые вероятности интервалов обрабатываются аддитивным сглаживанием с параметром 0,01 и повторной нормировкой. Пороги срабатывания детектора составляют: критическое значение F статистики Фишера (уровень значимости $\alpha = 0,05$), порог KL-дивергенции 0,5, порог интегрального критерия 0,5; весовые коэффициенты интегрального критерия приняты равными $\alpha = \beta = \gamma = 1/3$. Верификация качества моделей машинного обучения выполняется на основе истинной разметки, формируемой самопометкой пользователя в интерфейсе мобильного приложения (подтверждение наблюдаемых событий). Статистическая значимость улучшений качества моделей после запуска процедуры федеративного обучения подтверждается парным критерием Уилкоксона для связанных выборок при уровне значимости $\alpha = 0,05$ (раздел 4.2).

Управляющий сигнал при наличии дрейфа активирует процедуру федеративного обучения, при этом модуль агрегации данных на время проведения процедуры федеративного обучения временно отключает от работы модули определения показателей, тем самым архитектура клиентского узла программного комплекса адаптируется к событию федеративного обучения.

Выбор комбинации «T^2 Хотеллинга + KL-дивергенция» для детектирования дрейфа обусловлен сопоставлением с известными альтернативами. Индекс стабильности популяции (PSI) основан на сравнении эмпирических распределений, представленных гистограммами, и не требует параметрических допущений, однако его результат существенно зависит от выбора числа и границ интервалов разбиения, а также от выравнивания сетки интервалов эталонного и текущего распределений; вычислительная сложность оценки составляет $O(n)$ на признак. Критерий Колмогорова–Смирнова сравнивает эмпирические функции распределения и не применим напрямую к многомерным векторам признаков: для учета всех измерений требуется множественная проверка гипотез с соответствующим снижением мощности вследствие поправок на множественное сравнение; расчёт статистики требует сортировки данных с трудоёмкостью $O(n \cdot \log n)$. Максимальное среднее расхождение (MMD) является непараметрическим ядровым критерием, однако оценка по выборкам объёма n требует попарного вычисления ядерных функций с вычислительной сложностью $O(n^2)$ без применения аппроксимаций, а результат чувствителен к выбору ядра и его параметра масштаба. Критерий Page–Hinkley и алгоритм ADWIN ориентированы на последовательное обнаружение изменений скалярного сигнала: Page–Hinkley, являющийся модификацией метода кумулятивных сумм (CUSUM), чувствителен к выбору порога и допускаемого уровня изменения, а ADWIN отслеживает изменение среднего в адаптивном скользящем окне и предоставляет формальные гарантии на характеристики обнаружения в зависимости от параметра допуска δ, однако ни один из них не учитывает корреляционную структуру многомерных признаков. Применяемый подход отслеживает изменение многомерных векторов признаков в скользящем окне (локальная квазистационарность) с учётом их ковариационной структуры, что обеспечивает контроль вероятности ошибки первого рода через уровень значимости статистики Т-квадрат, и дополняется оценкой расхождения распределений по KL-дивергенции. Вычислительные затраты предлагаемой

комбинации составляют $O(W \cdot p^2 + p^3)$ для расчёта статистики Т-квадрат по скользящему окну (W – размер окна, p – число признаков) и $O(W \cdot p + b \cdot p)$ для гистограммной оценки KL-дивергенции (b – число интервалов гистограммы), что является умеренным для суточного мониторинга в мобильной среде.

2.4.4 Модуль контроля ресурсов

Основываясь на идеях [59], было принято решение расширить предлагаемую авторами методику оптимизации энергопотребления. Предлагаемый модуль контроля ресурсов выполняет анализ потребляемых ресурсов мобильного устройства Android и сообщает модулю агрегации клиентского узла программного комплекса режимы работы сенсоров. Мониторинг было решено осуществлять по следующим параметрам:

- заряд аккумулятора мобильного устройства;
- нагрузка на центральный процессор (ЦП);
- температура батареи;
- наличие существенного перемещения;

Для определения ресурсного состояния мобильного вычислительного узла используется механизм сервиса в Android (Service) и менеджер аккумулятора BatteryManager. Сервис выполняет регистрацию компоненты широковещательного приемника BroadcastReceiver, который при получении намерения (Intent) распаковывает данные батареи. Для получения информации о батареи используется разрешение BATTERY_STATS и фильтр намерения ACTION_BATTERY_CHANGED. При получении данных выполняется чтение следующих полей:

- BatteryManager.EXTRA_LEVEL – уровень заряда;
- BatteryManager.EXTRA_SCALE – масштабирующий коэффициент для расчета уровня заряда;
- BatteryManager.EXTRA_TEMPERATURE – температура.

Процент ресурсного состояния мобильного вычислительного узла определяется по формуле (2.10):

$$batLvl = level \frac{100}{scale}, \tag{2.10}$$

где: level – прочитанный уровень заряда, scale – масштабирующий коэффициент.

Температура в градусах Цельсия определяется по формуле (2.11):

$$T = \frac{temp}{10}, \tag{2.11}$$

где: temp – температура.

Для определения нагрузки на ЦП используются модули android.os.Process и android.os.SystemClock. Идея расчёта использования ЦП основывается на сравнении затраченного процессорного времени на процесс и системного времени. Периодически вычисляя разницу между последовательными тактами процессора Process.getElapsedCpuTime и системными тактами SystemClock.elapsedRealtime, можно рассчитать загрузку процессора для каждого ядра. Важно отметить, что сам расчет времени выполняется в каждом модуле отдельно, а модуль контроля ресурсов лишь обрабатывает. Так как в данном случае интересуют модули, работающие в фоновом режиме, то мониторинг осуществляется для следующих модулей:

- модуль анализа эмоций;
- модуль анализа кашля и чихания;
- модуль анализа моторики и движения;
- модуль агрегации, который направляет данные в серверный узел.

Для определения существенного движения устройства используется тип измерений кинематических сенсоров называемый Sensor.TYPE_SIGNIFICANT_MOTION.

Стратегия регулировки работы частоты обработки данных:

- заряд аккумулятора < 30% / < 50%;
- температура аккумулятора > 50°C;
- высокая нагрузка на ЦП (> 70%);
- наличие существенного движения.

Наличие существенного движения включает обработку данных кинематическими сенсорами (модуль анализа моторики движения, модуль агрегации), модули при этом переходят в режим ожидания. При заряде

аккумулятора от 50% до 100% частота дискретизации кинематических сенсоров составляет 50 Гц, без задержек. При заряде аккумулятора от 30% до 50% частота дискретизации составляет 45 Гц, добавляется пауза в 25 мс при чтении данных с микрофона. При заряде аккумулятора от 0% до 30% частота дискретизации составляет 40 Гц, добавляется пауза в 50 мс при чтении данных с микрофона. Превышение температуры аккумулятора или высокая нагрузка на ЦП снижают частоту дискретизации до 40 Гц, отключают модули, использующие микрофон. Ввиду большого количества критериев, для измерений сенсоров используется вариант наибольшего ограничения из рассматриваемых при мониторинге.

Пороговое значение температуры аккумулятора 50 °C выбрано исходя из свойств литий-ионных элементов питания с обеспечением запаса безопасности. По данным производителей элементов, верхняя граница рабочего диапазона температур разряда литий-ионных аккумуляторов составляет 60 °C: при превышении этого значения работоспособность элемента не гарантируется, что сопровождается ускоренной деградацией ёмкости и повышением риска теплового разгона. Порог 50 °C рассматривается как аварийный верхний предел последней ступени реагирования, установленный с запасом 10 °C относительно гарантируемой производителями верхней границы 60 °C, а не уровень комфортной работы устройства: в штатных условиях температура корпуса мобильного устройства при работе и зарядке, как правило, не превышает 45 °C, поэтому в нормальной эксплуатации этот порог не достигается. Штатное регулирование частоты обработки данных осуществляется на более ранних ступенях ресурсных ограничений (заряд аккумулятора менее 50 % и 30 %, нагрузка на ЦП более 70 %), тогда как превышение температуры 50 °C свидетельствует об аномальном режиме работы и вызывает наиболее существенное снижение интенсивности обработки – вплоть до отключения модулей, использующих микрофон.

Обоснование остальных пороговых значений носит эксплуатационный характер и опирается на типовые сценарии использования мобильных

устройств. Пороговое значение заряда 50% выбрано из условия обеспечения непрерывного часового цикла мониторинга при среднем энергопотреблении приложения 50 мА·ч: при заряде ниже этого уровня снижается частота обработки кинематических сигналов и деактивируются второстепенные модули анализа без влияния на определение доминирующих показателей; порог 30% резервирует минимальный запас энергии для критически важных функций – передачи агрегированных данных на серверный узел и приёма управляющих сигналов, при достижении этого уровня основная фоновая аналитика приостанавливается. Порог нагрузки ЦП 70% соответствует границе, за которой операционная система Android ограничивает фоновое выполнение длительных процессов согласно своим правилам ограничения фоновой работы, вследствие чего дальнейшая обработка данных в фоне теряет эффективность; при превышении порога модель машинного обучения переходит в режим пониженной частоты обработки.

Для оценки устойчивости алгоритма управления ресурсами к выбору порогов выполнен анализ чувствительности: значения порогов заряда варьировались в диапазоне 40–60 с шагом 5%, температуры аккумулятора – в диапазоне 25–40 с шагом 5 °C, нагрузки ЦП – в диапазоне 60–80 с шагом 5%. Для каждой комбинации порогов фиксировались время работы аккумулятора и значения метрик качества определения показателей. Результаты показали, что варьирование порогов в пределах ±5 не приводит к статистически значимому изменению времени работы АКБ (различие менее 2%), при этом качество анализа сохраняется в заданных пределах, что подтверждает устойчивость выбранных порогов к малым отклонениям.

В рамках разработанного алгоритма управление режимами работы компонентов реализуется на основе принципа многокритериальной адаптивности. Под многокритериальной адаптивностью понимается способность системы динамически изменять конфигурацию активных модулей и параметры их функционирования на основе анализа вектора состояния устройства с учетом нескольких критериев оптимизации.

В качестве основных критериев в работе рассматриваются:

- энергопотребление мобильного устройства;
- точность определения показателей жизнедеятельности.

Формально задача управления может быть представлена как задача многокритериальной оптимизации, в которой выбирается конфигурация системы, минимизирующая обобщенную целевую функцию:

$$F = \omega_1 \cdot E + \omega_2 \cdot (1 - accuracy), \quad (2.12)$$

где: E – энергопотребление, $accuracy$ – метрика корректности, ω – весовые коэффициенты. В рамках реализации используется приближенное решение на основе эвристических правил и пороговых значений.

2.5 Выводы по главе

1. Была рассмотрена базовая архитектура программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека и определены основные проблемы архитектуры, а именно: проблема негибкой архитектуры ввиду монолитности и проблема устаревания моделей машинного обучения, возникающая при возникновении дрейфа данных. Следовательно, сделан вывод, что необходимо выйти из рамок трехуровневой архитектуры. Данные проблемы предлагается решить с помощью самоадаптирующейся микросервисной архитектуры с использованием федеративного обучения.

2. Дано описание особенностей микросервисной архитектуры, а именно: децентрализация управления данными, автономность, отказоустойчивость и масштабируемость; которые позволят стать инструментами для решения проблемы негибкой архитектуры. Таким образом, был сделан вывод, что микросервисная архитектура является мощным средством для создания гибких, масштабируемых систем, но её внедрение является стратегическим решением, которое оправдано для сложных, быстро развивающихся комплексов (таких как система медицинского мониторинга) и требует комплексного подхода, сочетающего правильные архитектурные паттерны (Circuit Breaker) и автоматизацию всех этапов жизненного цикла.

3. Проблему устаревания моделей машинного обучения было предложено решать путем внедрения механизма федеративного обучения моделей. В данном механизме при необходимости совершенствования моделей машинного обучения выполняется локальное обучение, результат которого агрегируется с глобальной моделью, которая также может быть обновлена другими клиентскими узлами, получая конечную обновленную модель. Таким образом, синергия федеративного обучения с микросервисной архитектурой создает каркас для разработки сложных распределенных комплексов, позволяющий реализовать перспективные программные комплексы неинвазивного мониторинга показателей жизнедеятельности человека.

4. Представленная на рисунке 2.11 общая архитектура программного комплекса является практически реализуемым решением для комплекса задач неинвазивного мониторинга. Она включает архитектурную модель программного комплекса, механизм оркестровки микросервисов, модуль контроля дрейфа данных, обеспечивающий обнаружение изменения распределений признаков моделей машинного обучения и запуск процедуры их адаптации, а также модуль контроля ресурсов, обеспечивающий формирование информационного статуса контролируемых показателей, бережное использование ресурсов мобильных устройств, защиту данных пользователя и возможность непрерывного развития.

ГЛАВА 3. МЕТОДИКИ НЕИНВАЗИВНОГО ОПРЕДЕЛЕНИЯ ПОКАЗАТЕЛЕЙ ЖИЗНЕДЕЯТЕЛЬНОСТИ ЧЕЛОВЕКА ПО ДАННЫМ ВИЗУАЛЬНЫХ, КИНЕМАТИЧЕСКИХ И АКУСТИЧЕСКИХ СЕНСОРОВ

Смартфоны пользователя обладают множеством сенсоров, которые позволяют использовать их как инструмент неинвазивного мониторинга показателей жизнедеятельности. В данной главе представлены исследования и часть алгоритмов программного комплекса, которые выполняют анализ данных, получаемых от следующих сенсоров: кинематические (акселерометр, гироскоп), акустические (микрофоны), визуальные (основная камера со вспышкой, фронтальная камера).

В рамках решения задачи определения показателей жизнедеятельности состояние пользователя рассматривается как состояние системы, описываемое вектором параметров:

$$S\colon [SpO_2, HR, RR \ldots], \tag{3.1}$$

где компоненты вектора соответствуют измеряемым физиологическим показателям.

Задача определения состояния формализуется как:

$$S = f(X), \tag{3.2}$$

где: S – состояние пользователя, f – алгоритм принятия решений. Таблица 3.1 задаёт систему принятия решений, определяющую отображение вектора состояния в дискретное множество состояний.

Таблица 3.1 – Таблица состояний алгоритма многопараметрического анализа

№	Состояние	Условия (пороговые значения)
1	Гипоксия	$SpO_2 < 94\%$
2	Дыхательная недостаточность	$94\% \leq SpO_2 < 95\%$ И RR > 25
3	Тахикардия	HR > 100
4	Брадикардия	HR < 60
5	Гипотензия	SBP < 90
6	Гипертензия	SBP ≥ 140 ИЛИ DBP ≥ 90
7	Шок	SBP < 90 ИЛИ (HR ≥ 100 И $SpO_2 < 94\%$)
8	Стресс	Негативные эмоции = да И HR > 100 И RR > 20
9	Физическое переутомление	Borg ≥ 15
10	Критическое утомление	Borg ≥ 18
11	Риск травмы	Падение = да И Сильный удар = да
12	Потеря сознания (подозрение)	Падение = да И Поза «лежит» = да И Отсутствие движения > 30 с
13	Неврологическое нарушение	(Тремор рук = да ИЛИ Тремор головы = да) И Нарушение координации = да

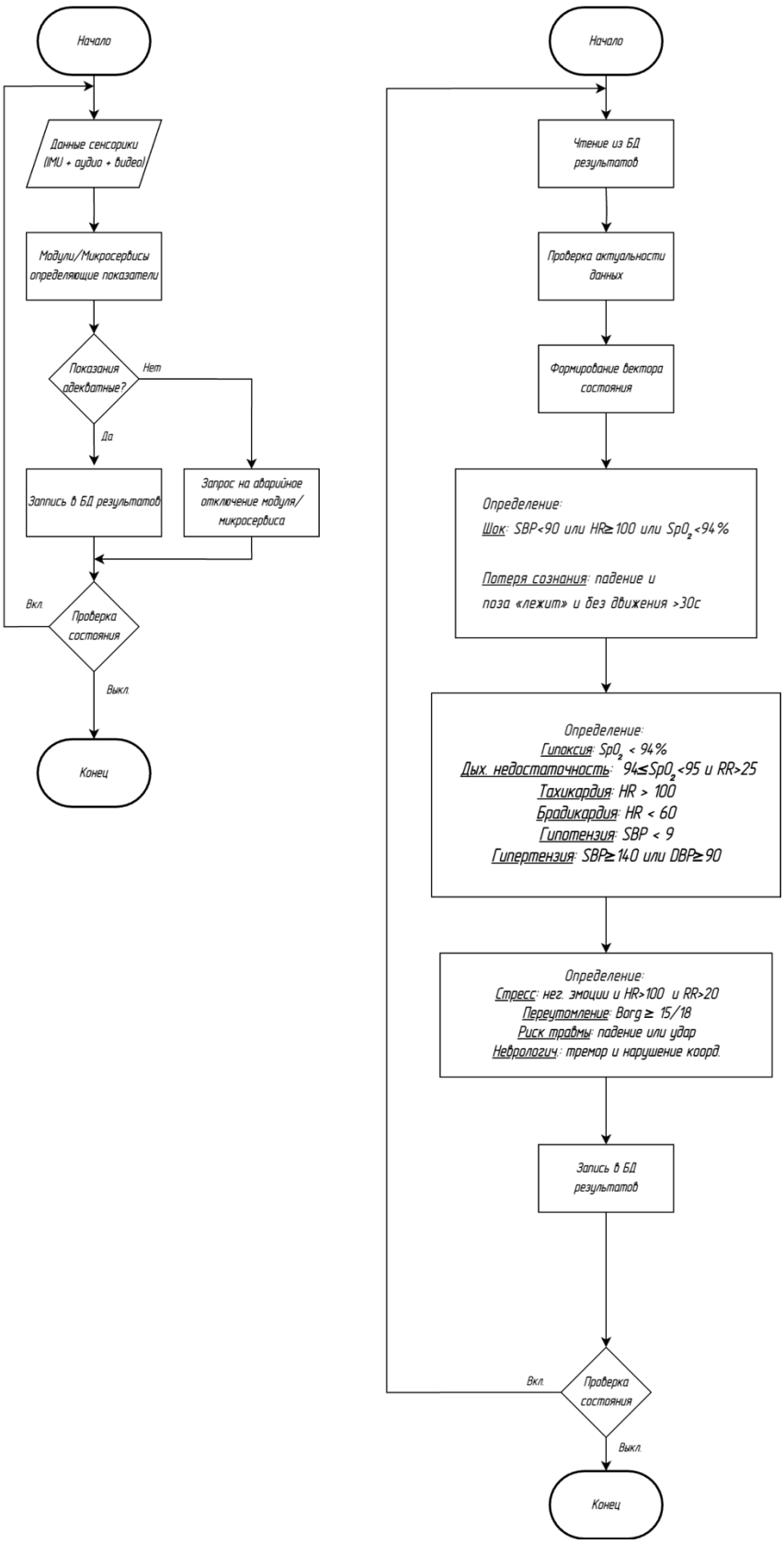


Рисунок 3.1 – Блок-схема алгоритма многопараметрического анализа

Алгоритм многопараметрического анализа реализует двухконтурную архитектуру обработки данных, обеспечивающую разделение процедур сенсорного сбора и алгоритмического анализа.

В рамках первого контура осуществляется непрерывная аккумуляция гетерогенной сенсорной информации, поступающей от инерциальных, акустических и оптических каналов измерения, с последующей первичной обработкой посредством оркестрованных модулей и фиксацией промежуточных результатов в хранилище. При выявлении недостаточной достоверности полученных показаний в течение четырех последовательных измерений – реализуется механизм обратной связи, инициирующий отключение соответствующего программного компонента.

Второй контур функционирует в дискретном режиме по событию и включает извлечение агрегированных данных, верификацию их актуальности и конструирование многомерного вектора состояния, параметризованного совокупностью параметров. Логический вывод основан на применении иерархической системы продукционных правил: на первом уровне выполняется проверка критического состояния, угрожающего жизнедеятельности; на втором – острые отклонения; на третьем – функциональные нарушения. Результат алгоритмического анализа формализуется как структурированная кортежная запись, содержащая идентификатор статуса, перечень активированных состояний, клиническую рекомендацию.

Асинхронная замкнутость контуров через единое хранилище обеспечивает слабую связанность архитектурных компонентов и повышает отказоустойчивость программного комплекса в условиях деградации отдельных сенсорных каналов.

3.1 Программно-алгоритмическая реализация методик анализа видеоданных для определения показателей жизнедеятельности

3.1.1 Определение пульса

Нарушения частоты сердечных сокращений (ЧСС), включая брадикардию и тахикардию, выступают значимыми индикаторами возможных патологий. Данные отклонения могут предшествовать синкопальным состояниям, указывать на наличие аритмий, а также ассоциированы с

повышенным риском развития инфаркта миокарда, острой сердечной недостаточности или ишемического инсульта. Учитывая, что сбои в работе сердца чреваты фатальным исходом, регулярный мониторинг ЧСС становится простым, но жизненно необходимым инструментом для раннего выявления угроз. Своевременное обращение к специалисту на основе данных мониторинга – ключ к предотвращению или минимизации вреда здоровью. В связи с этим актуальны современные методики контроля этого параметра.

Стремясь создать более портативную аппаратную платформу для регистрации ЧСС, исследователи П.Т. Тунггал, А. Латиф и Исванто [60] разработали миниатюрный сенсор пульса. Устройство интегрирует ИК-излучатель, фотодиод, усилительный каскад, фильтр низких частот (ФНЧ), одновибратор и микроконтроллер. Принцип действия основан на пропускании ИК-излучения через палец с последующей фиксацией фотодиодом. Слабый фотодиодный сигнал усиливается (неинвертирующий усилитель), очищается от помех фильтром нижних частот и преобразуется одновибратором в счетные импульсы. Микроконтроллер AVR ATMEGA8, запрограммированный для подсчета импульсов на своем входном контакте в течение минуты, вычисляет ЧСС и выводит значение на ЖК-дисплей. Алгоритм также детектирует брадикардию и тахикардию. Работоспособность системы подтверждена испытаниями на добровольцах.

Основу предложенной в работе [61] методики мобильного мониторинга составляет использование камеры смартфона и его светодиодной вспышки в качестве сенсора для регистрации фотоплетизмограмм и измерения ЧСС. Авторы указывают на ограниченность существовавших на тот момент систем [62], где смартфоны служили лишь для отображения результатов и дополнительных расчетов. Новизна подхода состоит в использовании штатных компонентов мобильного телефона для выполнения сенсорных измерений. Методика предполагает прижатие пальца к объективу камеры со вспышкой: вспышка создает постоянную подсветку, а камера записывает видеопоследовательность. Анализу подвергается усредненная яркость

каждого кадра. Сердечные сокращения вызывают пульсацию крови, что приводит к периодическому изменению объема крови в капиллярах кончика пальца. Это, в свою очередь, модулирует интенсивность проходящего света и вызывает колебания яркости изображения. Вычисление частоты сердечных сокращений производится путем идентификации пиковых значений на графике временных изменений яркости с последующим расчетом по формуле:

$$HR = \frac{60*n}{t}, \quad (3.3)$$

где n – количество локальных экстремумов, соответствующих притоку крови к капиллярам пальца, t – размер временного окна в секундах. Для верификации точности разработанной методики исследователи провели сравнительный анализ между фактическими показателями ЧСС и данными, полученными с использованием смартфонной камеры (25 кадров в секунду). Результаты эксперимента показали:

- при оптимальных условиях освещения: среднее отклонение 4,13%, предельная погрешность 8,11%;
- при недостаточной освещенности: среднее отклонение 4,67%, максимальная погрешность 10,13%.

Разработанное решение представляет собой экономичную платформу формирования информационного статуса контролируемых показателей, предназначенную для предварительного мониторинга и поддержки принятия решений и доступную для самостоятельного использования населением.

Автор работы [6], Викрам Чандрасекаран, предлагает развитие методики [63] со следующими особенностями:

- локализация анализа: обработке подвергается единственный квадрант изображения, демонстрирующий наиболее выраженные флуктуации яркости;
- цветовая селекция: из RGB-представления области интереса извлекаются исключительно значения красного канала (R), на основе которых рассчитывается усредненная интенсивность;

- фильтрация сигнала: временной ряд интенсивностей сглаживается с применением фильтра, скользящего среднего для выделения достоверных экстремумов и подавления шумовых артефактов;
- расчет ЧСС: на основе отфильтрованных экстремальных значений по формуле (3.3) определяется частота сердечных сокращений;
- оптимизация окна: установлена зависимость точности измерений от длительности временного окна (положительная корреляция). Для компенсации данного эффекта и сокращения необходимого времени записи предложен алгоритм калибровки.

Алгоритм калибровки рассчитывает калибровочный коэффициент по следующей формуле:

$$Ex = t - N * pk2pk, \tag{3.4}$$

где: t – размер временного окна в секундах, N – число пиков, pk2pk – время между двумя соседними пиками.

Далее, проводится подстройка:

1. Если значение калибровочного коэффициента больше или равно pk2pk, значение N корректируется по формуле:

$$N = N + 1. \tag{3.5}$$

2. Если значение калибровочного коэффициента больше или равно pk2pk / 2, значение N корректируется по формуле:

$$N = N - (pk2pk - Ex) * \frac{60}{t}. \tag{3.6}$$

Затем расчет частоты сердечных сокращений осуществляется на основании формулы (3.1). Проведенные эксперименты по оценке точности разработанного алгоритма продемонстрировали среднюю ошибку измерений в пределах 2% при использовании временного окна длительностью 15 секунд.

Авторы[7] (Хименес, Парнанди, Гутьеррес Осуна) используют подход, аналогичный [6, 63]: ЧСС регистрируется камерой и вспышкой iPhone 4. Эксперимент включал:

- контролируемое дыхание: 6/10/12 циклов/мин (управлялось звуковыми метрономами);
- спонтанная активность: обычное дыхание, упражнения сидя, ходьба. Видео (разрешение 1280x720 с частотой кадров 29,97 кадров в секунду) обрабатывалось в MATLAB: кадры извлекались (VideoReading), анализировался исключительно зеленый RGB-канал. Средняя G-интенсивность формировала PPG-сигнал, пики сигнала идентифицировались с использованием алгоритма. Вариабельность сердечного ритма (ВСР) вычислялась как временной интервал между последовательными пиками. На основе полученных значений ВСР производился расчет частоты сердечных сокращений (ЧСС) с применением формулы:

$$\text{ЧСС} = \frac{60}{BCP} \tag{3.7}$$

Синхронизация и повторная дискретизация сигналов от смартфона и биопояса обеспечили сопоставимую частоту отсчетов. Анализ показал высокую корреляцию между ЧСС, определенной камерой телефона, и данными коммерческого сенсора, однако с фиксированной временной задержкой мобильного сигнала. Объяснение авторов заключается в том, что измерение на пальце отражает периферический пульс, которому требуется время для распространения по сосудам от сердца. Грудной сенсор, будучи ближе к источнику пульсации (сердцу), детектирует события раньше. Точность определения частоты сердечных сокращений посредством камеры мобильного устройства оценивалась по четырем параметрам: средней абсолютной ошибке, стандартному отклонению погрешности, среднеквадратичному отклонению и коэффициенту корреляции. Валидация проводилась на трех наборах данных (управляемые/неуправляемые условия) при синхронной записи телефонной камерой и эталонным сенсором.

Управляемые испытания включали фиксированные частоты дыхательного цикла: 0,1 Гц (6 циклов/мин), 0,2 Гц (12 циклов/мин) и 0,16 Гц (10 циклов/мин).

В работе [64] был использован альтернативный метод фиксации ЧСС посредством фотоплетизмографии с применением камеры и вспышки смартфона, однако авторы не предоставили детали реализации алгоритма. В качестве эталонных данных использовались усредненные минутные показания ЧСС, полученные с помощью биоимпедансного кардиографа BioZ с применением 4-тактного скользящего усреднения. Данные со смартфона и пульсоксиметра Nonin агрегировались и сопоставлялись с синхронными ЭКГ-записями. Для анализа применялись коэффициент корреляции Пирсона (r) и стандартная ошибка оценки (SEE). Учитывая ожидаемую тесную взаимосвязь при измерении идентичных физиологических сигналов, согласованность показаний трех устройств оценивалась методом Бланда-Альтмана для повторных измерений (с 95% ДИ). Наблюдалось расширение диапазона согласия в условиях чтения текста вслух и видеоигр по сравнению с состоянием покоя. Результаты показали:

- исключительную корреляцию между смартфоном и ЭКГ BioZ ($r \geq 0,99$) при SEE ≤1 уд/мин во всех сценариях;
- сопоставимо высокую согласованность с Nonin 9560BT ($r \geq 0,99$, SEE ≤ 2,09 уд/мин).

Авторы подчеркивают, что высокая корреляция между устройствами не означает их клинической согласованности, поскольку даже линейно зависимые, но систематически смещенные данные могут демонстрировать статистически значимые коэффициенты корреляции (r-значения).

Разработанный в результате алгоритм контроля частоты пульса представлен на рисунке 3.2, где: f_{max} – максимальное значение и модуля спектра, $size$ – размер массива, sampFreq – частота семплирования.

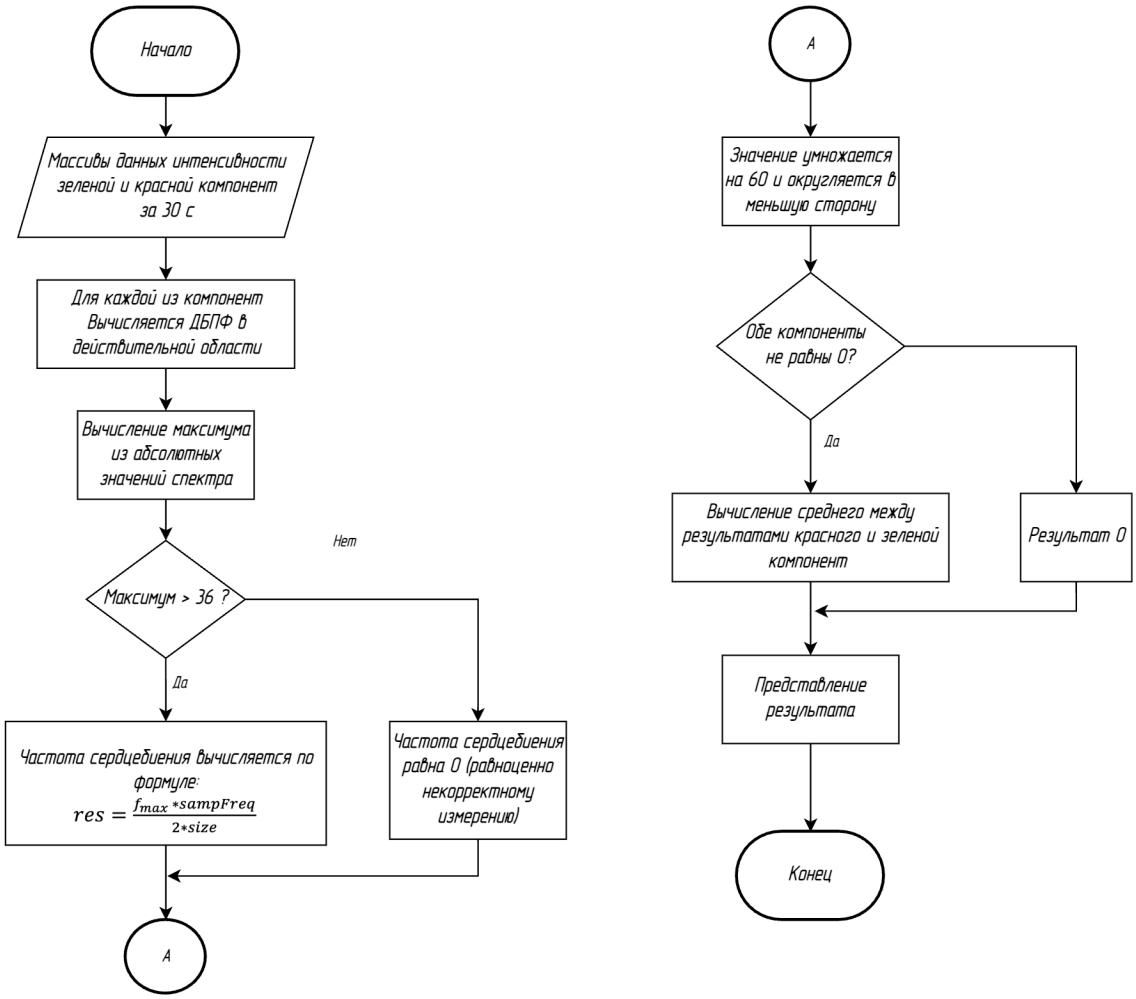


Рисунок 3.2 – Блок-схема алгоритма контроля частоты пульса

3.1.2 Определение тремора головы

Исследованию тремора – одному из ключевых симптомов неврологических расстройств – и ассоциированных состояний посвящено множество исследований.

В [65] работе разработана система компьютерного зрения для анализа осцилляций головы при цервикальной дистонии. Система фокусируется на объективной оценке тяжести тремора, вычисляя пиковую мощность, частоту и доминирующий вектор движений. Корреляционный анализ по Спирмену связывает эти параметры с клинической картиной. Точность решения (СКО = 0,89) достигается за счет трекинга угловых параметров головы (тангаж, крен, рыскание) методом [66]. Стоит также отметить, что в работе [67] разработан альтернативный подход на основе OpenCV, где углы ориентации головы

(крен, тангаж, рыскание) извлекаются напрямую с помощью алгоритмов детектирования позы.

В работе [68] обоснована применимость инерциальных сенсоров (акселерометр/гироскоп) для определения тремора, причем гироскоп продемонстрировал преимущество над акселерометром. Протокол измерений включал:

- фиксацию сенсора на теменной области;
- регистрацию данных в позициях, регламентированных шкалой TETRAS (1 минута на позицию);
- расчет диагностических метрик: среднее/максимальное смещение и среднее/максимальное вращение;
- балльную оценку тяжести тремора на основе этих показателей. Корреляция с клиническим рейтингом достигла 0,853.

В [69] авторами представлена архитектура машинного обучения GTSN (Global Temporal Shift Network) для автоматизированной оценки по шкале MDS-UPDRS. Основанная на остаточных сетях (ResNet), модель анализирует видеопотоки для выявления тремора различных сегментов тела, включая челюстную область. При диагностике тремора челюсти система продемонстрировала точность 89% (метрика accuracy).

В работе описана методика [70] обнаружения тремора рук с использованием веб-камер (частота съемки: 30 Гц). Протокол предусматривает строгие условия: фиксированная дистанция до камеры, нулевой угол наклона, стандартизированная позиция руки, контрастный фон и контролируемое освещение. Стоит обратить внимание, что в публикации отсутствуют результаты валидации системы и описание алгоритмической реализации экспресс-диагностики.

Современные исследования подтверждают принципиальную возможность обнаружения тремора, включая головной, посредством видеоанализа. Следует отметить, что прямая диагностика тремора головы остается слабо изученной областью. Большинство существующих методов

используют инерциальные сенсоры (акселерометры и гироскопы). В связи с этим принято решение разработать алгоритм детектирования тремора головы на основе видеоданных фронтальной камеры смартфона на Android.

Частота тремора головы характеризуется диапазоном 3–18 Гц [71]. Принимая во внимание стандартную частоту съемки фронтальных камер (≥30 fps), выбрано значение 30 кадров/с. Это оптимизирует охват устройств без потери информативности. Телефонное видео по своей природе – дискретная репрезентация визуальных данных через упорядоченные кадры. В соответствии с теоремой Котельникова-Найквиста [72]:

$$f_{\text{д}} \geq 2 * f_{\text{в}}, \tag{3.8}$$

где: $f_{\text{д}}$ – частота дискретизации, которого достаточно для полного восстановления аналогового сигнала из дискретного сигнала, $f_{\text{в}}$ – верхняя частота сигнала. Согласно данной теореме, частота дискретизации 30 Гц обеспечивает восстановление сигналов с верхней границей 15 Гц. Таким образом, предлагается алгоритм, анализирующий частотный диапазон головного тремора в пределах 3–15 Гц. Эффективность предложенной методики зависит от соблюдения следующих условий:

- полная неподвижность позы и отсутствие движений у человека;
- лицо должно быть полностью открыто (без волос, закрывающих лицо, и без очков);
- надежная фиксация смартфона во время видеозаписи (например, с использованием штатива);
- отключение функции стабилизации изображения на смартфоне для исключения искажений измеряемых данных;
- проведение измерений при достаточном освещении для обеспечения четкой видимости лица фронтальной камерой.

На рисунке 3.3 изображена блок-схема алгоритма определения тремора головы [73]. Данное решение реализовано как программный компонент мобильного приложения – модуль определения тремора головы программного комплекса неинвазивного мониторинга показателей жизнедеятельности

человека. Кодовая база алгоритма написана на языке программирования Kotlin [74].

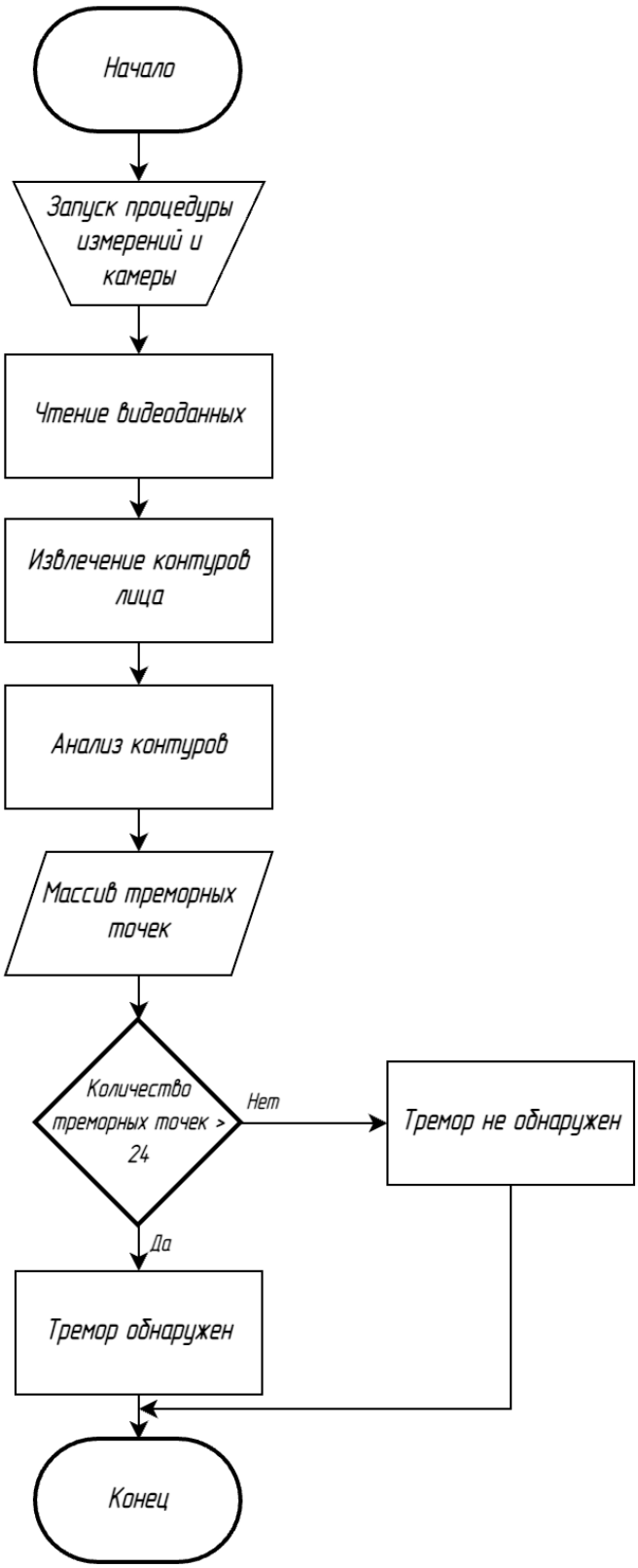


Рисунок 3.3 – Блок-схема алгоритма определения тремора головы

Система определения тремора головы, реализованная в модульном исполнении, включает следующие компоненты: модуль инициализации измерений, модуль чтения видеопотока, модуль экстракции лицевых контуров, модуль анализа контурных данных, модуль оценки результатов в балльной системе и модуль визуализации результатов. Перед пациентом ставится задача проведения серии измерений через фронтальную камеру смартфона на базе ОС Android для выявления тремора головы при строгом соблюдении ранее указанных требований к процедуре.

Блок запуска измерений находится в режиме ожидания до момента, когда пользователь инициирует процедуру измерений нажатием кнопки, показанной на рисунке 3.4. В течение последующих 10 секунд пациенту необходимо расположить лицо в области обзора фронтальной камеры смартфона и ожидать начала сбора данных при соблюдении указанных ранее требований.

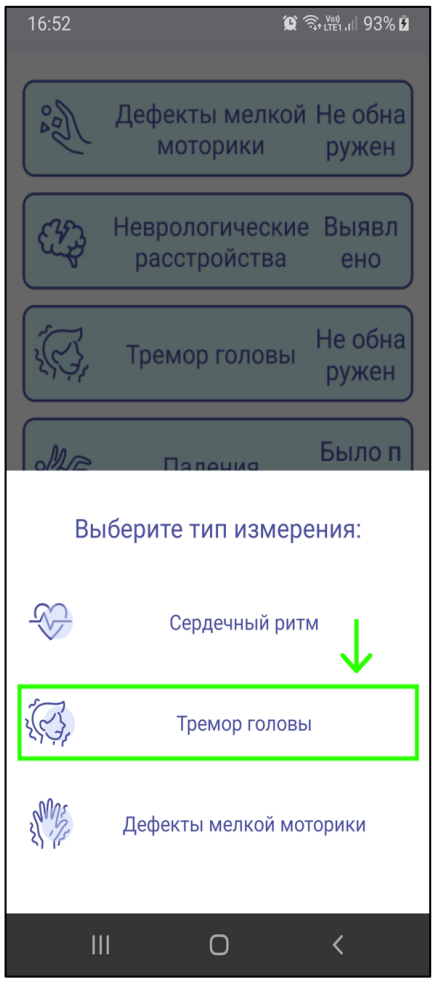


Рисунок 3.4 – Кнопка запуска процедуры измерений

Блок чтения видеоданных инициирует процесс записи видеопотока с использованием библиотеки CameraX. Продолжительность записи составляет 10 секунд. Полученные видеоданные передаются на последующие стадии обработки, где реализуется основная логика алгоритма. Сам процесс обработки визуально сопровождается шкалой прогресса обработки для оповещения пользователя и обеспечивает возможность отмены процедуры в случае необходимости (рисунок 3.5).

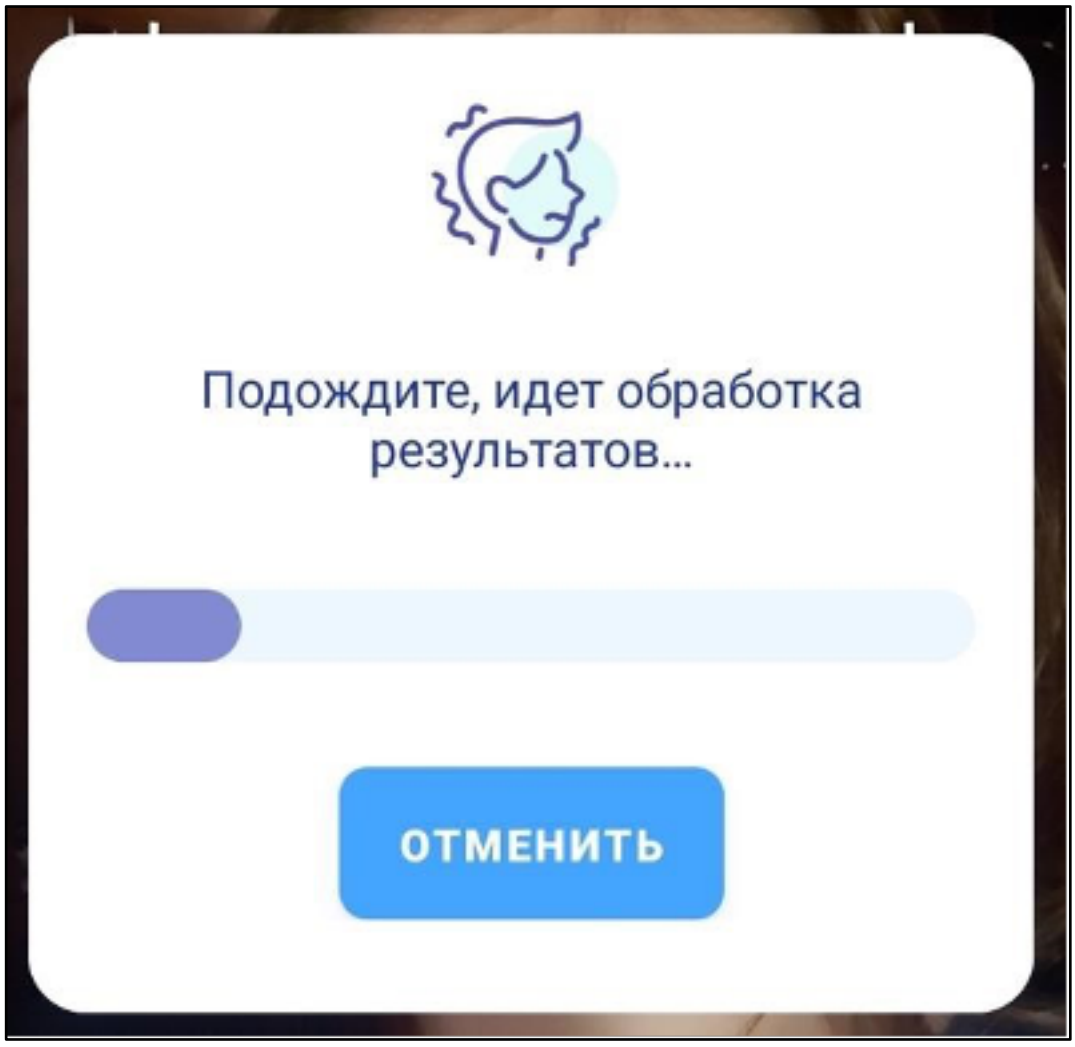


Рисунок 3.5 – Индикатор процесса определения тремора головы алгоритмом

Блок извлечения контуров лица осуществляет распознавание контуров лица в видеозаписи с применением машинного обучения. Алгоритм работает по следующим шагам:

- извлечение данных через библиотеку JCodec [75] с использованием метаданных видео;
- расчет частоты кадров из полученных метаданных;

- последовательный анализ кадров моделью машинного обучения ML Kit [76] (специализированный фреймворк Android для мобильного компьютерного зрения), которая идентифицирует контуры лица согласно Таблице 3.2.

Для определения тремора головы используется строго определенный класс контуров «Овал лица», представленный набором из 36 точек с координатами в формате чисел с плавающей запятой. Координаты предварительно нормируются относительно разрешения камеры. Результатом работы модуля является массив контурных точек овала лица, который передается в модуль анализа контуров.

Таблица 3.2 – Классы контуров лица модели машинного обучения ML Kit [76]

Тип контура лица	Количество точек, формирующих контур
Овал лица	36
Верхняя часть левой брови	5
Нижняя часть левой брови	5
Верхняя часть правой брови	5
Нижняя часть правой брови	5
Левый глаз	16
Правый глаз	16
Центр левой щеки	1
Центр правой щеки	1
Верхняя часть верхней губы	11
Нижняя часть верхней губы	9
Верхняя часть нижней губы	9
Нижняя часть нижней губы	9
Переносица	2
Нижняя часть носа (ноздри)	3

Блок анализа контуров лица осуществляет обработку полученных контуров лица. Для каждой точки контура из массива, содержащего все обнаруженные контуры, в цикле выполняется вычисление расстояний между точками по формуле:

$$r = \sqrt{\left(pt_{x\,j-1} - pt_{x\,j}\right)^2 + \left(pt_{y\,j-1} - pt_{y\,j}\right)^2}, \quad (3.9)$$

где: pt – точка контура лица, j – индекс элемента списка, хранящего все распознанные контуры лица, x – ось X изображения, y – ось Y изображения. Затем производится вычисление медианного значения для полученных расстояний и отклонений от медианного значения. Полученные отклонения интерпретируются как сигнал, который подвергается частотному анализу посредством быстрого преобразования Фурье (БПФ) в области действительных чисел, основанного на дискретном преобразовании Фурье, задаваемом формулой:

$$S_k = \sum_{i=0}^{N-1} s_i * e^{-\frac{2\pi * i}{N} ki}, \quad (3.10)$$

где: S_k – k-ый элемент массива частотного представления дискретного сигнала, s_i – i-ый элемент массива временного представления дискретного сигнала, N – количество отсчетов дискретного представления сигнала, i – индекс, применяемый для итерации по массиву дискретного представления сигнала во временной области, k – индекс, применяемый для итерации по массиву дискретного представления сигнала в частотной области. Затем осуществляется нормализация значений амплитудно-частотной характеристики (АЧХ) спектра для каждой точки относительно максимального значения. Выполняется подсчет количества точек контура овала лица, у которых амплитуда в частотном диапазоне тремора (3–15 Гц) превышает пороговое значение 0,8. Полученное количество точек передается в модуль оценки результатов.

Блок подсчета баллов на основе полученных данных определяет наличие тремора головы. Принцип работы основан на сравнении с пороговым значением: если количество точек, соответствующих критериям тремора,

составляет не менее 25, фиксируется наличие тремора головы, в противном случае – его отсутствие. Результат работы блока передается в блок представления результата.

Блок представления результата отображает итоги работы алгоритма пользователю, а также инициирует передачу результатов на серверный узел программного комплекса.

3.2 Программно-алгоритмическая реализация методик анализа кинематических данных для определения показателей жизнедеятельности

3.2.1 Определение положения тела

Классификация позы человека осуществлялась посредством обработки сигналов следующих кинематических сенсоров смартфона: 3-осевой акселерометр, 3-осевой линейный акселерометр, 3-осевой гироскоп. В работе [77] разработана рекуррентная модель нейронной сети (рисунок 3.6), идентифицирующая позы по данным указанных сенсоров. Обучение модели проводилось с применением фреймворка TensorFlow и библиотеки Sklearn на наборе данных [78], итоговая точность классификации достигла 93,57%.

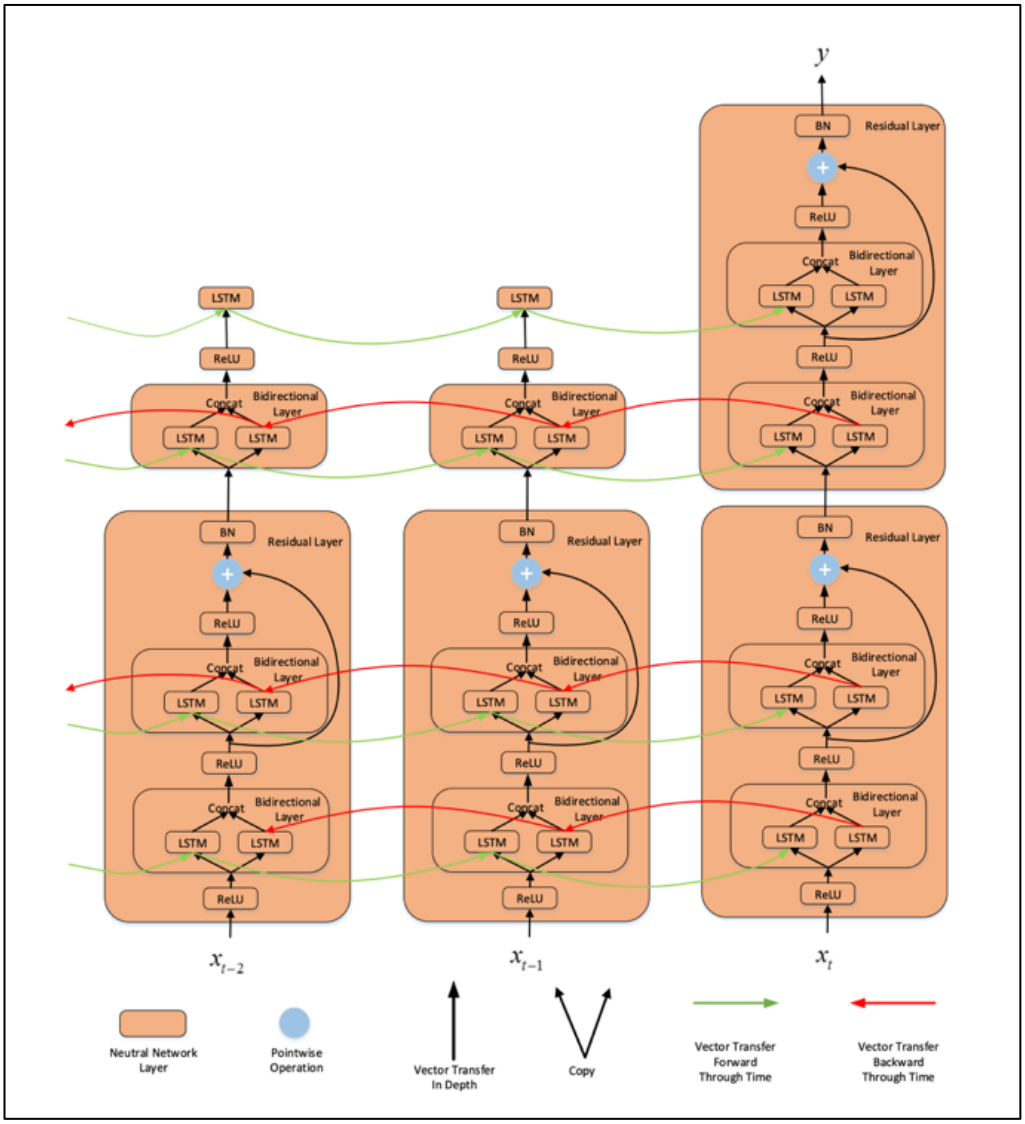


Рисунок 3.6 – Архитектура нейронной сети [79]

Алгоритм определения позы человека представлен на рисунке 3.7:

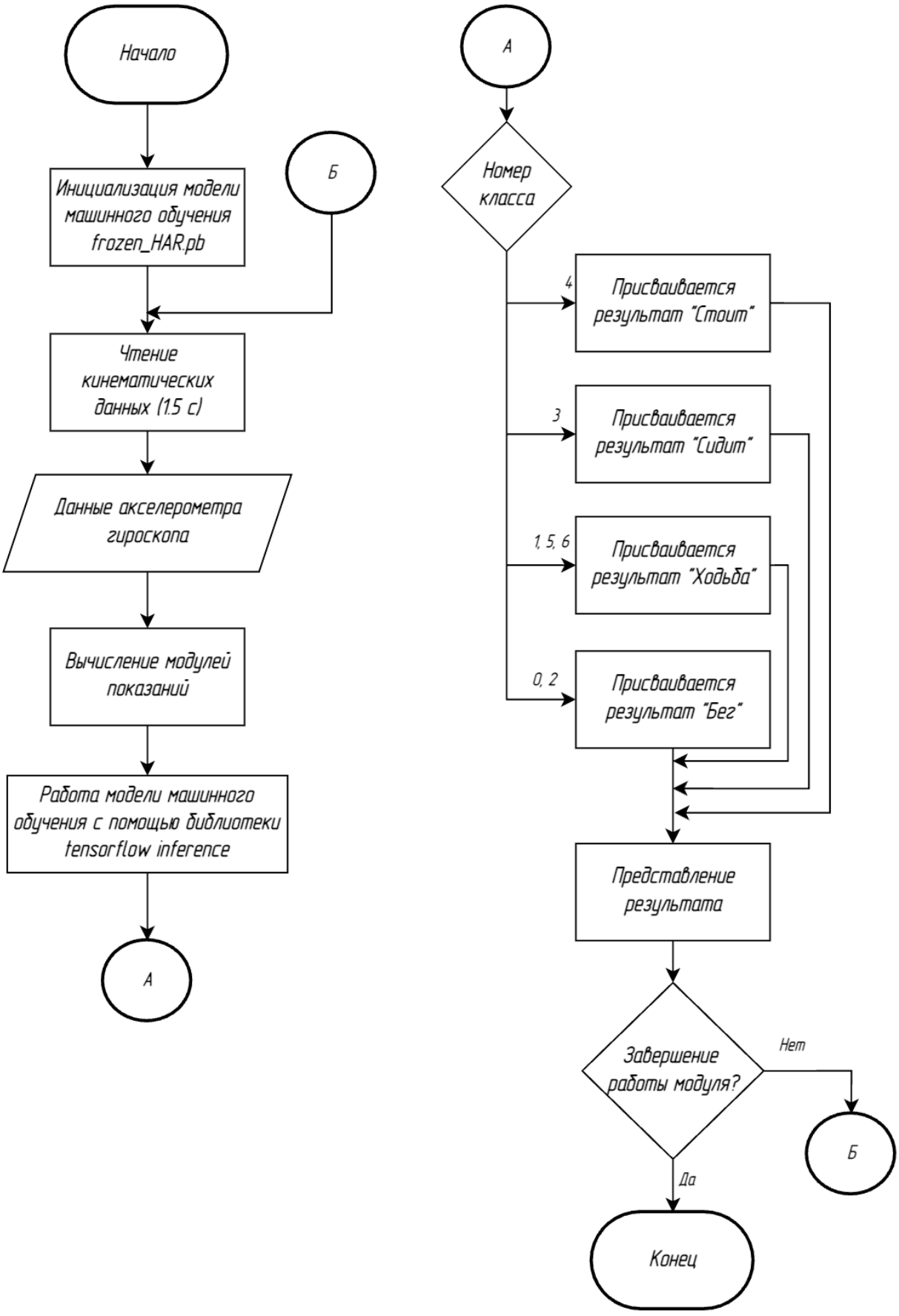


Рисунок 3.7 – Блок-схема алгоритма определения положения тела

3.2.2 Определение мелкой моторики

Исследования в сфере мелкой моторики в основном сфокусированы на терапевтических методиках, направленных на её нормальное развитие у человека [80-81]. Диагностика нарушений мелкой моторики изучена достаточно глубоко. Значительное число научных работ посвящено анализу двигательных расстройств, сопровождающих различные отклонения. В частности, в работе [82] подробно анализируются нарушения мелкой моторики, возникающие при центральных и периферических парезах верхних конечностей. Утрата навыков мелкой моторики у таких пациентов ограничивает предметно-манипулятивную деятельность и снижает социально-профессиональную адаптацию. В качестве инструментов оценки функциональных возможностей кисти авторы предлагают динамометрию и гониометрию (измерение диапазона движений в суставах). Патент[83] раскрывает технологию оценки двигательной функции через анализ видеопотока с частотой 120 Гц, фиксирующего движения кисти с 23 светоотражающими полусферическими маркерами; метод в основном ориентирован на определение объёма движений кисти. Авторами [84] была создана пространственно-временная архитектура нейронной сети для точной регрессии 3D-координат суставов руки на основе инфракрасной видеозаписи. Статистическая значимость метода (коэффициент Пирсона) находилась в диапазоне 0,82–0,86. В исследовании [85] авторы применили технологию магнитного зондирования для оценки мелкой моторики у детей. Участники эксперимента выполняли задание по попеременному касанию большим пальцем указательного пальца на обеих руках в течение 10-секундного интервала. Для исследования возрастных особенностей моторного развития использовался двухфакторный дисперсионный анализ (ANOVA). По результатам исследования авторы пришли к заключению, что данная методика обладает значительным потенциалом для диагностики моторных нарушений. В работе [86] рассматривается использование методов анализа изображений для оценки выполнения детьми тестов на мелкую моторику. В исследовании

применялись следующие алгоритмы машинного обучения: сверточные нейронные сети (CNN), метод k-ближайших соседей (k-NN), метод опорных векторов (SVM) и алгоритмы деревьев решений. Наивысшая точность классификации достигла 82,5% среди протестированных моделей. В публикации [87] представлен комплекс ритмичных упражнений, выполняемых под счет, направленных на развитие связной речи и мелкой моторики у детей. Программа включает: сжимание кисти в кулак; сведение и разведение пальцев на плоскости стола; поочередное формирование колец разными пальцами при раскрытой ладони; последовательное касание большим пальцем всех остальных пальцев (одной и двумя руками одновременно); синхронное отведение на обеих руках вторых и третьих, затем вторых и пятых пальцев (по демонстрационному образцу и вербальной инструкции); одновременное наложение вторых пальцев на третьи на обеих руках и обратная последовательность. К диагностическим критериям нарушений отнесены: нарушение синхронности работы рук, отклонения в скорости выполнения упражнений (брадикинезия/тахикинезия) и ограниченная амплитуда движений.

Проведенный анализ исследований подтверждает значительный потенциал технических средств для диагностики состояния мелкой моторики. Вместе с тем, решения для выявления её нарушений с использованием сенсорики мобильных устройств остаются недостаточно разработанными. Следует отметить, что современные диагностические методики преимущественно основаны на выполнении пациентами специализированных упражнений, по результатам которых производится оценка уровня развития моторных функций.

Современные акселерометры в мобильных устройствах способны обеспечивать высокую частоту дискретизации, достигающую значений около 100 Гц. Однако поддержка столь высоких частот реализована не на всех моделях устройств. Согласно открытой официальной документации Google [88], разработка приложений для работы с акселерометрами на платформе

Android имеет определенные особенности. Основное ограничение состоит в том, что разработчик не может произвольно задавать частоту опроса сенсора, а должен выбирать из четырех предустановленных вариантов частоты считывания данных. Для управления частотой считывания разработчику доступны четыре предустановленные опции:

- SENSOR_DELAY_NORMAL: режим по умолчанию с интервалом дискретизации 200 000 мкс (микросекунд);
- SENSOR_DELAY_UI: предназначен для визуализации данных в реальном времени в графическом интерфейсе. Обеспечивает задержку 60 000 мкс, что эквивалентно частоте ~16,6 Гц;
- SENSOR_DELAY_GAME: предназначен для игровых приложений. Интервал считывания: 20 000 мкс (соответствует 50 Гц);
- SENSOR_DELAY_FASTEST: максимально возможная частота выборки с нулевой задержкой (0 мкс). Также стоит отметить, что верхняя частота ничем не ограничивается и зависит от аппаратного сенсора, в случае использования SENSOR_DELAY_FASTEST, что создает необходимость выполнения процедуры передискретизации сигнала. Эта проблема решается путем применения, например, сглаживающего фильтра Чебышева, АЧХ которого представлена в формуле (3.11):

$$|H(j\omega)| = \frac{1}{\sqrt{1+\varepsilon^2 T_n^2\left(\frac{\omega}{\omega_c}\right)}}, \quad (3.11)$$

где: ω – частота сигнала, ω_c – частота среза фильтра, ε – показатель пульсаций, $T_n(x)$ – многочлен Чебышева n-го порядка. С учетом вышеописанного, принято решение выполнять чтение данных с частотой 50 Гц, используя опцию SENSOR_DELAY_GAME.

Представленная в исследовании методика детекции нарушений мелкой моторики представляет собой автономное решение, функционирующее непосредственно на смартфоне пользователя с возможностью интеграции в диагностико-аналитические системы. В основе методики лежит анализ данных акселерометра мобильного устройства, а программная реализация

выполнена на языке Kotlin для платформы Android. Решение оформлено в виде программного модуля, интегрируемого в мобильное приложение комплекса неинвазивного мониторинга показателей жизнедеятельности человека. Архитектурная схема решения представлена на рисунке 3.8.



Рисунок 3.8 – Структурная схема модуля определения наличия дефектов мелкой моторики

Модуль определения наличия дефектов мелкой моторики, реализованный в виде программного компонента, включает следующие блоки: инициализации измерений, подготовки пациента к процедуре, регистрации показаний акселерометра, анализа данных и визуализации результатов. Для выявления возможных нарушений пациенту предлагается выполнить тестовое задание по удержанию смартфона согласно следующему протоколу:

- запустить приложение и выбрать соответствующий пункт меню для начала измерений;
- занять положение стоя или сидя;
- расположить вытянутую руку со смартфоном перед собой на уровне плеча;
- ориентировать устройство перпендикулярно поверхности пола экраном к себе, удерживая его в ладони (исходное положение);
- зафиксировать исходное положение в течение 3 секунд;
- сохранять неподвижность в течение 10-секундного периода измерений.

Блок запуска процедуры измерений находится в режиме ожидания нажатия пользователем соответствующей интерфейсной кнопки графического интерфейса смартфона для запуска диагностической процедуры (см. рисунок 3.9).

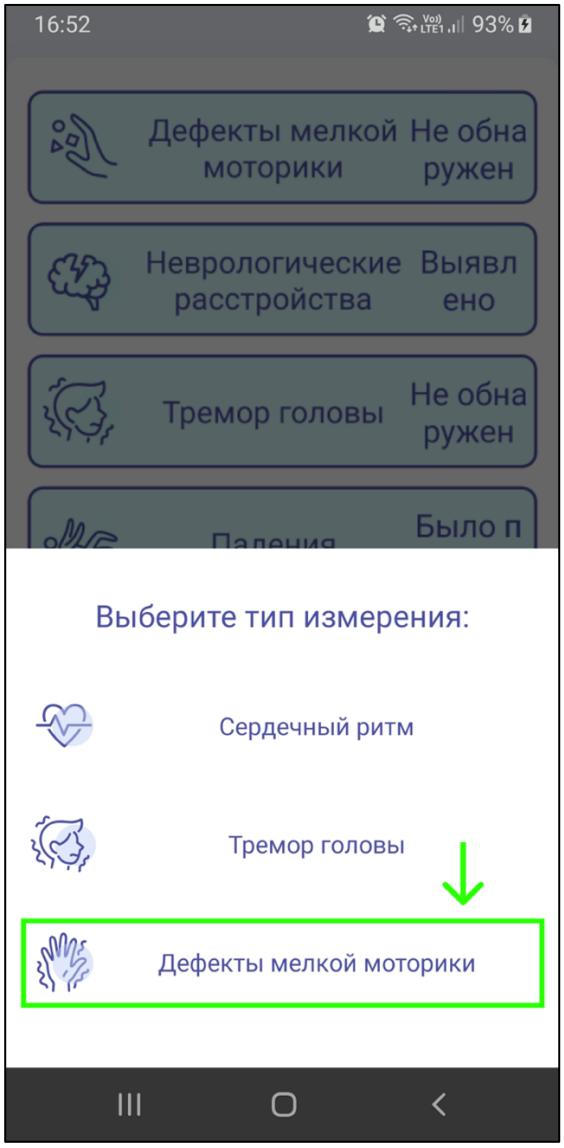


Рисунок 3.9 – Кнопка запуска процедуры измерений

Блок подготовки пациента обеспечивает трехсекундную паузу для принятия корректного исходного положения перед началом диагностической процедуры, в соответствии с вышеописанным протоколом. Параллельно блок регистрации данных осуществляет считывание показаний акселерометра и передает полученную информацию модулю анализа.

Блок анализа выполняет обработку результатов измерений в соответствии с приведенным алгоритмом детекции нарушений мелкой моторики, блок-схема которого представлена на рисунке 3.10.

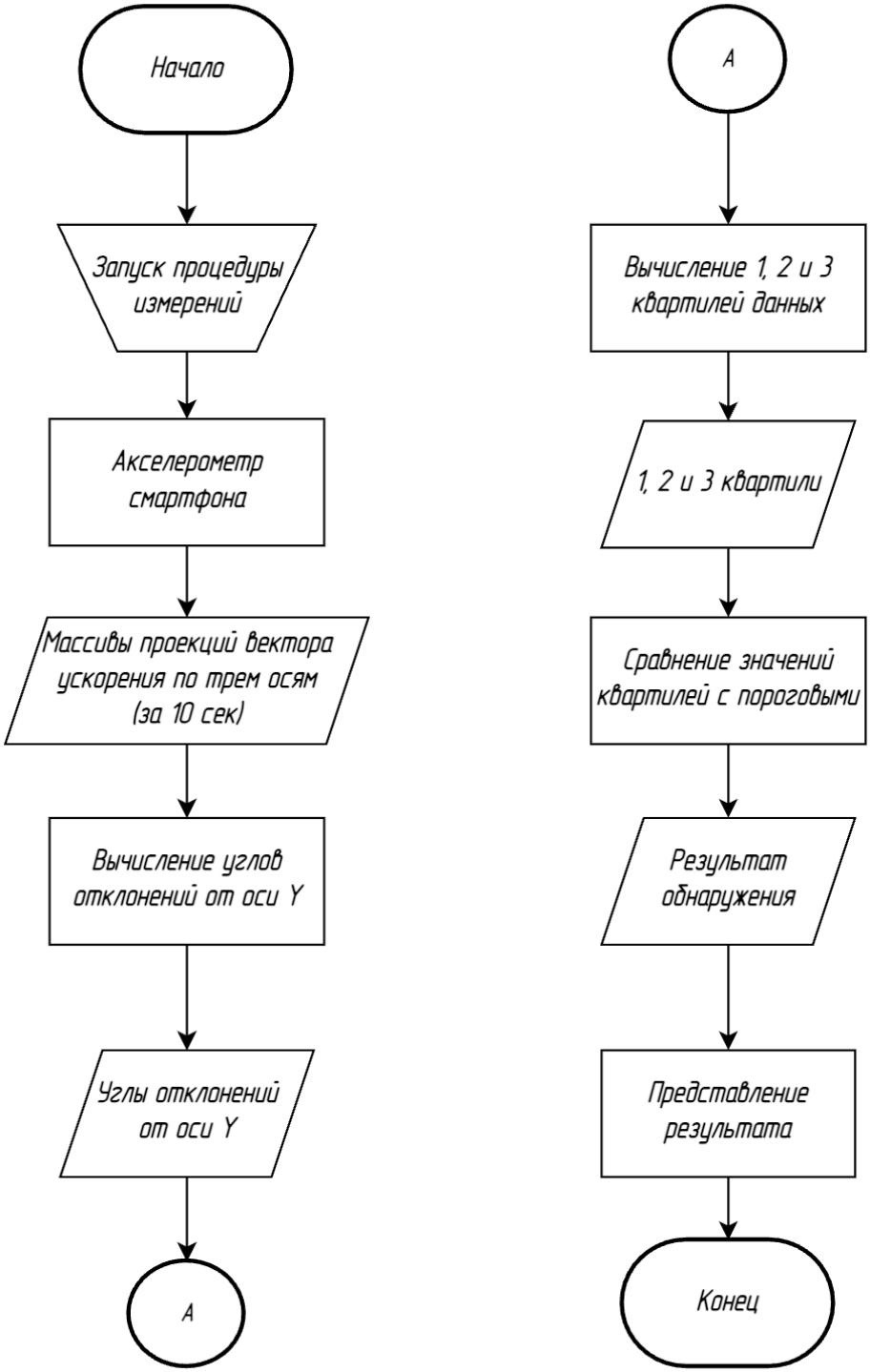


Рисунок 3.10 – Блок-схема алгоритма определения наличия дефектов мелкой моторики

Алгоритм работы данного модуля реализован следующим образом:

- на вход алгоритма поступают данные акселерометра смартфона, содержащие значения компонент ускорения по осям X, Y и Z, полученные с частотой дискретизации 50 Гц;

- корректировка нулевых значений Y: величины проекции на ось Y, равные нулю, заменяются минимальными ненулевыми значениями (~0,000001). Это исключает возможность деления на ноль в формулах (3.12) и (3.13);

- вычисление угла отклонения: для каждого отсчета рассчитывается угол между вектором ускорения и осью Y акселерометра по формуле (3.12):

$$\alpha Y = tan^{-1}\left(\frac{\sqrt{a_x^2+a_z^2}}{a_{y^*}}\right), \quad (3.12)$$

где: a_x - проекция ускорения на ось X акселерометра, a_z - проекция ускорения на ось Z акселерометра, a_{y^*} - скорректированное значение проекции ускорения на ось Y. Если значение проекции ускорения на ось X больше или равно 0, то расчет угла выполняется по формуле (3.13):

$$\alpha Y = -tan^{-1}\left(\frac{\sqrt{a_x^2+a_z^2}}{a_{y^*}}\right). \quad (3.13)$$

Выбор оси Y обусловлен расположением акселерометра на смартфонах с ОС Android, который изображён на рисунке 3.11;

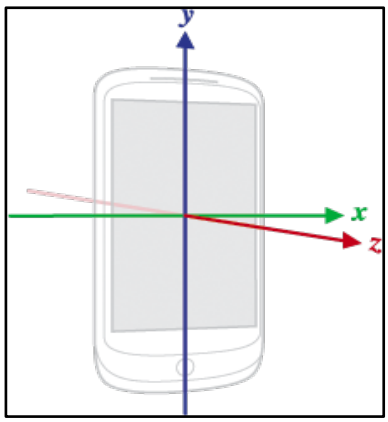


Рисунок 3.11 – Ориентация осей акселерометра смартфонов с ОС Android [88]

- вычисление квартильных значений: полученные данные сортируются в порядке возрастания с последующим расчетом первого (Q1), второго (Q2) и третьего (Q3) квартилей [89];

- анализ пороговых отклонений: для каждого квартиля проводится сравнение с референтными значениями. Нарушение мелкой моторики

фиксируется при выполнении условия |Q1 – Q2| > 20 или |Q3 – Q2| > 20. В противном случае отклонения не фиксируются. Пороговое значение 20 установлено экспериментальным путем.

Блок представления результата осуществляет отображение итогов работы алгоритма для пользователя, а также транслирует данные на серверный узел программного комплекса.

3.3 Программно-алгоритмическая реализация методик анализа аудиоданных для определения показателей жизнедеятельности

3.3.1 Определение кашля и чихания

Проблема определения кашля, чихания и классификации разновидностей кашля представляет собой глубоко исследованную область, для которой разработано множество методов, преимущественно использующих алгоритмы машинного обучения [90-92].

В работе [93] предложен подход к выявлению кашля у пациентов с ХОБЛ с использованием решающих деревьев. Анализ аудиозаписей (частота дискретизации 44,1 кГц) осуществлялся методом скользящего окна (длительность 1 с). Для обучения модели рассматривались следующие аудиохарактеристики:

- спектральные: индекс максимума ДПФ, мел-кепстральные коэффициенты, мощность в частотной области;
- статистические: дисперсия, стандартное отклонение, максимум/минимум сигнала, энтропия, СКЗ, энергия;
- прочие: уровень звукового давления, скорость пересечения нуля.

Модель достигла точности (precision) 85,5% на выборке пациентов 30–89 лет.

В работе [94] описана система распознавания кашля и чихания, применяющая конфиденциальное машинное обучение и сенсоры умных часов. Показания акселерометра устройства запускают анализ звукового сигнала. Признаки (мел-кепстральные характеристики, спектральные свойства, частота пересечения нуля) извлекаются из 5-секундных аудиозаписей. Точность системы достигла 98,75%. В исследовании [95]

авторы предлагают метод диагностики COVID-19 на основе анализа кашля с использованием сверточных нейронных сетей (CNN), где в качестве признаков выступают мел-кепстральные коэффициенты; предложенный метод показал чувствительность (sensitivity) на уровне 98,5%.

В работе [96] для детекции кашля по данным акселерометра применена модель ResNet50 с точностью 96,71%. Использовались следующие типы признаков:

- признаки сигнала акселерометра: спектральная плотность мощности, среднеквадратичное значение сигнала, эксцесс, скользящие средние и пик-коэффициент;
- признаки звукового сигнала: мел-кепстральные коэффициенты, скорость пересечения нуля и эксцесс.

Альтернативные модели (логистическая регрессия, SVM, MLP, CNN, LSTM) показали более низкую эффективность.

Работа [97] посвящена анализу различий между влажным и сухим кашлем, ключевые отличия которых проявляются в частотных характеристиках аудиосигнала, включая коэффициент передачи мощности. Предложенная методология включает двухэтапный процесс: первичную верификацию кашля и последующую классификацию его типа. Сигнал предварительно нормализуется по энергетической оболочке. Далее реализуется пороговый алгоритм (эмпирически настроенный) для детекции пиков в кашлевом сигнале. Анализ демонстрирует повышенную пиковую активность при влажном кашле. Применение полосовой фильтрации (в сравнении с НЧ-фильтром) при извлечении признаков повышает точность дифференциации типов кашля.

Несмотря на успешное использование авторами моделей машинного обучения для выявления кашля и чихания, задача классификации типов кашля (влажный/сухой) остается менее изученной. На основании исследования [97], была разработана система, разделяющая процессы определения кашля/чихания и классификации их типа [98].

Для решения этих задач был разработан модуль, архитектура которого представлена на рисунке 3.12. На вход системы подается аудиопоток, записанный встроенным микрофоном смартфона со следующими параметрами: длительность 2,5 секунды, частота дискретизации 16000 Гц, монофонический режим. Аудиоданные представляют собой массив чисел с плавающей запятой (float). Этот массив поступает в блок обнаружения кашля и чихания, который присваивает одну из трех категорий: «кашель», «чихание» или «другое». При идентификации кашля исходные данные направляются в классификатор типа кашля, определяющий принадлежность к классам «влажный кашель» или «сухой кашель». Таким образом, итоговый результат обработки аудиосигнала относится к одному из четырех классов: «другое», «чихание», «влажный кашель», «сухой кашель».

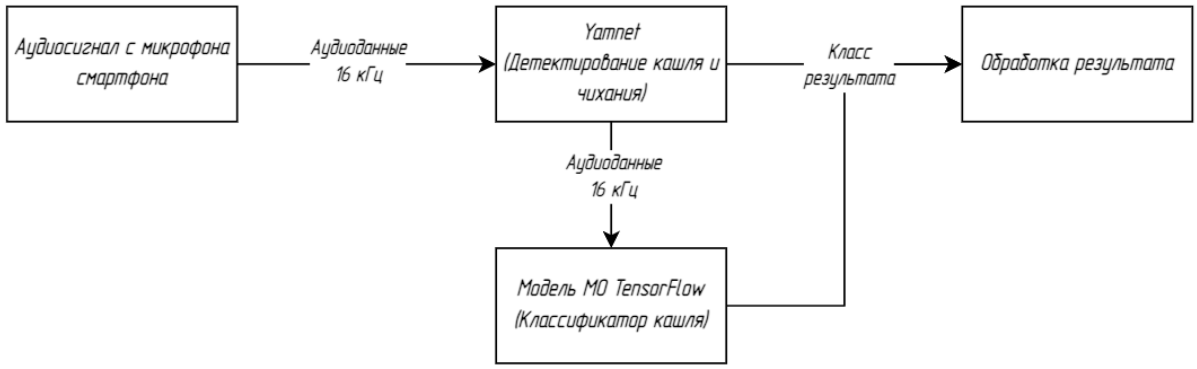


Рисунок 3.12 – Структурная схема детектирования кашля и чихания классификации типа кашля

В данном решении применяется предварительно обученная модель Yamnet [99] из открытого репозитория TensorFlow, способная идентифицировать аудиосигналы по 521 категории, включая чихание и кашель. Модель обрабатывает монофонические аудиозаписи (16 кГц), представленные в формате чисел с плавающей запятой (Float), что требует соответствующей предобработки данных. Yamnet предоставляет вероятностные оценки для каждого распознаваемого класса. Для использования в мобильном приложении файл модели.tflite включается в ресурсы проекта на этапе компиляции. Взаимодействие с TensorFlow Lite

обеспечивается подключением в Gradle-файл модуля следующих зависимостей: org.tensorflow:tensorflow-android:1.13.1, org.tensorflow:tensorflow-lite-task-audio:0.2.0, org.tensorflow:tensorflow-lite:2.4.0. Пороговое значение вероятности для классов "кашель" и "чихание", установленное экспериментальным путем на основе результатов Yamnet, составляет 0,1. Разработка осуществлялась на языке Kotlin с минимальным уровнем API Android 23. Определение пороговых значений выполнялось эмпирическим методом.

На основе моделей определения кашля и чихания была получена матрица ошибок для обученной модели, показанная в таблице 3.3.

Таблица 3.3 – Матрица ошибок обучения модели определения кашля и чихания

Истинный\ Предсказанный	Кашель	Чихание	Ничего
Кашель	0,84	0,11	0,05
Чихание	0,13	0,81	0,06
Ничего	0,04	0,04	0,91

Ввиду отсутствия готовых открытых моделей для классификации кашля на сухой и влажный, было принято решение об обучении специализированной модели. Основной сложностью стал подбор обучающего датасета. Однако задача упрощалась необходимостью различать лишь два класса ("сухой кашель", "влажный кашель"), так как остальные звуки идентифицируются Yamnet. Для обучения в данной работе использован датасет COUGHVID [100], содержащий порядка 30000 аудиозаписей с разметкой, выполненной четырьмя экспертами. Важное преимущество этого набора – наличие в файлах аннотаций пометки о типе кашля (сухой/влажный), наряду с подробными описаниями. Набор также включает дополнительную информацию: возраст, пол, диагноз пациента и т.д. Отбор данных из набора проходил по следующим критериям:

- данные должны иметь в файле разметки тип кашля: либо сухой, либо влажный;

- данные должны иметь в файле разметки указание, что кашель обнаружен;
- данные должны иметь в файле разметки количество экспертов, давших оценку о типе кашля не менее 1;
- отношение количества экспертов, давших оценку одного класса, к общему количеству экспертов, должно быть больше значения 0,5.

Для реализации отбора данных, согласно заданным критериям, был создан скрипт на Python 3.10, использующий библиотеки NumPy и csv. Блок-схема алгоритма отображена на рисунке 3.13. Алгоритм загружает CSV-файл с разметкой датасета, извлекает поля, содержащие тип кашля, его качество и экспертные оценки (по их числовым идентификаторам и позициям в строке CSV). Каждая запись последовательно проверяется на соответствие критериям. Если запись соответствует и имеет тип "влажный кашель", имя её аудиофайла записывается в “wet.txt”; для типа "сухой кашель" – в “dry.txt”. Неподходящие записи игнорируются.

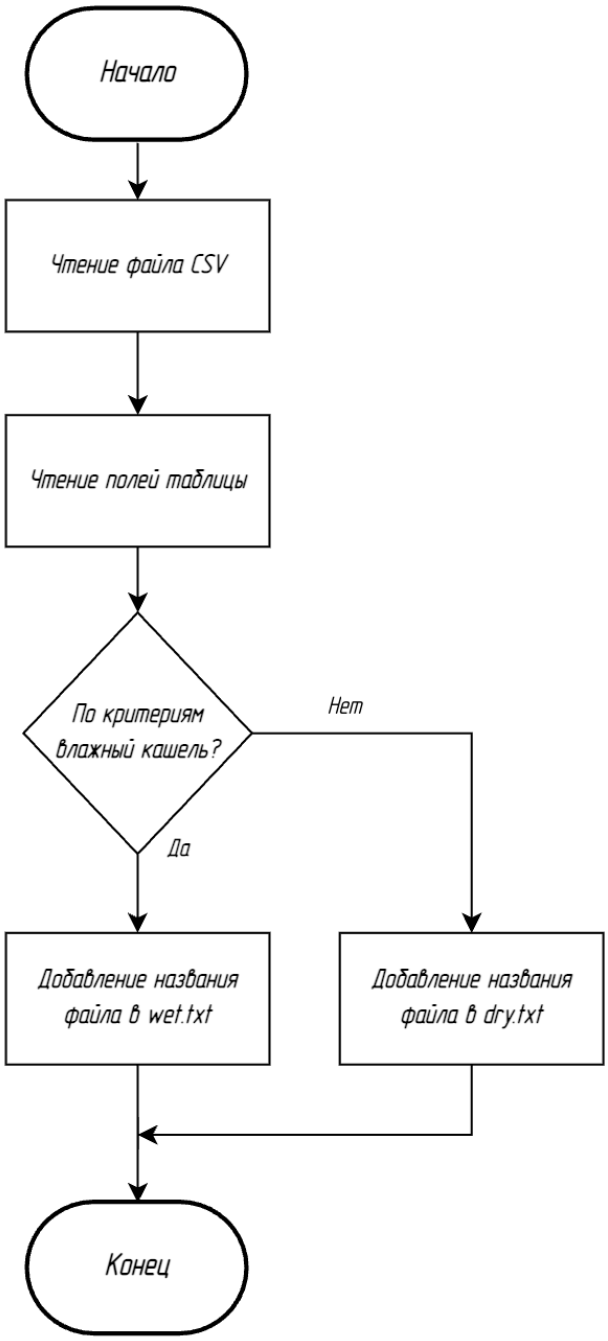


Рисунок 3.13 – Блок-схема алгоритма отбора обучающей выборки

Таким образом, была сформирована выборка, содержащая в себе 527 аудиозаписей с влажным кашлем и 1729 аудиозаписей с сухим кашлем. 77% аудиозаписей принадлежит людям в возрасте до 40 лет, что потенциально может стать проблемой для более возрастных групп и привести к дрейфу данных.

Для формирования векторов признаков принято решение использовать признаки из работы [101], а именно:

- оценка соотношений мощности частотных полос от 0 до 750 Гц и от 1 кГц до 2,5 кГц, определяемой по формуле:

$$PR = \frac{P_{0-750}}{P_{1000-2500}}, \quad (3.14)$$

где: P_{0-750} – мощность на полосе от 0 – 750 Гц, $P_{1000-2500}$ – мощность на полосе от 1000 – 2500 Гц;

- скорость пересечения нуля, определяемая по формуле:

$$ZCR = \frac{crossNum}{len-1}, \quad (3.15)$$

где: crossNum – количество пересечений нуля, len – длина массива аудиосигнала;

- среднеквадратичная мощность, определяемая по формуле:

$$RMSP = \sqrt{\frac{\sum_{i=0}^{len} x_i^2}{len}}, \quad (3.16)$$

где: x_i – значение дискретного представления аудиосигнала, len – длина массива аудиосигнала;

- спектральная неравномерность, определяемая по формуле:

$$SF = \frac{gmean}{amean}, \quad (3.17)$$

где: gmean – геометрическое среднее спектральной плотности мощности сигнала, amean – арифметическое среднее спектральной плотности мощности сигнала;

- спектральное стандартное отклонение по формуле:

$$STD = \sqrt{\frac{\sum_{i=0}^{len}(p_i - amean)^2}{len-1}}, \quad (3.18)$$

где: p_i – элемент массива спектральной плотности мощности сигнала, amean – арифметическое среднее спектральной плотности мощности сигнала, len – длина массива аудиосигнала;

- спектральный наклон по формуле:

$$SS = \frac{\sum_{k=b_1}^{b_2}(f_k - \mu_f)(s_k - \mu_s)}{\sum_{k=b_1}^{b_2}(f_k - \mu_f)^2}, \quad (3.19)$$

где: f_k – частота, которая является k элементом из массива частот, μ_f – среднее значение частоты, s_k – k элемент массива спектра, μ_s – среднее значение спектра, b_1 и b_2 – границы частотной полосы;

- спектральное убывание, по формуле:

$$SD = \frac{\sum_{k=b_1+1}^{b_2} \frac{s_k - s_{b_1}}{k-1}}{\sum_{k=b_1+1}^{b_2} s_k}, \quad (3.20)$$

где: s_k – k элемент массива спектра, b_1 и b_2 – границы частотной полосы;

- пик-коэффициент, по формуле:

$$CF = \frac{peak}{RMSP}, \quad (3.21)$$

где: peak – максимальное значение сигнала (пик), RMSP – среднеквадратичное значение мощности сигнала;

- спектральная мощность для групп частот: от 0 Гц до 200 Гц, от 300 Гц до 425 Гц, от 500 Гц до 650 Гц, от 950 Гц до 1150 Гц, от 1400 Гц до 1800 Гц, от 2300 Гц до 2400 Гц, от 2850 Гц до 2950 Гц, от 3800 Гц до 3900 Гц. Спектральная плотность мощности определяется методом Вельча [102]. Мощность полосы определяется с помощью интегратора Симпсона [103] в соответствии с формулой:

$$SIMPS[a,b] \approx \frac{b-a}{6}\left[p(a) + 4p\left(\frac{a+b}{2}\right) + p(b)\right], \quad (3.22)$$

где: a – нижняя граница частотной полосы, b – верхняя граница частотной полосы, p – спектральная плотность мощности.

Модель классификации кашля обучалась с помощью скрипта на Python 3.10 (NumPy, TensorFlow). Данные разбиты на обучающую (90%) и валидационную (10%) выборки. Модель построена как полносвязная нейросеть (структура на рис. 3.14), где для регуляризации и борьбы с переобучением добавлены dropout-слои между скрытыми слоями.

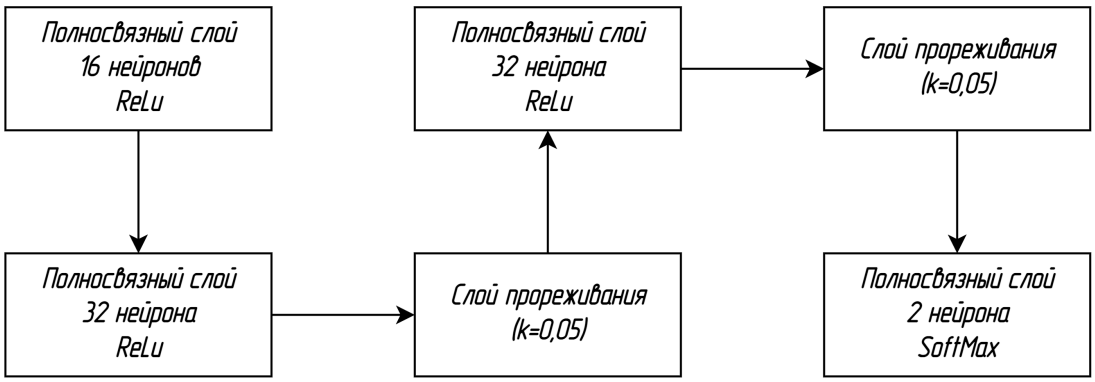


Рисунок 3.14 – Структура нейронной сети классификатора типа кашля

В качестве оптимизатора при обучении сети выбран метод стохастической оптимизации Адама [104], функция потерь – перекрестная энтропия [105]. Обучение проходило в 5 этапов по 2000 эпох каждый. Скрипт обучения сначала загружает списки аудиофайлов (из текстовых файлов dry.txt и wet.txt, полученных из разметки COUGHVID). Затем для каждого файла в этих списках с помощью librosa производится чтение аудиоданных и извлечение указанных ранее признаков. Процедура предварительной обработки аудиосигнала состояла из следующих этапов:

- проверка числа каналов аудиоданных и перевод в моноканальный режим, если аудиоданные записаны в стереоформате;
- нормировка аудиосигнала по пиковому значению;
- обработка аудиосигнала фильтром низких частот (ФНЧ) Баттерворта [106]4 порядка с частотой среза 4 кГц;
- передискретизация сигнала в соответствии с частотой среза сигнала и теоремой Котельникова-Найквиста путем применения сглаживающего фильтра Чебышева 8 порядка 1 рода, АЧХ которого представлена в формуле:

$$|H(j\omega)| = \frac{1}{\sqrt{1+\varepsilon^2 T_n^2\left(\frac{\omega}{\omega_c}\right)}}, \quad (3.23)$$

где: ω – частота сигнала, ω_c – частота среза фильтра, ε – показатель пульсаций, $T_n(x)$ – многочлен Чебышева n-го порядка.

После этапа предобработки система извлекает указанные ранее аудиопризнаки. Сформированный массив признаков передается в модель

машинного обучения для обучения. В результате тренировочного процесса создается модель в формате H5, которая затем преобразуется в совместимый с мобильными устройствами Android формат TensorFlow Lite (tflite).

3.3.2 Определение эмоций

Эмоции могут оказывать влияние на состояние здоровья. Негативные эмоции влияют на сердечно-сосудистую систему, могут приводить к паническим расстройствам, гипертонии, депрессии и к другим отклонениям [107]. Существует множество работ, посвященных анализу эмоций в контексте решения узких задач. Авторы [108] предлагают модель для решения задачи определения эмоций, которая основана на использовании технологии глубокого обучения (Deep Learning) для анализа аудиоданных с помощью сверточных нейронных сетей и полносвязных слоев. Основными признаками являются мел-кепстральные коэффициенты. Среди уже созданных наборов данных пригодным для данной модели оказался набор данных RAVDESS. Набор содержал в себе как женские, так и мужские голоса. Набор данных состоял из следующих элементов: путь к файлу, источник, номер голоса, пол, интенсивность, повторение, эмоция. Авторы в своей работе рассматривают решение, основанное только для анализа мужского голоса ввиду того, что тембр и высота голоса отличаются для мужчин и женщин. Авторы представили матрицу неточностей для пяти классов эмоций, которая показана на рисунке 3.15. Была проведена работа по валидации модели, представленной авторами для женского пола, в результате чего была получена матрица неточностей на рисунке 3.16. Из сравнения результатов было выяснено, что имеются сильные расхождения с решением авторов. Более того ограниченность в принадлежности к полу человека является очень существенным недостатком, вследствие чего модель, представленная в исследовании, не является пригодной для реализации в конечном решении.

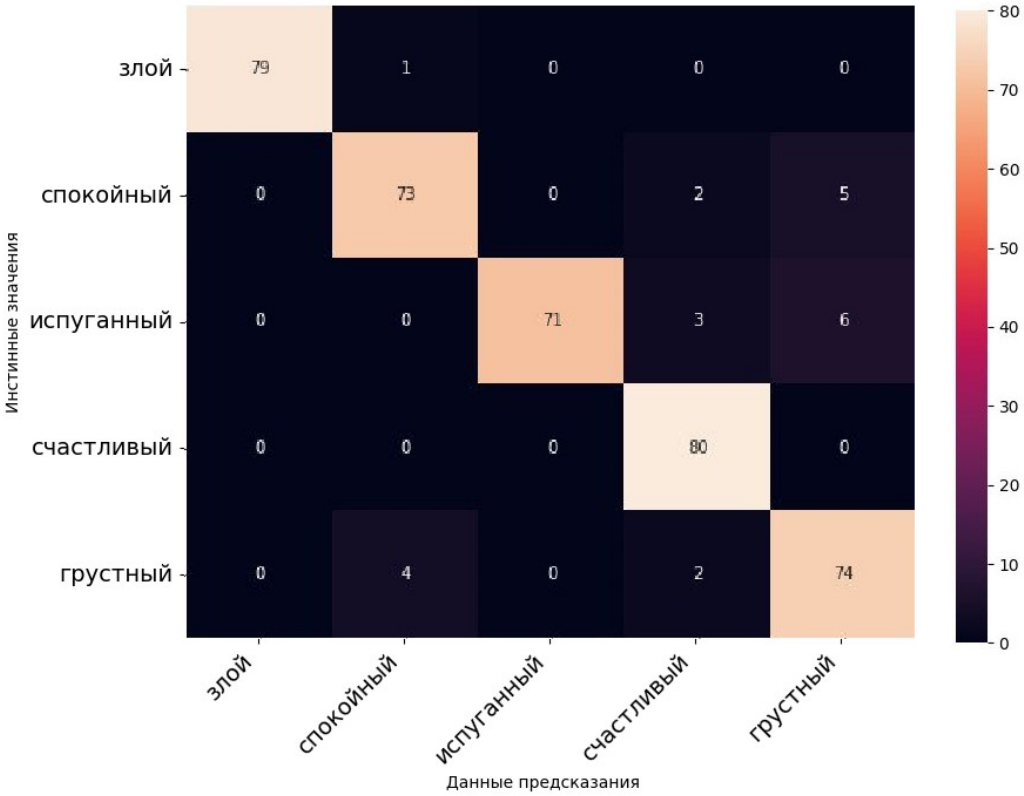


Рисунок 3.15 – Матрица ошибок для определения эмоций для лиц мужского пола по [108]

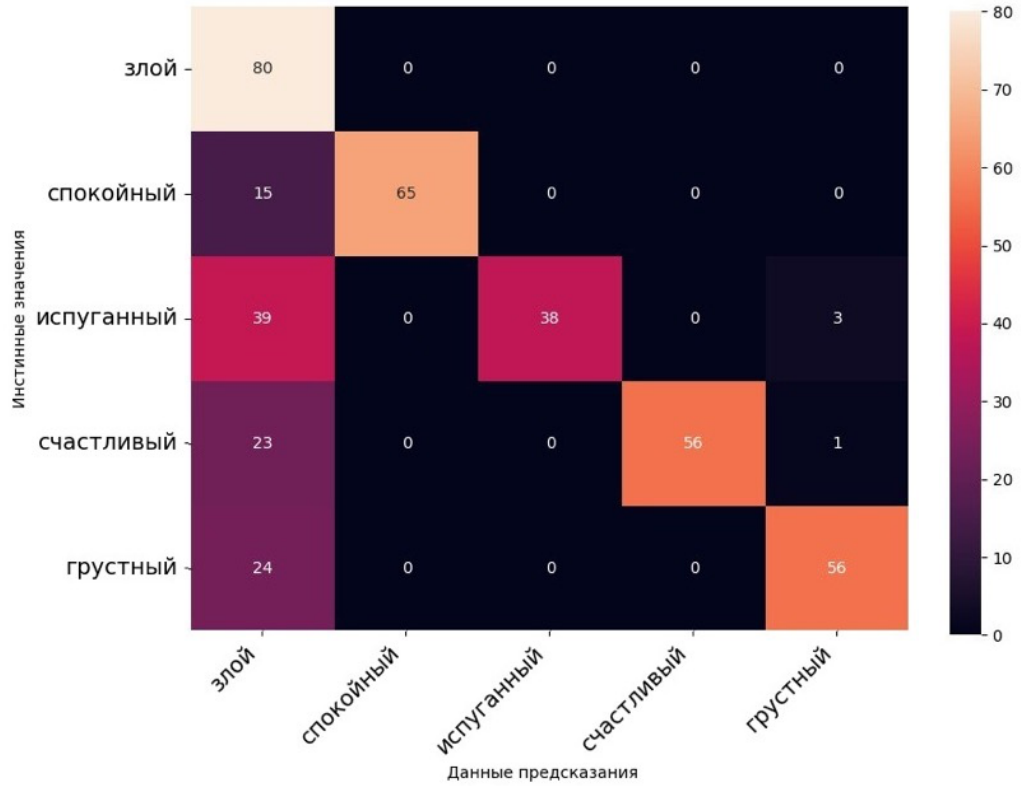


Рисунок 3.16 – Матрица ошибок для определения эмоций для лиц женского пола по [108]

На платформе Kaggle авторы [109] описали процесс построения модели определения эмоций начиная от разбиения на табличную структуру данных и заканчивая построением модели классификатора голоса человека и проверкой данных. Матрица неточностей изображена на рисунках 3.17 и 3.18.

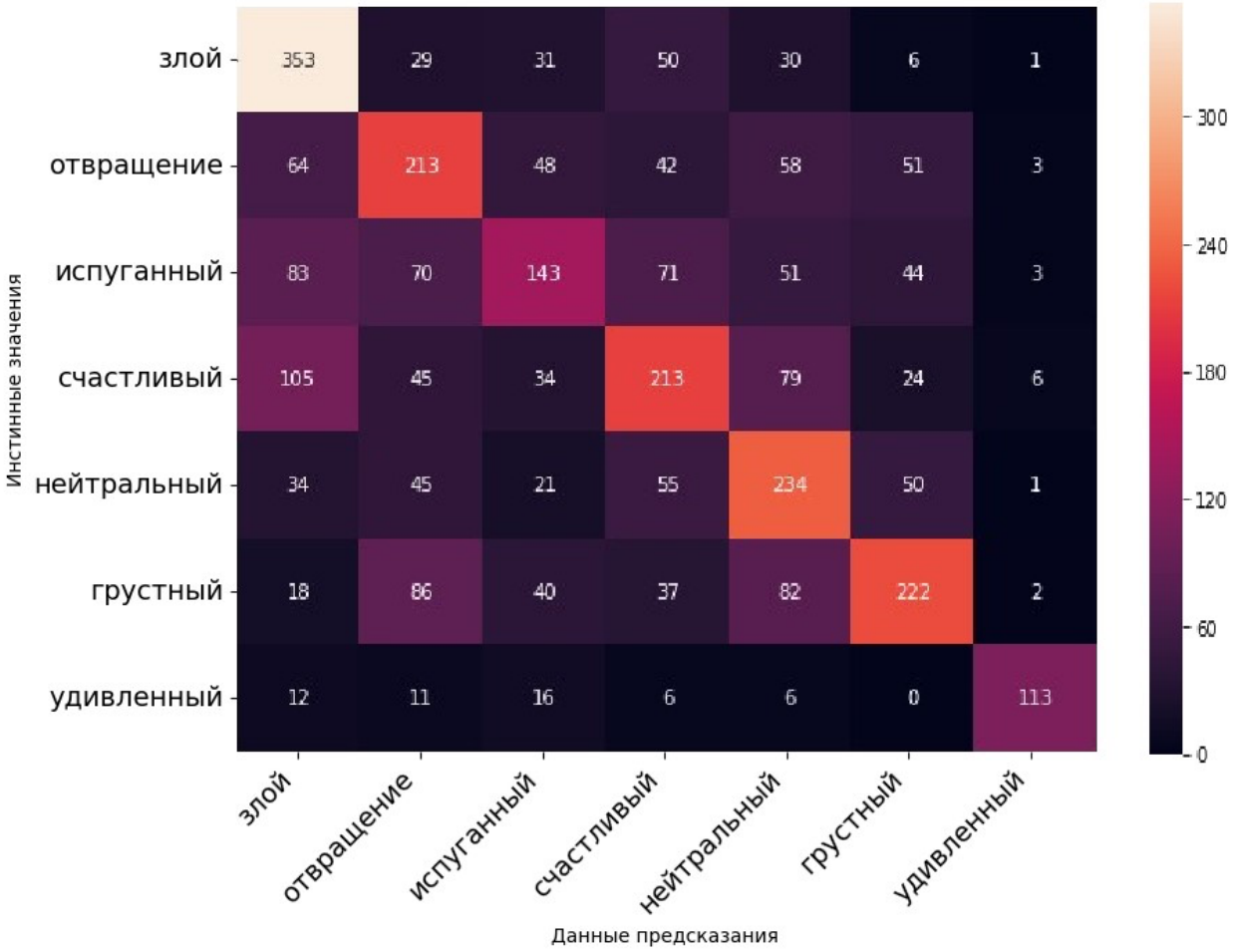


Рисунок 3.17 – Матрица ошибок для определения эмоций для лиц мужского пола по [109]

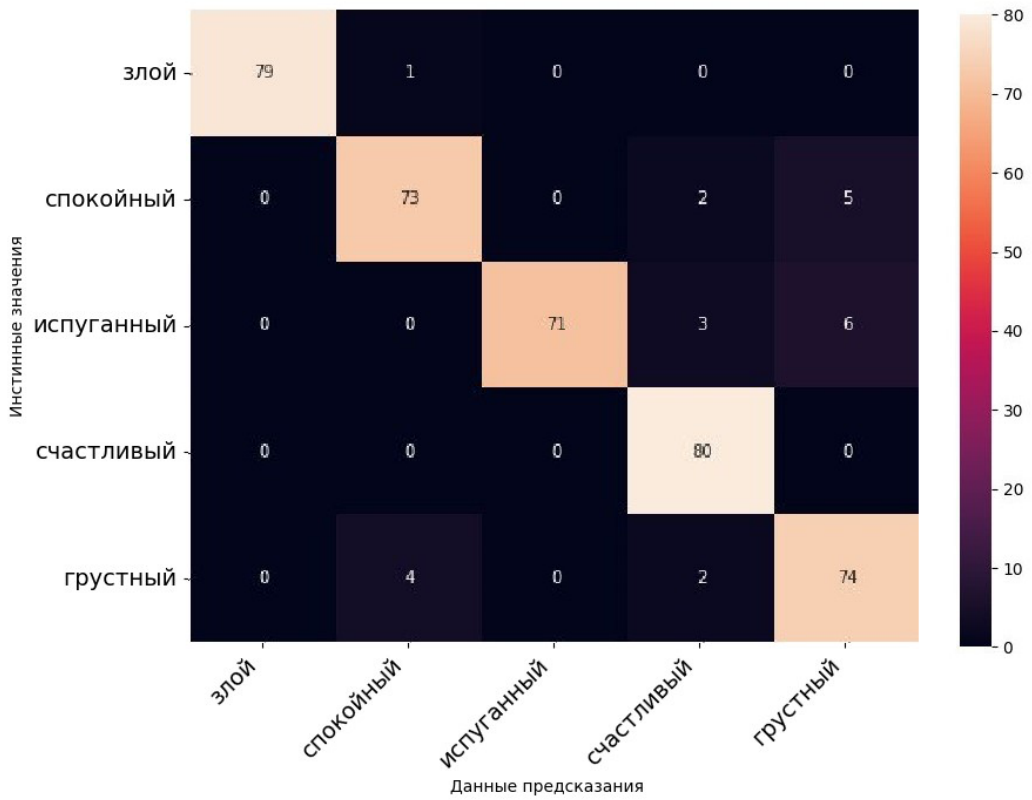


Рисунок 3.18 – Матрица ошибок для определения эмоций для лиц женского пола по [109]

Таким образом, было выяснено, что существует множество решений, связанных с определением эмоций; основная сложность в этом вопросе заключалась в поиске набора данных для обучения.

Для проверки готовых решений и обучения модели, был произведен поиск данных. Критерии, по которым были отобраны наборы данных:

- Четкость аудиозаписей (без шума)
- Сбалансированность по количеству голосов (акцент, национальность)
- Объем набора
- Язык

Были отобраны наиболее распространенные наборы данных, которые находятся в открытом доступе:

- SAVEE [110] Коллекция аудиозаписей, созданная при участии четырех добровольцев – носителей английского языка. Все они являются аспирантами и научными сотрудниками Университета Суррея возрастом до 31

года. Набор данных охватывает семь классов эмоций: гнев, отвращение, страх, радость, грусть, удивление, а также нейтральное.

- RAVDESS [111] набор данных, состоящий из 1440 записей, полученных от 24 человек: 12 мужчин и 12 женщин. Каждый участник предоставил по 60 аудиофайлов, отражающих семь эмоций: печаль, гнев, счастье, нейтральное состояние, испуг, удивление и отвращение. Все записи представлены в двух вариантах эмоциональной выраженности – умеренном и усиленном.

- CREMA-D [112] представляет собой коллекцию из 7442 аудиозаписей, созданных при участии 91 человека (48 мужчин и 43 женщин) в возрасте от 20 до 73 лет. В процессе записи участники произносили 12 заранее заданных фраз, передавая шесть эмоций: злость, счастье, испуг, отвращение, печаль и нейтральное состояние. Каждая эмоция была выражена с четырьмя степенями интенсивности: слабой, умеренной, сильной и неоднозначной.

В качестве основных признаков было принято решение использовать статистические признаки и мел-кепстральные коэффициенты (MFCC), которые отражают частотные характеристики звука, учитывая чувствительность человеческого уха к различным частотам. Человеческих слух воспринимает высоты звука (Мел) не линейно, по отношению к частоте, и описывается следующим образом при помощи формулы мел-шкалы образуя тем самым механизм перехода от Гц к Мелам:

$$M(f) = 1125 \ln\left(1 + \frac{f}{700}\right). \quad (3.24)$$

График зависимости мел-шкалы от частоты представлен на рисунке 3.19.

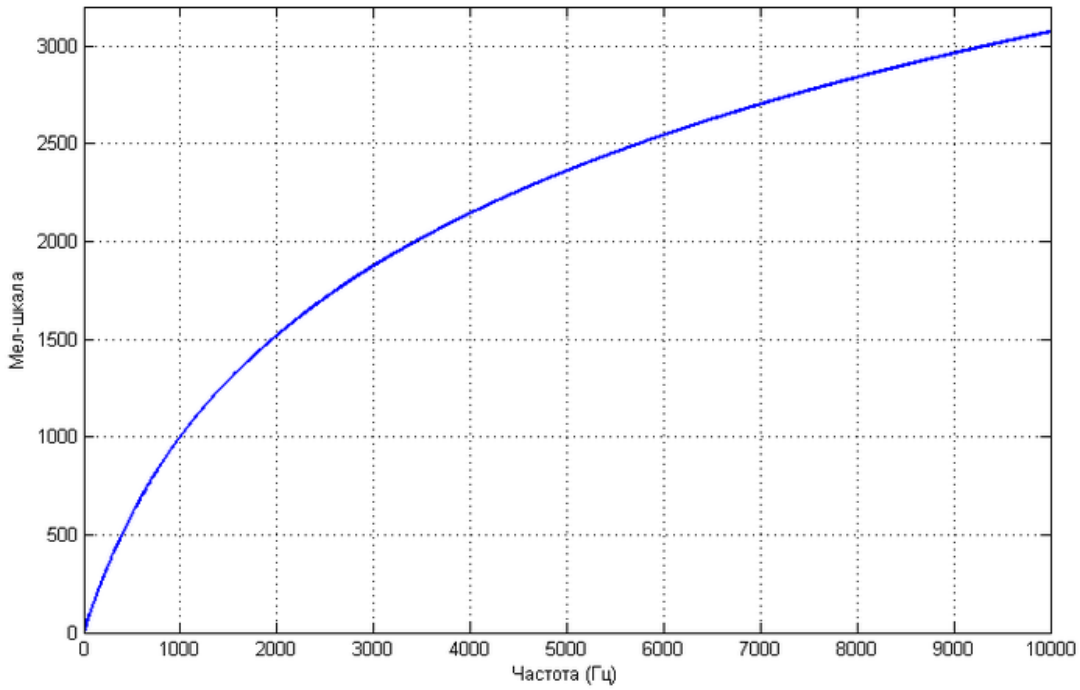


Рисунок 3.19 – График зависимости мел-шкалы от частоты

Расчет мел-кепстральных коэффициентов была реализована следующим образом:

- преобразование Фурье для получения спектра частот;
- применение мел-фильтров для получения мел-спектра;
- вычисление логарифма мел-спектра;
- применение обратного преобразования Фурье для получения мел-кепстральных коэффициентов.

Реализацию извлечения звуковых признаков аудиоданных выполнялась с помощью средств библиотеки openSMILE [113]. Также для представления данных в единый вид набор данных, ввиду наличия в них более 6 классов, необходимо было отсортировать по трем классам эмоций: нейтральный, положительный и отрицательный.

Каждый из наборов имел свой идентификатор имени файла. В наборе SAVEE аудиофайлы названы таким образом, 'DC_a03.wav'. Первые два символа имени файла представляют инициалы говорящего. Буквы перед числовым значением описывают классы эмоций следующим образом:

- ‘a’ – anger(злость);

- ‘d’ – disgust(отвращение);
- ‘f’ – fear(страх);
- ‘h’ – happiness(радость);
- ‘n’ – neutral(нейтральное);
- ‘sa’ – sadness(грусть);
- ‘su’ – surprise(удивление).

Число в имени файла обозначает номер прочитанного предложения.

В наборе RAVDESS, например файл ‘03-01-01-01-01-01-01.wav’ именовался набором чисел, где каждый номер имеет обозначение, описание которых представлено в таблице 3.4:

Таблица 3.4 – Описание имени файла набора RAVDESS

Значение	**Модальность**	**Речевой канал**	**Эмоция**	**Эмоциональная напряженность**	**Высказывание**	**Повторение**	**Актер**
01	Аудио с видео	Речь	Нейтральный	Нормальное	Высказывание 1	Первое повторение	Муж. 1
02	Только видео	Песня	Спокойный	Сильное	Высказывание 2	Второе повторение	Жен. 1
03	Только аудио	-	Радостный	-	-	-	Муж. 2
04	-	-	Грустный	-	-	-	Жен. 2
05	-	-	Злой	-	-	-	Муж. 3
06	-	-	Напуганный	-	-	-	Жен. 3
07	-	-	Разраженный	-	-	-	Муж. 4
08	-	-	Удивленный	-	-	-	Жен. 4
09	-	-	-	-	-	-	Муж. 5
10	-	-	-	-	-	-	Жен. 5
11	-	-	-	-	-	-	Муж. 6
12	-	-	-	-	-	-	Жен. 6
13	-	-	-	-	-	-	Муж. 7
14	-	-	-	-	-	-	Жен. 7
15	-	-	-	-	-	-	Муж. 8
16	-	-	-	-	-	-	Жен. 8
17	-	-	-	-	-	-	Муж. 9
18	-	-	-	-	-	-	Жен. 9
19	-	-	-	-	-	-	Муж. 10
20	-	-	-	-	-	-	Жен. 10
21	-	-	-	-	-	-	Муж. 11
22	-	-	-	-	-	-	Жен. 11
23	-	-	-	-	-	-	Муж. 12
24	-	-	-	-	-	-	Жен. 12

Также было принято решение использовать набор данных CREMA-D. Для данного набора имеется файл формата CSV с идентификаторами

распределения по эмоциям и человека. Каждый из наборов данных был записан в табличную структуру данных, которые были объединены для дальнейшего извлечения признаков.

В качестве модели машинного обучения было принято решение использовать архитектуру сверточной нейронной сети, структура которой изображена на рисунке 3.20.

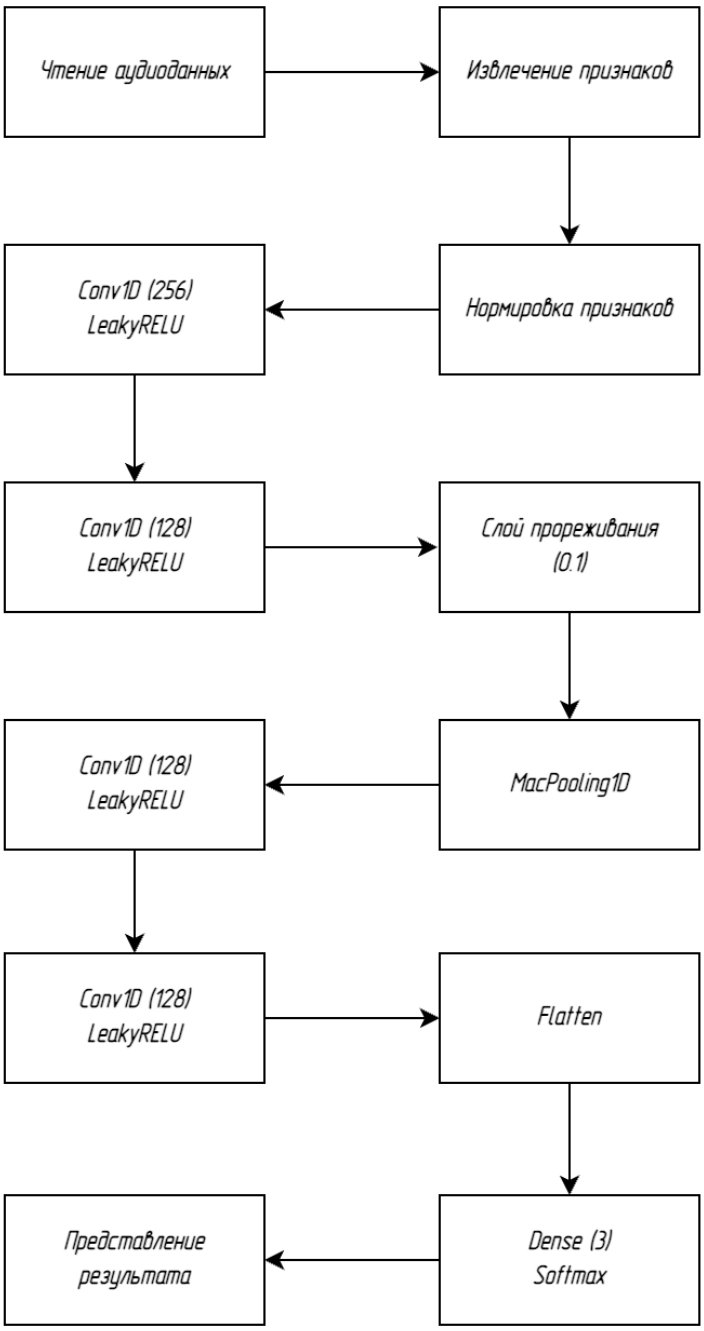


Рисунок 3.20 – Структура алгоритма машинного обучения для определения эмоций человека

На основе модели определения эмоций была получена матрица ошибок, показанная в таблице 3.5.

Таблица 3.5 – Матрица ошибок обучения модели определения эмоций

Истинный\ Предсказанный	Негативная	Нейтральная	Положительная
Негативная	0,74	0,15	0,11
Нейтральная	0,11	0,75	0,24
Положительная	0,15	0,16	0,69

3.4 Необходимость использования федеративного обучения

Ранее в работе были представлены модели определения кашля и чихания и определения эмоций. Однако есть существенная особенность при работе с аудиоданными голоса человека. Это потенциальное наличие дрейфа данных.

Причинами дрейфа данных в модели машинного обучения, работающей с голосом человека, могут стать следующие факторы:

- изменения в записи аудиоданных: смена микрофонов, условий записи, уровня и типов фона (шумы, эхо и пр.), которые влияют на спектральные характеристики голоса и, соответственно, на значения признаков [114];
- изменения в самой популяции говорящих: возрастные изменения, смена говорящих, различия в акцентах, эмоциональных состояниях, сезонные изменения, социальные изменения, отклонения, которые могут повлиять на восприятие и выражение голоса [115-117];
- несбалансированная обучающая выборка;
- непредвиденные внешние события или изменения контекста, которые влияют на поведение человеческого голоса и тем самым меняют статистику входных данных.

Ввиду того, что количество причин, влияющих на голос человека и его изменение, может быть достаточно обширным, уже имеющиеся наборы данных и обученные на них модели не решают полностью задачу классификации. Таким образом, может возникнуть ситуация, которая потребует совершенствования моделей машинного обучения. В связи с этим

было принято решение построить механизм совершенствования моделей машинного обучения, который состоит из двух важных компонент:

- детектор дрейфа данных, реализация которого описана в предыдущей главе;
- федеративное обучение, механизм работы которого описан в прошлой главе.

Для реализации федеративного обучения в программном комплексе используются средства библиотеки TensorFlow в модуле TensorFlow Federated. Для локального обучения на мобильном устройстве TensorFlow Lite версий выше 2.4 также предоставляет инструментарий [118].

Для федеративного обучения использовался алгоритм FedAvg, который работает следующим образом:

- Сервер располагает начальной моделью и рассылает веса клиентским устройствам.
- При проведении локального обучения вычисляются локальные весовые коэффициенты, которые при отправке на серверный узел позволяют обновить глобальную модель по следующей формуле агрегации:

$$\omega_{global} = \sum_k \left(\frac{n_k}{n}\right) \cdot \omega_{k\ local}, \tag{3.25}$$

где: n_k – размер обучающей выборки на клиенте k, n – общий размер обучающих выборок среди всех клиентов, участвующих в обучении, $\omega_{k\ local}$ – весовой коэффициент локальной модели клиента k.

Локальное дообучение выполняется по всем слоям модели: на клиентском устройстве весовые коэффициенты инициализируются от текущей глобальной модели, после чего вся сеть обучается на локальных данных пользователя в течение 100 эпох, что обеспечивает сходимость процедуры при малом объёме локальных данных, характерном для персонального мониторинга. По окончании локального обучения на серверный узел передаются обновленные весовые коэффициенты локальной модели в зашифрованном виде в соответствии со схемой гомоморфного шифрования Пэйе, после чего выполняется их агрегация по формуле (3.25) и

обновление глобальной модели. Верификация обновлённой модели выполняется в последующих сессиях мониторинга тех же пользователей: данные, накопленные в последующих сессиях, подаются на вход обновлённой модели, а полученные значения метрик сопоставляются с результатами испытаний до обновления (глава 4, таблицы 4.13–4.22).

Для реализации аспекта, связанного с информационной безопасностью, была выбрана криптографическая система Пэйе [119]. Она обеспечивает сложение зашифрованных чисел, умножение зашифрованного числа на незашифрованное скалярное значение, сложение зашифрованного числа с незашифрованным скалярным значением. Принцип работы этой криптосистемы построен следующим образом:

- Задаются два больших простых числа p и q, причем:

$$\lambda = \text{НОК}(p-1, q-1). \quad (3.26)$$

- Вычисляется.$n = p \cdot q$

- Выбирается число g из множества $Z^*_{n^2}$ - множества всех чисел от 1 до$n^2 - 1$, которые взаимно просты с n^2 (НОД = 1). При этом:

$$\mu = \left(L\left(g^{\lambda}\right) mod\, n^2\right)^{-1} mod\, n, \quad (3.27)$$

где: L(u) – вспомогательная функция, определяемая по формуле (3.28); μ – параметр, вычисляемый по формуле (3.27); λ – параметр, вычисляемый по формуле (3.26); n – модуль шифрования (n = p·q); g – элемент множества Z*n², взаимно простой с n; c – шифротекст; m – открытый текст; r – случайное число, взаимно простое с n.

$$L(u) = \frac{(u-1)}{n}. \quad (3.28)$$

- Пара (n, g) является открытым ключом, а (λ, μ) – закрытым ключом.

- Шифрование текста выполняется путем выбора случайного числа r из множества $Z^*_{n^2}$ и вычислением зашифрованного текста по формуле:

$$c = g^{,m} r^n \, mod\, n^2. \quad (3.29)$$

- Расшифровка определяется по формуле:

$$m = L\left(c^{\lambda} \bmod n^2\right)\mu \bmod n. \quad (3.30)$$

Стоит отметить, что гомоморфное шифрование применяется не к самим исходным показателям жизнедеятельности пользователей программного комплекса, а к весовым коэффициентам локальной модели машинного обучения.

Реализация криптографической схемы Пэйе выполнена в программном комплексе для защиты весовых коэффициентов локальных моделей машинного обучения при их передаче на серверный узел и выполнении операции гомоморфной агрегации. Так как шифрованию подвергаются только весовые коэффициенты, формируемые по завершении процедуры локального обучения, а не потоковые данные сенсоров, накладные вычислительные расходы возникают лишь на этапе агрегации модели. Вычислительная сложность операций шифрования и расшифрования определяется операциями модульного возведения в степень по модулю n (формула 3.29), что характерно для схем асимметричного шифрования, и является допустимой для мобильного клиента с учетом ограниченного объема передаваемых весовых коэффициентов по сравнению с вычислительными затратами на обучение моделей непосредственно на устройстве. Накладные расходы были измерены экспериментально: агрегация зашифрованных обновлений весовых коэффициентов от 50 клиентских устройств выполняется на серверном узле за время, не превышающее 0,4 с, и осуществляется однократно по завершении локального обучения в отличие от задержек обработки отдельных HTTP-запросов, приведенных в таблице 4.23; шифрование весовых коэффициентов на мобильном устройстве занимает в среднем 15 мс. Суммарные накладные расходы при этом не превышают 3% вычислительных затрат на обучение локальной модели.

3.5 Выводы по главе

1. В данной главе были показаны методики определения показателей жизнедеятельности с помощью визуальной сенсорикой. Показана методика определения частоты сердечных сокращений методом фотоплетизмографии,

которая заключается в анализе интенсивности светового сигнала регистрируемого оптической камерой смартфона. Также показан алгоритм определения тремора головы, который основан на спектральном анализе поведения точек контура лица человека. Таким образом, визуальные сенсоры смартфона позволяют строить модули для неинвазивного определения целого ряда показателей жизнедеятельности.

2. Также были описаны методики определения показателей жизнедеятельности кинематическими сенсорами для определения положения тела моделью машинного обучения по данным гироскопа и акселерометра, и мелкой моторики, используя анализ средних значений квартилей данных акселерометра. Таким образом, кинематические сенсоры являются значимой частью программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека, которые позволяют решать задачи определения показателей не только на основе методов цифровой обработки сигналов, но и с использованием методов машинного обучения.

3. Представлены методики определения показателей жизнедеятельности акустическими сенсорами для определения кашля и чихания, а также типа кашля и определения эмоций человека применяя модели машинного обучения. К тому же было установлено, что модели, работающие с аудиоданными, подвержены риску возникновения дрейфа данных, вследствие чего был сделан вывод, что для обеспечения долговременной релевантности методов определения показателей жизнедеятельности по аудиоданным, основанных на моделях машинного обучения необходимо использовать детектор дрейфа и проводить модификацию модели машинного обучения.

4. Обоснована необходимость использования федеративного обучения для моделей машинного обучения, работающих с данными разнородных сенсоров и прежде всего с голосом человека: вследствие многообразия факторов дрейфа данных (изменение условий и каналов записи, смена говорящих и их возрастных и эмоциональных характеристик, несбалансированность обучающих выборок, непредвиденные внешние

события) статичные наборы данных и обученные на них модели не обеспечивают долговременной релевантности результатов. Для реализации федеративного обучения используется алгоритм FedAvg, а защита передаваемых обновлений обеспечивается криптографической системой Пэйе, позволяющей выполнять гомоморфное агрегирование зашифрованных весовых коэффициентов локальных моделей на серверном узле.

ГЛАВА 4. ПРОВЕДЕНИЕ ИСПЫТАНИЙ ПРОГРАММНОГО КОМПЛЕКСА

Для оценки работоспособности разработанной архитектуры программного комплекса необходимо провести комплекс испытаний, который направлен на проверку факта повышения эффективности программного комплекса в аспектах, изложенных в задачах работы. В данной главе изложены условия и методика испытаний разработанной системы, приведены полученные результаты.

4.1 Экспериментальная установка и методика проведения испытаний

4.1.1 Экспериментальная установка и используемое оборудование

Для проведения испытаний разработанного программного комплекса, основанного на самоадаптирующейся микросервисной архитектуре с использованием федеративного обучения, использовалась следующая экспериментальная установка:

- серверный узел – представляет собой серверный компьютер со следующими техническими характеристиками: процессор Intel® Core™ i9-9920X, 64 Гб ОЗУ, SSD-накопитель емкостью 1 Тб, видеоадаптер NVIDIA Quadro RTX 6000, клавиатура, монитор, мышь. На серверном узле была установлена и развернута серверная компонента разработанного программного комплекса, на языке Python, а также развернута СУБД PostgreSQL, осуществляющая работу с базами данных. Все эти программные средства работали на операционной системе Ubuntu 22.04;
- клиентский узел – набор устройств на базе операционной системы Android, представленные в таблице 4.1:

Таблица 4.1 – Устройства на базе ОС Android и их вычислительные характеристики

Название устройства	Емкость АКБ, мА*ч	Процессор (SoC)	ОЗУ, ГБ	Год выпуска	Класс производительности
Huawei nova 8i	4300	Qualcomm Snapdragon 662	8	2020	Начальный
Samsung A8 (2018)	3000	Samsung Exynos 7885	4	2018	Средний
Xiaomi Mi 9 Lite	4030	Qualcomm Snapdragon 710	6	2019	Средний
Redmi Note 9S	5020	Qualcomm Snapdragon 720G	6	2020	Средний
POCO X3 Pro	5160	Qualcomm Snapdragon 860	8	2021	Средне-высокий
Samsung SM-A515F	4000	Samsung Exynos 9611	6	2019	Средний
Huawei nova Y72S	6000	Qualcomm Snapdragon 680	8	2025	Средний
Samsung Galaxy A06	5000	MediaTek Helio G85	6	2024	Начальный
Xiaomi Redmi 13	5030	MediaTek Helio G91 Ultra	8	2024	Начальный
Honor X7c	5200	Qualcomm Snapdragon 4 Gen 2	8	2025	Средний
Realme 14T	6000	MediaTek Dimensity 6300	8	2025	Средний
POCO M7 Pro	5110	MediaTek Dimensity 7025 Ultra	8	2024	Средний
Tecno SPARK 20 Pro	5000	MediaTek Helio G99	8	2023	Средний
Samsung Galaxy A26	5000	Samsung Exynos 1280	8	2025	Средний
Samsung Galaxy A36	5000	Qualcomm Snapdragon 6 Gen 3	8	2025	Средний

Для анализа вычислительной нагрузки устройства, участвовавшие в испытаниях, сгруппированы по классам производительности, определяемым по установленному процессору (SoC) и объему оперативной памяти: начальный класс (процессоры Qualcomm Snapdragon 6xx и MediaTek Helio серии G8x/G9x), средний класс (Qualcomm Snapdragon 7xx, 4 Gen 2 и 6 Gen 3,

Samsung Exynos 7885/9611/1280, MediaTek Dimensity 6xxx/7xxx, Helio G99) и средне-высокий класс (Qualcomm Snapdragon 860). Вычислительные ограничения в наибольшей степени проявляются на устройствах начального класса при выполнении моделей глубокого обучения (определение кашля и чихания, распознавание эмоций) в фоновом режиме и при одновременной обработке аудио- и кинематических данных. Компенсация ограничений обеспечивается модулем контроля ресурсов, который при высокой нагрузке на ЦП снижает частоту дискретизации сенсоров и вводит паузы при чтении аудиоданных (раздел 2.4). Состояние аккумуляторов устройств контролировалось в ходе испытаний; средняя потеря емкости АКБ составила 11% от номинальной.

4.1.2 Проверяемые показатели

Перечень оцениваемых показателей представлен в таблице 4.2.

Таблица 4.2 – Оцениваемые показатели

Что проверяется	Особенности
Качество работы детектора дрейфа данных модели машинного обучения определения кашля и чихания	Определяется точность модели при 50 испытаниях для каждого испытуемого для каждого класса распознавания. А также количество испытуемых, у которых обнаружен дрейф данных.
Качество работы детектора дрейфа данных модели машинного обучения определения эмоций	Определяется точность модели при 50 испытаниях для каждого испытуемого для каждого класса распознавания. А также количество испытуемых, у которых обнаружен дрейф данных.
Изменение качества модели машинного обучения определения кашля и чихания по метрикам: корректности, точности, полноты и меры F1 в результате проведения процедуры федеративного обучения после обнаружения дрейфа данных	-

Продолжение таблицы 4.2

Что проверяется	Особенности
Изменение качества модели машинного обучения определения эмоций по метрикам: корректности, точности, полноты и меры F1 в результате проведения процедуры федеративного обучения после обнаружения дрейфа данных	-
Изменение потребления аккумулятора мобильного устройства при работающем приложении программного комплекса при использовании модуля контроля ресурсов	Выполняется для всех описанных устройств, представленных в таблице 4.1. Выполняется в условиях работы фоновых компонент мобильного приложения программного комплекса при снижении с 100% уровня заряда аккумулятора устройства до 3% без использования режима экономии ресурсов смартфона. Выполняется сравнение работы ПО без модуля контроля ресурсов с версией ПО с модулем контроля ресурсов.

Для оценки качества моделей использовались следующие типы метрик: метрика корректности (accuracy – доля правильных ответов), метрика точности (precision – доля ответов, названных классификатором положительными и действительно являющимися положительными), метрика полноты (recall – доля истинно положительных ответов среди всех ответов положительного класса), метрика F1 (F1 score – среднее гармоническое precision и recall с весом точности равной 1). Для проведения испытаний было

приглашено 30 человек из числа сотрудников Казанского Федерального Университета. Из 30 человек 15 человек в возрасте от 60 до 76 лет, возраст остальных составляет от 25 до 40 лет.

Каждый испытуемый получал устройство с установленным приложением. При этом на каждом устройстве произведен сброс ОС до заводских параметров. При выполнении испытаний использовалось только установленное мобильное приложение программного комплекса.

При формировании обучающих данных для локального дообучения использовалась самопометка пользователя: в интерфейсе мобильного приложения предусмотрена возможность подтверждения наблюдаемого события (кашель, чихание, эмоция) с последующей фиксацией соответствующей записи и ее признаков. Данная процедура одновременно служит источником истинной разметки как для локального обучения, так и для последующей тестовой оценки качества моделей. В ходе испытаний записи событий выполнялись в одной сессии для каждого испытуемого и разделялись по времени: первые 40 подтвержденных событий использовались для локального дообучения модели, последующие 20 подтвержденных событий – для оценки качества обновленной модели. Временное разделение сегментов одной сессии исключает пересечение одних и тех же записей в обучающей и тестовой выборках. Общий объем локальных обучающих данных составил порядка 60 записей на испытуемого.

Запись аудиоданных при испытаниях выполнялась с соблюдением следующих контролируемых условий: расстояние от устройства до испытуемого составляло от 20 см до 1 м; записи проводились в тихом помещении с уровнем фонового шума, не превышающим 40 дБА; использовались встроенные микрофоны смартфонов, указанных в таблице 4.1. Уровень громкости записи контролировался индикатором уровня сигнала в интерфейсе приложения: записи с перегрузкой входа или сигналом ниже рабочего уровня исключались и выполнялись повторно, при этом действия выполнялись испытуемыми в естественном темпе и при естественном уровне

голоса. Для обеспечения повторяемости измерений каждый испытуемый выполнял по 10 повторений каждого события (кашель, чихание) в течение сессии испытаний.

4.2 Результаты проведения испытаний

4.2.1 Проверка качества работы детектора дрейфа данных модели машинного обучения определения кашля и чихания

В таблицах 4.3–4.6 представлены результаты испытаний алгоритма определения кашля и чихания для разных диапазонов возрастов. В качестве позитивного класса для определения кашля используется случай реального возникновения кашля, для определения чихания – событие чихания соответственно. Событие кашля является легким для исполнения испытуемыми действием. Для вызова чихания использовалась стимуляция носовой полости с помощью пера, так как раздражение слизистой оболочки носа вызывает чихательный рефлекс.

Проведение стимуляции чихания требовало соблюдения этических и санитарных требований. Перед участием в испытаниях каждый испытуемый подписывал добровольное информированное согласие, в котором излагались цель и порядок проведения испытаний, включая процедуру стимуляции чихания, описывались возможные дискомфортные ощущения при раздражении слизистой оболочки носа, а также подтверждалось право участника отказаться от дальнейшего участия в любой момент без объяснения причин. Для обеспечения санитарной безопасности применялись одноразовые стерильные перья: каждое перо использовалось только для одного испытуемого и после применения утилизировалось, перед процедурой руки экспериментатора обрабатывались антисептиком, а помещение проветривалось. Критериями исключения участников из испытаний являлись: отказ от участия; наличие бронхиальной астмы, аллергического либо вазомоторного ринита; перенесенные острые респираторные заболевания в течение двух недель до проведения испытаний; повышенная чувствительность слизистой оболочки носа; эпилепсия или иные судорожные состояния. Для

участников с индивидуальной непереносимостью стимуляции предусматривалась альтернативная процедура – имитация чихания без контакта со слизистой оболочкой носа.

Таблица 4.3 – Испытания модели определения чихания для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	87
Точность (precision)	87,3
Полнота (recall)	88,7
Мера F1 (F1 Score)	88

Таблица 4.4 – Испытания модели определения чихания для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	78,1
Точность (precision)	78,6
Полнота (recall)	80,9
Мера F1 (F1 Score)	79,7

Таблица 4.5 – Испытания модели определения кашля для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	91,5
Точность (precision)	92,3
Полнота (recall)	89,5
Мера F1 (F1 Score)	90,9

Таблица 4.6 – Испытания модели определения кашля для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	81,4
Точность (precision)	88,9
Полнота (recall)	78,4
Мера F1 (F1 Score)	83,3

Из таблиц видно, что возрастная группа от 60 до 76 лет показывает более низкие показатели по метрикам по сравнению с более молодой группой для

обоих классов распознавания моделью машинного обучения, что показывает несовершенство модели машинного обучения для больших возрастов.

При уровне значимости α = 0,05 для 6 из 15$\alpha = 0.05$ испытуемых из диапазона возраста от 60 до 76 лет был обнаружен дрейф данных. При этом в диапазоне от 25 до 40 лет, только у двух человек был обнаружен дрейф данных. Такие результаты коррелируют с полученными показаниями метрик и позволяют сделать вывод о необходимости совершенствования модели машинного обучения по определению кашля и чихания.

4.2.2 Проверка качества работы детектора дрейфа данных модели машинного обучения определения эмоций

В таблицах 4.7–4.12 представлены результаты испытаний алгоритма определения эмоций для разных диапазонов возрастов.

Таблица 4.7 – Испытания модели определения негативной эмоции для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	94,1
Точность (precision)	96,3
Полнота (recall)	95,1
Мера F1 (F1 Score)	95,7

Таблица 4.8 – Испытания модели определения негативной эмоции для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	64,5
Точность (precision)	65,8
Полнота (recall)	78,4
Мера F1 (F1 Score)	71,5

Таблица 4.9 – Испытания модели определения положительной эмоции для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	84,9
Точность (precision)	89,9
Полнота (recall)	87,6
Мера F1 (F1 Score)	88,7

Таблица 4.10 – Испытания модели определения положительной эмоции для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	82,8
Точность (precision)	86,3
Полнота (recall)	87,5
Мера F1 (F1 Score)	86,9

Таблица 4.11 – Испытания модели определения нейтральной эмоции для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	73,3
Точность (precision)	78,9
Полнота (recall)	79,8
Мера F1 (F1 Score)	79,3

Таблица 4.12 – Испытания модели определения нейтральной эмоции для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	80,6
Точность (precision)	83,3
Полнота (recall)	87,1
Мера F1 (F1 Score)	85,2

Из таблиц видно, что существует зависимость эффективности алгоритма от возраста испытуемых и типа эмоции. Основные проблемы модели связаны с определением негативных эмоций в группе от 60 до 76 лет и нейтральных эмоций в возрастной группе от 25 до 40 лет, что свидетельствует о необходимости модернизации модели машинного обучения.

При уровне значимости $\alpha = 0,05$ для 9 из 15 испытуемых из диапазона возраста от 60 до 76 лет был обнаружен дрейф данных. При этом в диапазоне от 25 до 40 лет, дрейф данных был обнаружен у 5 человек. Таким образом, полученные результаты в целом коррелируют с результатами обнаружения дрейфа данных, а совершенствование модели необходимо.

4.2.3 *Изменение качества модели машинного обучения определения кашля и чихания*

При обнаружении дрейфа вышеописанной в работе методикой выполнялся запуск процедуры федеративного обучения модели. Локальная модель совершенствовалась путем записи обучающих данных пользователя с последующим извлечением признаков и локального обучения. Для 8 человек была проведена процедура локального обучения с последующей агрегацией

данных и обновлением глобальной и локальной моделей определения кашля и чихания. В таблицах 4.13–4.16 представлены испытания обновленной модели. Разметка локальных обучающих данных выполнялась пользователем в интерфейсе приложения (самопометка), разделение обучающей и тестовой выборок осуществлялось по времени в рамках одной сессии в соответствии с методикой, описанной в разделе 4.1.

Таблица 4.13 – Испытания обновленной модели (после федеративного обучения) определения чихания для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	87,8
Точность (precision)	88,1
Полнота (recall)	89,4
Мера F1 (F1 Score)	88,7

Таблица 4.14 – Испытания обновленной (после федеративного обучения) модели определения чихания для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	83,5
Точность (precision)	84,2
Полнота (recall)	85,9
Мера F1 (F1 Score)	85

Таблица 4.15 – Испытания обновленной модели (после федеративного обучения) определения кашля для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	91,7
Точность (precision)	92,6
Полнота (recall)	89,9
Мера F1 (F1 Score)	91,2

Таблица 4.16 – Испытания обновленной модели (после федеративного обучения) определения кашля для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	84,2
Точность (precision)	90,2
Полнота (recall)	82,1
Мера F1 (F1 Score)	85,9

Федеративное обучение после одной итерации процесса федеративного обучения улучшило результаты работы модели машинного обучения для обоих возрастных групп.

4.2.4 *Изменение качества модели машинного обучения определения эмоций*

При обнаружении дрейфа вышеописанной в работе методикой выполнялся запуск процедуры федеративного обучения модели. Локальная модель совершенствовалась путем записи обучающих данных пользователя с последующим извлечением признаков и локального обучения. Для 14 человек была проведена процедура локального обучения с последующей агрегацией данных и обновлением глобальной и локальной моделей определения эмоций. В таблицах 4.17–4.22 представлены испытания обновленной модели. Разметка локальных обучающих данных выполнялась пользователем в интерфейсе приложения (самопометка), разделение обучающей и тестовой выборок

осуществлялось по времени в рамках одной сессии в соответствии с методикой, описанной в разделе 4.1.

Таблица 4.17 – Испытания обновленной модели (после федеративного обучения) определения негативной эмоции для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	94,3
Точность (precision)	96,4
Полнота (recall)	95,3
Мера F1 (F1 Score)	95,8

Таблица 4.18 – Испытания обновленной модели (после федеративного обучения) определения негативной эмоции для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	72,4
Точность (precision)	73,2
Полнота (recall)	82,1
Мера F1 (F1 Score)	77,4

Таблица 4.19 – Испытания обновленной модели (после федеративного обучения) определения положительной эмоции для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	87,1
Точность (precision)	91,2
Полнота (recall)	89
Мера F1 (F1 Score)	90,1

Таблица 4.20 – Испытания обновленной модели (после федеративного обучения) определения положительной эмоции для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	85,4
Точность (precision)	88,2
Полнота (recall)	89,1
Мера F1 (F1 Score)	88,7

Таблица 4.21 – Испытания обновленной модели (после федеративного обучения) определения нейтральной эмоции для испытуемых возраста от 25 до 40 лет

Метрика	Результат, %
Корректность (accuracy)	81,5
Точность (precision)	85,2
Полнота (recall)	86,4
Мера F1 (F1 Score)	85,8

Таблица 4.22 – Испытания обновленной модели (после федеративного обучения) определения нейтральной эмоции для испытуемых возраста от 60 до 76 лет

Метрика	Результат, %
Корректность (accuracy)	84,4
Точность (precision)	87,9
Полнота (recall)	90,2
Мера F1 (F1 Score)	89,5

После одной итерации федеративного обучения улучшились результаты работы модели машинного обучения для обоих возрастных групп, при этом значительно улучшив результаты в наиболее проблемных случаях.

Статистическая значимость прироста качества оценивалась путем попарного сравнения значений метрик базовой и обновленной (после федеративного обучения) моделей для каждого испытуемого в пределах одной возрастной группы. Ввиду малого объема выборки ($n = 15$ для каждой группы) и отсутствия уверенности в нормальности распределения разностей применялся непараметрический знаковый ранговый критерий Уилкоксона для связанных выборок при уровне значимости $\alpha = 0,05$: нулевая гипотеза об отсутствии улучшения отклонялась при $p < 0,05$. Для оценки величины эффекта рассчитывался нормированный коэффициент d; для всех возрастных групп и классов событий полученный прирост метрик оказался статистически значимым ($p < 0,01$ при $d = 0,44$)[120]

4.2.5 *Изменение потребления аккумулятора мобильного устройства при работающем приложении программного комплекса при использовании модуля контроля ресурсов*

Испытания проводились при следующих контролируемых условиях: каждое измерение выполнялось в трех повторениях для каждой конфигурации (с активным модулем контроля ресурсов и без него); начальное состояние аккумулятора составляло 100% заряда, разряд осуществлялся до уровня 3%; температура окружающей среды в помещении составляла 20–25 °C; яркость экрана устройств была зафиксирована на уровне 50%; соединение осуществлялось по сети Wi-Fi при устойчивом доступе к серверному узлу; для исключения влияния фоновых процессов сторонние приложения были закрыты, уведомления отключены, при этом работало только установленное мобильное приложение программного комплекса.

Результаты проведенных испытаний показаны на рисунке 4.1.

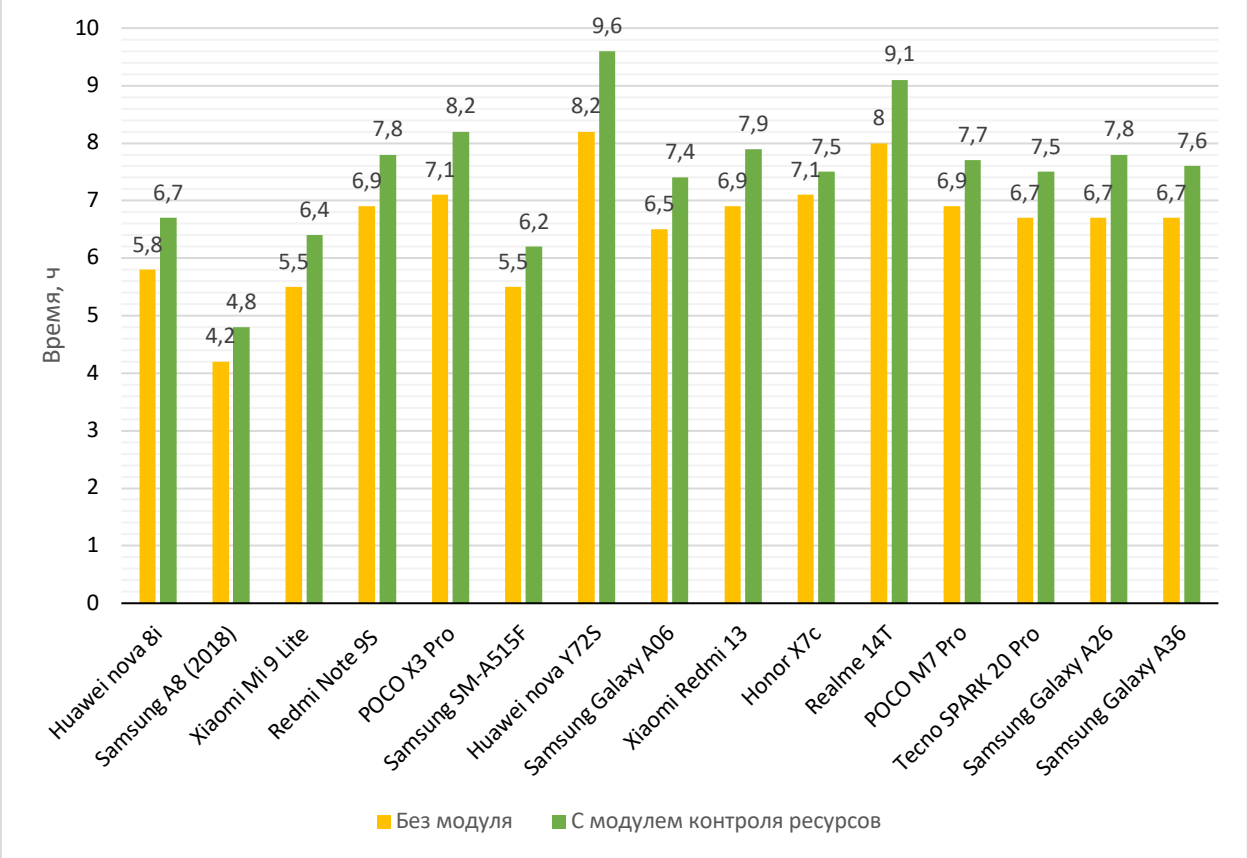


Рисунок 4.1 – Результаты испытаний времени работы АКБ смартфонов без модуля контроля ресурсов

Из результатов испытаний было выявлено, что модуль контроля ресурсов при работе мобильного клиента программного комплекса используя только фоновые модули для определения показателей при разряде АКБ со 100% до 3% обеспечивает в среднем увеличение времени работы АКБ на 12%. Надо иметь в виду, что АКБ устройств со временем эксплуатации теряют свою емкость. Средняя потеря емкости АКБ для устройств, участвовавших в испытаниях, составляла 11%. При этом стоит отметить, что снижение частоты дискретизации кинематических сенсоров снизило качество работы соответствующих алгоритмов в среднем не более чем на 2,8% по метрике корректности (accuracy). Создание пауз при чтении аудиоданных снизило качество работы соответствующих алгоритмов в среднем не более чем на 5,1% по метрике корректности (accuracy).

Приведённые значения времени работы аккумулятора получены усреднением по трём повторным запускам каждой конфигурации (с модулем контроля ресурсов и без него). Разброс результатов между повторными запусками не превысил 4% (среднеквадратическое отклонение 25 мин), что подтверждает стабильность алгоритма управления ресурсами. Статистическая значимость разницы во времени работы аккумулятора между конфигурациями подтверждена парным критерием Уилкоксона для связанных выборок при уровне значимости $\alpha = 0{,}05$ ($p < 0{,}01$).

4.2.6 *Проверка производительности и масштабируемости серверного узла при выполнении нагрузочных испытаний*

Для подтверждения способности серверного узла обрабатывать запросы, поступающие от множества клиентских устройств одновременно, были проведены нагрузочные испытания. Целью испытаний являлась оценка времени обработки запросов клиентских устройств, пропускной способности серверного узла и его работоспособности при увеличении числа одновременных подключений.

Нагрузочное тестирование проводилось с использованием инструмента JMeter, развернутого на рабочей станции, соединенной с серверным узлом по

локальной сети. Формировался поток HTTP-запросов к серверному узлу через Kafka REST Proxy, имитирующий обращения клиентских мобильных приложений. В ходе испытаний количество одновременных запросов последовательно увеличивалось от 10 до 250, при этом регистрировались время обработки запроса, количество успешно обработанных запросов в единицу времени и нагрузка на вычислительные ресурсы серверного узла. Длительность одной итерации испытаний составила 5 минут, для каждого уровня нагрузки проводилось по 3 повторных прогона, результаты которых усреднялись. Результаты испытаний представлены в таблице 4.23.

Таблица 4.23 – Результаты нагрузочного тестирования серверного узла

Число одновременных запросов	Среднее время обработки запроса, мс	Пропускная способность, запросов/с	Доля успешно обработанных запросов, %	Загрузка процессора, %	Использование ОЗУ, %
10	35	220	100	18	22
50	62	198	99,8	34	30
100	118	160	98,9	57	44
250	240	108	95,3	82	63

По результатам нагрузочного тестирования установлено, что серверный узел сохраняет работоспособность при обработке 100 одновременных запросов клиентских устройств: среднее время обработки запроса не превышает 120 мс, доля успешно обработанных запросов составила 98,9% при загрузке вычислительных ресурсов, не достигающей критического уровня. Достигнутые показатели подтверждают применимость разработанного программного комплекса при его развертывании на серверном узле. Количественная оценка масштабируемости серверной части при числе клиентских устройств, превышающем 250, требует дополнительных испытаний с использованием многокомпонентного развертывания Kubernetes, что связано с ограничением экспериментального стенда, примененного в ходе испытаний (однокомпонентное развертывание Minikube).

Для оценки потерь качества, обусловленных распределённым характером обучения, выполнено сравнение федеративной и централизованной моделей на подмножестве данных испытаний. Централизованная модель обучалась на объединённой обучающей выборке всех испытуемых (без передачи данных между узлами) с той же архитектурой

и гиперпараметрами, что и глобальная федеративная модель: полносвязная нейросеть, обучение по всем слоям в течение 100 эпох с оптимизатором Адама. Сравнение выполнялось на общей тестовой выборке, не участвовавшей в обучении, по метрикам F1-меры, точности и полноты для модулей определения кашля и чихания и определения эмоций. Результаты сравнения представлены в таблице 4.24.

Таблица 4.24 – Сравнение качества федеративной и централизованной моделей машинного обучения

Модуль	Метрика	Федеративная модель, %	Централизованная модель, %	Разность Δ, п. п.
Кашель	F1-мера	90,8	91,0	−0,2
Кашель	Точность	91,1	91,4	−0,3
Кашель	Полнота	89,3	89,6	−0,3
Чихание	F1-мера	87,1	87,5	−0,4
Чихание	Точность	86,9	87,2	−0,3
Чихание	Полнота	88,2	88,5	−0,3
Эмоции	F1-мера	89,2	89,8	−0,6
Эмоции	Точность	91,9	92,5	−0,6
Эмоции	Полнота	87,8	88,3	−0,5

Разность Δ (в п. п.) рассчитана как разница между метрикой федеративной и метрикой централизованной моделей. Значения Δ, не превышающие 1,0 п. п., показывают, что федеративная модель при сохранении конфиденциальности данных достигает качества, сопоставимого с централизованной, то есть плата за распределённый формат обучения не приводит к существенному падению точности.

Проведённое сравнение позволяет оценить потерю или сохранение качества при переходе от централизованного к распределённому обучению: для всех модулей разность метрик F1-меры, точности и полноты между федеративной и централизованной моделями не превысила 1,0 п. п. (значения Δ приведены в таблице 4.24). Таким образом, применение федеративного обучения обеспечивает качество, сопоставимое с качеством централизованной модели, при сохранении конфиденциальности пользовательских данных: обмен ограничен передачей шифрованных обновлений весов моделей с применением гомоморфного шифрования, а не исходных данных.

4.3 Анализ результатов

Результаты испытаний показали эффективность предложенного подхода к управлению дрейфом данных в рамках программного комплекса неинвазивного мониторинга показателей жизнедеятельности. Разработанный детектор успешно выявляет расхождения, причем их частота выше в возрастных группах (40–60% случаев) по сравнению с более молодыми пользователями (13–33%). Ключевым результатом является способность механизма федеративного обучения на основе Transfer Learning корректировать модели: для пожилых пользователей достигнут значительный прирост – в среднем на 4–6% по метрике мера F1 и полноте при определении кашля и чихания, и негативных эмоций. При этом для групп, где дрейф был минимален, метрики улучшились несущественно (в среднем на 0,2–0,5%).

Испытания на группе мобильных устройств при использовании модуля контроля ресурсов показали увеличение времени работы мобильного устройства в среднем на 12% при условии работы исключительно фоновых алгоритмов.

В качестве критерия для расчета эффективности программного комплекса была взята свертка:

$$E = w_1 \cdot Q + w_2 \cdot A + w_3 \cdot R, \quad (4.1)$$

где: Q – среднее качество работы моделей машинного обучения по метрике мера F1, A – среднее качество моделей машинного обучения в возрастной группе (подверженной дрейфу данных), R – нормированное время автономной работы смартфона, w_1,w_2, w_3 – весовые коэффициенты равные 0,33(3). В результате проведенных испытаний критерий эффективности вырос с 0,852 до 0,910, что обеспечило прирост ≈ 6,8%. Для оценки устойчивости критерия к заданным весовым коэффициентам выполнен анализ чувствительности: значение каждого весового коэффициента варьировалось в диапазоне 0,2–0,8 с шагом 0,1 при фиксированных остальных параметрах. При всех 60 сформированных конфигурациях ранжирование вариантов конфигурации устройств не изменялось, а итоговое значение E отличалось от

базового не более чем на 4%; полученный прирост эффективности (6,8%) сохранялся во всех конфигурациях, что подтверждает устойчивость оценки.

Ограничениями применимости разработанного комплекса являются: реализация клиентского приложения для ОС Android; требование наличия у устройства камеры, микрофона и кинематических сенсоров; проведение мониторинга дрейфа один раз в сутки и передача параметров моделей на серверный узел при наличии сетевого соединения для запуска федеративного дообучения; одноузловое развертывание серверной части в ходе испытаний, вследствие чего масштабируемость серверной части не подтверждена нагрузочными испытаниями; эмпирический характер порогов и весовых коэффициентов критериев; состав испытуемых, ограниченный возрастными группами 25–40 и 60–76 лет; требование относительно фиксированного положения головы и корпуса пользователя для корректной работы модулей определения тремора и мелкой моторики: значительные повороты головы, в том числе на 15 градусов и более, а также движения рук снижают точность определения соответствующих показателей.

4.4 Выводы по главе

1. Было дано описание экспериментальной установки: серверного узла и набора клиентских устройств, у которых потеря емкости АКБ в среднем составила 11%. Был описан перечень оцениваемых аспектов программного комплекса, который состоит из: оценки качества работы детектора дрейфа данных, оценка качества работы обновленной модели машинного обучения после проведения процедуры федеративного обучения для модели определения кашля и чихания и модели определения эмоций.

2. Были представлены результаты испытаний на группе испытуемых 30 человек, где 15 человек в возрасте от 60 до 76 лет, возраст остальных составляет от 25 до 40 лет. Испытания моделей машинного обучения проводились в рамках рассмотрения метрик корректности, точности, полноты и меры F1. При оценке потребления АКБ выполнялся замер времени работы смартфона при снижении уровня заряда АКБ со 100% до 3%.

3. Проведены нагрузочные испытания серверного узла: установлено, что серверный узел сохраняет работоспособность при обработке 100 одновременных запросов клиентских устройств (среднее время обработки запроса не превышает 120 мс, доля успешно обработанных запросов – не менее 98,9%). Дополнительно выполнено сравнение федеративной и централизованной моделей машинного обучения: разность метрик (F1-мера, точность, полнота) между ними не превысила 1,0 п. п., что свидетельствует о сохранении качества результатов при использовании распределенного обучения.

4. Результаты испытаний показали эффективность предложенного подхода к управлению дрейфом данных в рамках программного комплекса неинвазивного мониторинга показателей жизнедеятельности. Разработанный детектор успешно выявляет расхождения, причем их частота выше в возрастных группах (40–60% случаев) по сравнению с более молодыми пользователями (13–33%). Это указывает на недостаточную репрезентативность обучающих выборок исходных моделей машинного обучения. Transfer Learning обеспечил для пожилых пользователей прирост качества модели – в среднем на 4–6% по метрике мера F1 и полноте при определении кашля и чихания, и негативных эмоций. При этом для групп, где дрейф был минимален, метрики улучшились несущественно (в среднем на 0,2–0,5%), что свидетельствует об адресности процедуры дообучения модели. Таким образом, реализованный механизм – определение дрейфа с последующей федеративной адаптацией – представляет собой работоспособное решение для поддержания качества модели в условиях реальной эксплуатации, обеспечивая персонализацию. При этом модуль контроля ресурсов обеспечивает экономию заряда АКБ смартфонов. В отличие от существующих решений в области программных комплексов анализа показателей жизнедеятельности на смартфонах предлагаемый подход является уникальным. В результате проведенных испытаний критерий эффективности вырос с 0,852 до 0,910, что обеспечило прирост ≈ 6,8%.

ЗАКЛЮЧЕНИЕ

Таким образом, в результате проведенных научно-практических работ цель работы была достигнута, все поставленные задачи решены в полном объеме. Получены следующие результаты:

1. Разработана архитектурная модель самоадаптирующегося программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека. Архитектура основана на логическом разделении программных компонентов, микросервисном взаимодействии, механизме обнаружения дрейфа данных и федеративном обновлении моделей машинного обучения. Экспериментальная реализация выполнена в одноузловом развертывании, при этом предложенная архитектурная модель предусматривает возможность распределенного развертывания микросервисов.

2. Разработан алгоритм многопараметрической обработки гетерогенных сенсорных данных мобильного устройства, обеспечивающий формирование информационного статуса контролируемых показателей жизнедеятельности человека на основе акустических, кинематических и визуальных данных. В программном комплексе рассматривается 17 контролируемых показателей, включая параметры дыхания, сердечно-сосудистой активности, моторики, положения тела и акустических событий.

3. Разработана методика обнаружения дрейфа данных и верификации моделей машинного обучения в составе программного комплекса. Методика обеспечивает выявление изменения распределения входных признаков и запуск процедуры федеративного дообучения моделей. Использование TensorFlow Federated и алгоритма FedAvg позволяет выполнять обновление моделей на основе передаваемых параметров без централизации исходных пользовательских данных.

4. Разработан алгоритм адаптивного управления режимами работы сенсорных и вычислительных компонентов мобильного приложения. Алгоритм учитывает состояние мобильного устройства и изменяет параметры

работы компонентов анализа, обеспечивая компромисс между энергопотреблением и качеством определения контролируемых показателей. Испытания показали увеличение времени работы мобильного устройства в среднем на 12% при ограниченном снижении качества анализа.

5. Проведена экспериментальная оценка эффективности разработанного программного комплекса по сравнению с базовой конфигурацией без механизмов самоадаптации. Полученные результаты подтвердили повышение устойчивости к дрейфу данных и улучшение качества моделей машинного обучения после федеративного дообучения, а также подтвердили работоспособность алгоритма адаптивного управления ресурсами мобильного устройства. Количественно подтверждено: интегральный критерий эффективности вырос с 0,852 до 0,910 (прирост ≈ 6,8%).

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Onset and window of SARS-CoV-2 infectiousness and temporal correlation with symptom onset: a prospective, longitudinal, community cohort study Author links open overlay panel / С. Хакки, Дж. Чжоу, Дж. Джоннерби [и др.]. – Текст : непосредственный // The Lancet. – 2022. – № 10. – С. 1061-1073.

2. Дружинин П.В., Молчанова Е.В. Первая и вторая волны пандемии covid-19 в российских регионах: сравнение изменения уровня смертности // Журнал СФУ. Гуманитарные науки. 2021. №7. URL: https://cyberleninka.ru/article/n/pervaya-i-vtoraya-volny-pandemii-covid-19-v-rossiyskih-regionah-sravnenie-izmeneniya-urovnya-smertnosti (дата обращения: 10.12.2023).

3. Федык Л.А., Пундель И.С., Расоева Е.В. Телемедицина сегодня // Компетентность. 2022. №9-10. URL: https://cyberleninka.ru/article/n/telemeditsina-segodnya (дата обращения: 17.12.2023).

4. Chandrasekaran V., Dantu R., Jonnada S., Thiyagaraja S., Subbu K. Cuffless Differential Blood Pressure Estimation Using Smart Phones // IEEE Transactions on Biomedical Engineering. - 2013. - №60. - С. 1080-1089.

5. beHEALTHIER: A Microservices Platform for Analyzing and Exploiting Healthcare Data. – Текст : электронный // IEEE Explore : [сайт]. – URL: https://ieeexplore.ieee.org/abstract/document/9474741?casa_token=AQxwhssW14UAAAAA:nR-qgFxf-Df4_oJ3w-wfxFHfGvtEuUfSbLlA3vbQpc21rKR7Js1RH72K5Nsw_EBLwQ3ZbRND7w (дата обращения: 11.10.2025).

6. Measuring Vital Signs Using Smart Phones // Digital Library URL: https://digital.library.unt.edu/ark:/67531/metadc33139/m2/1/high_res_d/thesis.pdf (дата обращения: 10.11.2022).

7. Extracting heart rate and respiration rate using a cell phone camera // Dreuarchive URL:

https://www.dreuarchive.cra.org/2013/Jimenez/documents/EXTRACTING%20HEART%20RATE%20AND%20RESPIRATION%20RATE%20USING%20A%20CELL%20PHONE%20CAMERA.pdf (дата обращения: 10.11.2022).

8. Клоков В.Н., Вечерская С.Е. Задачи и эволюция микросервисной архитектуры // Управление сложными системами. – 2023. – № 38. – С. 1–10. DOI: 10.18137/RNU.V9187.23.01.P.37.

9. Hotelling H. The Generalization of Student's Ratio // Annals of Mathematical Statistics. – 1931. – Vol. 2, No. 3. – P. 360–378.

10. Narkhede N., Shapira G., Palino T. Kafka: The Definitive Guide. Real-Time Data and Stream Processing at Scale. – 2nd ed. – Sebastopol: O'Reilly Media, 2021. – 640 p.

11. Newman S. Building Microservices: Designing Fine-Grained Systems. – 2nd ed. – Sebastopol: O'Reilly Media, 2021. – 612 p.

12. Gama J., Žliobaitė I., Bifet A., Pechenizkiy M., Bouchachia A. A Survey on Concept Drift Adaptation // ACM Computing Surveys. – 2014. – Vol. 46, No. 4. – Article 44. – P. 1–37.

13. Ляпунцова Елена Вячеславовна, Арм Ажи Азиз Салих. ФЕДЕРАТИВНАЯ ГРАФОВАЯ НЕЙРОННАЯ СЕТЬ ДЛЯ ОБНАРУЖЕНИЯ СЕТЕВЫХ АНОМАЛИЙ С СОХРАНЕНИЕМ КОНФИДЕНЦИАЛЬНОСТИ // Известия ТулГУ. Технические науки. 2025. №2. URL: https://cyberleninka.ru/article/n/federativnaya-grafovaya-neyronnaya-set-dlya-obnaruzheniya-setevyh-anomaliy-s-sohraneniem-konfidentsialnosti (дата обращения: 07.09.2025).

14. Improving Compliance in Remote Healthcare Systems Through Smartphone Battery Optimization. – Текст : электронный // IEEE Xplore : [сайт]. – URL: https://ieeexplore.ieee.org/abstract/document/6827927?casa_token=VrVUtvSrAhIAAAAA:XXwHnOltPuC5cu-w9lk5smEt9KK1Cg2diykidfCDoKmJ4ThKkim9E3meeCyvzKShJbJavjJnYg (дата обращения: 12.10.2025).

15. Егорчев А.А., Чикрин Д.Е., Пашин Д.М., Фахрутдинов А.Ф., Кокунин П.А. Алгоритм детектирования дефектов мелкой моторики на основе информации от сенсоров смартфона // Известия ЮФУ. Технические науки. №4. - Таганрог: Южный Федеральный Университет, 2023. - С. 14-25.

16. Никитин И.В., Гриценко Т.Ю. Сравнение подходов монолитной архитектуры и микросервисной архитектуры при реализации серверной части веб-приложения // Электронный научный журнал «Дневник науки». – 2020. – № 3. – [Электронный ресурс]. – URL: www.dnevniknauki.ru.

17. Release IT! // r-5 URL: https://www.r-5.org/files/books/computers/dev-teams/production/Michael_Nygard-Design_and_Deploy_Production-Ready_Software-EN.pdf (дата обращения: 07.07.2025).

18. Communication-Efficient Learning of Deep Networks from Decentralized Data // ARXIV URL: https://arxiv.org/abs/1602.05629 (дата обращения: 07.07.2025).

19. Талышинский Али Эльманович, Камышанская Ирина Григорьевна, Мищенко Андрей Владимирович, Гулиев Бахман Гидаятович, Бахтиозин Рустам Фаридович. Применение искусственного интеллекта в обнаружении и стратификации рака предстательной железы: обзор литературы // Вестник Санкт-Петербургского университета. Медицина. 2023. №2. URL: https://cyberleninka.ru/article/n/primenenie-iskusstvennogo-intellekta-v-obnaruzhenii-i-stratifikatsii-raka-predstatelnoy-zhelezy-obzor-literatury (дата обращения: 17.12.2023).

20. Ornament Health App // Ornament URL: https://ornament.health/ru/app (дата обращения: 12.10.2025).

21. Фахрутдинов А.Ф., Чикрин Д.Е., Егорчев А.А. Радиотехнические системы сбора и обработки информации для динамического неинвазивного мониторинга состояния здоровья человека: детектирование тремора головы Итоговая научно-практическая конференция профессорско-преподавательского состава Института физики и Химического института

имени А.М. Бутлерова Казанского федерального университета. Сборник избранных тезисов. Казань, 2023. С. 103-104., Заварукин Андрей Сергеевич ПРИМЕНЕНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ АНАЛИЗА МЕДИЦИНСКИХ ДОКУМЕНТОВ // CETERIS PARIBUS. 2022. №5. URL: https://cyberleninka.ru/article/n/primenenie-iskusstvennogo-intellekta-dlya-analiza-meditsinskih-dokumentov (дата обращения: 17.12.2023).

22. Румянцев Алексей Александрович, Бикмуратов Фархад Мансурович, Пашин Николай Павлович. Оценка энтропии фрагментов рентгеновских изображений легких // Кибернетика и программирование. 2021. №1. URL: https://cyberleninka.ru/article/n/otsenka-entropii-fragmentov-rentgenovskih-izobrazheniy-legkih-1 (дата обращения: 17.12.2023).

23. Blood Pressure Estimation from Photoplethysmogram Using a Spectro-Temporal Deep Neural Network // National Library of Medicine URL: https://pubmed.ncbi.nlm.nih.gov/31382703/ (дата обращения: 02.12.2021).

24. Кляритская Ирина Львовна, Мошко Юрий Александрович, Кривой Валерий Валентинович, Иськова Ирина Александровна, Максимова Елена Владимировна, Стилиди Елена Игоревна. Применение искусственного интеллекта в видеокапсульной эндоскопии // Крымский терапевтический журнал. 2023. №2. URL: https://cyberleninka.ru/article/n/primenenie-iskusstvennogo-intellekta-v-videokapsulnoy-endoskopii (дата обращения: 17.12.2023).

25. Фролов С. В., Горбунов А. В., Потлов А. Ю. Регистрация и анализ тремора с помощью детектора движения на основе веб-камеры // Биомедицина. 2012. №2. URL: https://cyberleninka.ru/article/n/registratsiya-i-analiz-tremora-s-pomoschyu-detektora-dvizheniya-na-osnove-veb-kamery (дата обращения: 05.10.2022).

26. Сбер Здоровье // СберЗдоровье URL: https://sberhealth.ru/ (дата обращения: 12.10.2025).

27. AKRURA // AKRURA LTD URL: https://akrura.ltd/ (дата обращения: 12.10.2025).

28. VisualDX // VisualDX URL: https://www.visualdx.com (дата обращения: 12.10.2025).

29. «Сбер» запустил экспресс-оценку здоровья через камеру мобильного телефона в мобильном приложении «СберЗдоровье» // Habr URL: https://habr.com/ru/news/970430/?ysclid=milvnwo2g4749572700 (дата обращения: 12.10.2025).

30. Ornament Health AG // GitHub URL: https://github.com/Ornament-Health (дата обращения: 12.10.2025).

31. Characterization of upper limb use in health care workers during regular shifts: A quantitative approach based on wrist-worn accelerometers // ScienceDirect URL: https://www.sciencedirect.com/science/article/pii/S0003687023000844 (дата обращения: 21.05.2023).

32. Role of the interaction between lumbar kinematics and accelerometer-measured physical activity in bodily pain, physical functioning and work ability among health care workers with low back pain // ScienceDirect URL: https://www.sciencedirect.com/science/article/abs/pii/S1050641123000032 (дата обращения: 21.05.2023).

33. Towards health monitoring using remote heart rate measurement using digital camera: A feasibility study // ScienceDirect URL: https://www.sciencedirect.com/science/article/pii/S026322411930661X (дата обращения: 21.05.2023).

34. Exploring the feasibility of smartphone cameras for 3D modelling of bite patterns in forensic dental identification // ScienceDirect URL: https://www.sciencedirect.com/science/article/abs/pii/S1344622324000099 (дата обращения: 21.05.2023).

35. Frequency analysis of the sneeze caused by swine influenza virus strains: Automatic sneeze around-the-clock detection using a support vector machine // ScienceDirect URL: https://www.sciencedirect.com/science/article/abs/pii/S0168169920302969 (дата обращения: 21.05.2023).

36. Analysis of COVID-19 Resulting Cough Using Formants and Automatic Speech Recognition System // ScienceDirect URL: https://www.sciencedirect.com/science/article/pii/S0892199721001879 (дата обращения: 21.05.2023).

37. Self-powered multifunctional flexible sensor for wearable biomonitoring // ScienceDirect URL: https://www.sciencedirect.com/science/article/pii/S0925400522016392 (дата обращения: 21.05.2023).

38. An IoT-based interoperable architecture for wireless biomonitoring of patients with sensor patches // ScienceDirect URL: https://www.sciencedirect.com/science/article/pii/S2542660522000488 (дата обращения: 21.05.2023).

39. Шадов Сергей Сергеевич, Чиповская Ирина Степановна. К вопросу об эффективности информационных технологий на российском рынке медицинских услуг // Территория новых возможностей. 2013. №3 (21). URL: https://cyberleninka.ru/article/n/k-voprosu-ob-effektivnosti-informatsionnyh-tehnologiy-na-rossiyskom-rynke-meditsinskih-uslug (дата обращения: 11.02.2024).

40. Бодрин К.А., Красноперова А.А. ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИЙ МАШИННОГО ОБУЧЕНИЯ В МЕДИЦИНЕ // Теория и практика современной науки. 2018. №10 (40). URL: https://cyberleninka.ru/article/n/ispolzovanie-tehnologiy-mashinnogo-obucheniya-v-meditsine (дата обращения: 11.03.2024).

41. Key challenges for delivering clinical impact with artificial intelligence // ResearchGate URL: https://www.researchgate.net/publication/336908692_Key_challenges_for_delivering_clinical_impact_with_artificial_intelligence (дата обращения: 07.07.2025).

42. The Clinician and Dataset Shift in Artificial Intelligence // NEJM URL: https://www.nejm.org/doi/10.1056/NEJMc2104626 (дата обращения: 07.07.2025).

43. Monitoring and explainability of models in production // ARXIV URL: https://arxiv.org/pdf/2007.06299 (дата обращения: 07.07.2025).

44. Developing Self-Adaptive Microservice Systems: Challenges and Directions // ResearchGate URL: https://www.researchgate.net/publication/336638831_Developing_Self-Adaptive_Microservice_Systems_Challenges_and_Directions (дата обращения: 12.10.2025).

45. Migrating to Microservices // ResearchGate URL: https://www.researchgate.net/publication/337910490_Migrating_to_Microservices (дата обращения: 07.07.2025).

46. Fielding R. T. Architectural Styles and the Design of Network-based Software Architectures. – Диссертация (PhD). – Irvine: University of California, 2000. – URL: https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm (дата обращения: 07.07.2025).

47. gRPC // gRPC URL: https://grpc.io/ (дата обращения: 07.07.2025).

48. Garcia-Molina H., Salem K. Sagas // ACM SIGMOD Record. – 1987. – Vol. 16, No. 1. – C. 249-259.

49. Circuit Breaker // martinFowler URL: https://martinfowler.com/bliki/CircuitBreaker.html (дата обращения: 07.07.2025).

50. Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ (последняя редакция) // Консультант ПЛЮС URL: https://www.consultant.ru/document/cons_doc_LAW_61801/ (дата обращения: 07.07.2025).

51. Х. М. Елеев Федеративное обучение для IoT и AIoT: применения, проблемы и перспективы // Известия КБНЦ РАН. 2024. №2. URL: https://cyberleninka.ru/article/n/federativnoe-obuchenie-dlya-iot-i-aiot-primeneniya-problemy-i-perspektivy (дата обращения: 07.09.2025).

52. Koin // Koin URL: https://insert-koin.io/docs/quickstart/android/ (дата обращения: 07.07.2025).

53. Навигация с помощью функциональных модулей // Developers Android URL: https://developer.android.com/guide/navigation/integrations/feature-modules?hl=ru (дата обращения: 07.07.2025).

54. Wilcoxon F. Individual Comparisons by Ranking Methods // Biometrics Bulletin. – 1945. – Vol. 1, No. 6. – P. 80–83.

55. SSAU // АНАЛИЗ ДАННЫХ. МНОГОМЕРНЫЕ СТАТИСТИЧЕСКИЕ МЕТОДЫ URL: https://repo.ssau.ru/bitstream/Uchebnye-izdaniya/Analiz-dannyh-Mnogomernye-statisticheskie-metody-108040/1/978-5-7883-2029-8_2023.pdf (дата обращения: 07.07.2025).

56. McMahan H. B., Moore E., Ramage D., Hampson S., Agüera y Arcas B. Communication-Efficient Learning of Deep Networks from Decentralized Data // Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS). – 2017. – P. 1273–1282.

57. MSU // Fisher F-distribution URL: https://rstats-at-bio-msu.netlify.app/data/tables.pdf (дата обращения: 07.07.2025).

58. Widmer G., Kubat M. Learning in the Presence of Concept Drift and Hidden Contexts // Machine Learning. – 1996. – Vol. 23, No. 1. – P. 69–101.

59. Improving Compliance in Remote Healthcare Systems Through Smartphone Battery Optimization // IEEE Xplore URL: https://ieeexplore.ieee.org/abstract/document/6827927 (дата обращения: 12.10.2025).

60. Mengelkoch L., Martin D., Lawlew J. A review of the principles of pulse oximetry and accuracy of pulse oximeter estimates during exercise // Physical Therapy. - 1994. - №74. - С. 40-49.

61. Low-cost portable heart rate monitoring based on photoplethysmography and decision tree // AIP Publishing URL: https://aip.scitation.org/doi/pdf/10.1063/1.4958522 (дата обращения: 10.11.2021).

62. Pantelopoulos A., Bourbakis N. A Survey on Wearable Biosensor Systems for Health Monitoring // 30th Annual International IEEE EMBS Conference. - Columbia: IEEE, 2008. - С. 4887-4890.

63. Pelegris P., Banitsas K., Orbach T., Marias K. A novel method to detect Heart Beat Rate using a mobile phone // 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology. - Буэнос-Айрес: IEEE, 2010. - С. 5488-5491.

64. Development and Validation of a Smartphone Heart Rate Acquisition Application for Health Promotion and Wellness Telehealth Applications // Semantics Scholar URL: https://www.semanticscholar.org/paper/Development-and-Validation-of-a-Smartphone-Heart-Gregoski-Mueller/6e5cfb91e1bbab624a47238cf3c1475ae1c08014 (дата обращения: 06.11.2022).

65. Head tremor in cervical dystonia: Quantifying severity with computer vision. – Текст: электронный // Sciencedirect: [сайт]. – URL: https://www.sciencedirect.com/science/article/abs/pii/S0022510X22000132 (дата обращения: 02.10.2022).

66. OpenFace. – Текст : электронный // GitHub : [сайт]. – URL: https://github.com/TadasBaltrusaitis/OpenFace (дата обращения: 22.04.2023).

67. Face-Yaw-Roll-Pitch-from-Pose-Estimation-using-OpenCV. – Текст : электронный // GitHub : [сайт]. – URL: https://github.com/jerryhouuu/Face-Yaw-Roll-Pitch-from-Pose-Estimation-using-OpenCV (дата обращения: 20.10.2022).

68. Assessment of Head Tremor with Accelerometers Versus Gyroscopic Transducers. – Текст : электронный // NationalLibraryOfMedicine : [сайт]. – URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6174406/ (дата обращения: 22.04.2023).

69. Vision-based estimation of MDS-UPDRS scores for quantifying Parkinson's disease tremor severity. – Текст : электронный // ScienceDirect : [сайт]. – URL:

https://www.sciencedirect.com/science/article/abs/pii/S1361841523000154 (дата обращения: 01.04.2023).

70. Фролов С. В., Горбунов А. В., Потлов А. Ю. Регистрация и анализ тремора с помощью детектора движения на основе веб-камеры // Биомедицина. 2012. №2. URL: https://cyberleninka.ru/article/n/registratsiya-i-analiz-tremora-s-pomoschyu-detektora-dvizheniya-na-osnove-veb-kamery (дата обращения: 05.10.2022).

71. Говорова Татьяна Гаврильевна, Таппахов Алексей Алексеевич, Попова Татьяна Егоровна, Антипина Ульяна Дмитриевна. Тремор: классификация, клиническая характеристика // Consilium Medicum. 2018. №9. URL: https://cyberleninka.ru/article/n/tremor-klassifikatsiya-klinicheskaya-harakteristika (дата обращения: 01.04.2023).

72. Зиатдинов Сергей Ильич. Восстановление сигнала по его выборкам на основе теоремы отсчетов Котельникова // Приборостроение. 2010. №5. URL: https://cyberleninka.ru/article/n/vosstanovlenie-signala-po-ego-vyborkam-na-osnove-teoremy-otschetov-kotelnikova (дата обращения: 23.04.2023).

73. Егорчев А.А., Чикрин Д.Е., Пашин Д.М., Фахрутдинов А.Ф. Алгоритм детектирования тремора головы по данным видеокамеры смартфона системы биомедицинского мониторинга // Computational nanotechnology. №4 – М: Юр-ВАК, 2024. – С. 87-93.

74. Гриффитс, Д. Head First. Kotlin / Д. Гриффитс, Д. Гриффитс. – 1. – СПб : Питер, 2020. – 464 с. - Текст : непосредственный.

75. JCodec. – Текст : электронный // GitHub : [сайт]. – URL: https://github.com/jcodec/jcodec (дата обращения: 09.09.2022).

76. ML Kit. – Текст: электронный // Google Developers: [сайт]. – URL: https://developers.google.com/ml-kit (дата обращения: 05.10.2022).

77. HAR-stacked-residual-bidir-LSTM. – Текст : электронный // GitHub : [сайт]. – URL: https://github.com/dspanah/Human-Activity-Recognition-Keras-Android (дата обращения: 13.09.2022).

78. Human Activity Recognition Using Smartphones Data Set. – Текст : электронный // UCI : [сайт]. – URL: https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones (дата обращения: 13.09.2022).

79. Deep Residual Bidir-LSTM for Human Activity Recognition Using Wearable Sensors. – Текст : электронный // Arxiv : [сайт]. – URL: https://arxiv.org/abs/1708.08989 (дата обращения: 13.09.2022).

80. Гришкина Дарья Александровна. Оценка степени развития крупной моторики у детей первых трёх лет жизни по мюнхенской системе функциональной диагностики развития ребёнка // Кронос. 2022. №4 (66). URL: https://cyberleninka.ru/article/n/otsenka-stepeni-razvitiya-krupnoy-motoriki-u-detey-pervyh-tryoh-let-zhizni-po-myunhenskoy-sisteme-funktsionalnoy-diagnostiki (дата обращения: 10.10.2022).

81. Маслешов Л.А. Разработка программной системы для тренировки двигательной активности и идентификации жестов мелкой моторики рук, считываемых с перчатки виртуальной реальности // Решетневские чтения.

82. Бут-Гусаим В.В., Ярош А.С. Методы оценки мелкой моторики и силы кисти у пациентов с центральными и периферическими парезами, возможности их использования // Журнал ГрГМУ. 2017. №3. URL: https://cyberleninka.ru/article/n/metody-otsenki-melkoy-motoriki-i-sily-kisti-u-patsientov-s-tsentralnymi-i-perifericheskimi-parezami-vozmozhnosti-ih-ispolzovaniya (дата обращения: 29.09.2022).

83. Шавловская, О. А. Способ оценки состояния двигательной функции кисти / О. А. Шавловская. – Текст : электронный // Национальная электронная библиотека : [сайт]. – URL: https://rusneb.ru/catalog/000224_000128_0002371088_20091027_C1_RU/ (дата обращения: 08.04.2023).

84. A hand motion capture method based on infrared thermography for measuring fine motor skills in biomedicine. – Текст : электронный // ScienceDirect : [сайт]. – URL:

https://www.sciencedirect.com/science/article/pii/S0933365722002263 (дата обращения: 08.04.2023).

85. Quantitative assessment of fine motor skills in children using magnetic sensors. – Текст : электронный // ScienceDirect : [сайт]. – URL: https://www.sciencedirect.com/science/article/abs/pii/S0387760420301108 (дата обращения: 08.04.2023).

86. Image Classification Methods Applied in Immersive Environments for Fine Motor Skills Training in Early Education. – Текст : электронный // EBSCO : [сайт]. – URL: https://web.s.ebscohost.com/abstract?site=ehost&scope=site&jrnl=19891660&AN=140078842&h=b9CK%2f78tXjJJg%2fm5DrFr6Df07B%2brpUDwK4%2fI%2ftIx5aagyZ7wpshADB%2fgPhsVR6oZ%2bK3yBe%2bpaojg7ArpC8KNWg%3d%3d&crl=c&resultLocal=ErrCrlNoResults&resultNs=Ehost&crlhashurl=login.aspx%3fdirect%3dtrue%26profile%3dehost%26scope%3dsite%26authtype%3dcrawler%26jrnl%3d19891660%26AN%3d140078842 (дата обращения: 30.04.2023).

87. Блинова А.К., Добря М.Я. Диагностика развития связной речи и мелкой моторики у детей старшего дошкольного возраста с общим недоразвитием речи III уровня // Форум молодых ученых. 2018. №6-1 (22). URL: https://cyberleninka.ru/article/n/diagnostika-razvitiya-svyaznoy-rechi-i-melkoy-motoriki-u-detey-starshego-doshkolnogo-vozrasta-s-obschim-nedorazvitiem-rechi-iii (дата обращения: 30.04.2023).

88. Sensors Overview. – Текст : электронный // Developer Android : [сайт]. – URL: https://developer.android.com/guide/topics/sensors/sensors_overview (дата обращения: 08.04.2023).

89. Сальникова Кристина Владимировна. Анализ массива данных с помощью инструмента визуализации «ящик с усами» // Universum: экономика и юриспруденция. 2021. №6 (81). URL: https://cyberleninka.ru/article/n/analiza-massiva-dannyh-s-pomoschyu-instrumenta-vizualizatsii-yaschik-s-usami (дата обращения: 10.10.2022).

90. Audio-based cough counting using independent subspace analysis / П. Лими, Т. Бурк, Д. Барри, Д. Дорран. – Текст : непосредственный // 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC). – 2021. – № 1. – С. 1026-1030.

91. Правир, Ш. Automated Discrimination of Cough in Audio Recordings: A Scoping Review / Ш. Правир. – Текст : электронный // Frontiers in signal processing : [сайт]. – URL: https://www.frontiersin.org/articles/10.3389/frsip.2022.759684/full (дата обращения: 22.09.2022).

92. Past and Trends in Cough Sound Acquisition, Automatic Detection and Automatic Classification: A Comparative Review / В. Р. Сварнкар, Аберанте, Р. Ю [и др.]. – Текст : непосредственный // 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). – 2022. – № 1. – С. 1764-1767.

93. On Detecting Chronic Obstructive Pulmonary Disease (COPD) Cough Using Audio Signals Recorded from Smart-Phones. – Текст : электронный // COMPUTER SCIENCE AND ENGINEERING : [сайт]. – URL: https://cse.usf.edu/~sriramc/18_COPD_WMC.pdf (дата обращения: 22.09.2022).

94. Cover Your Cough: Detection of Respiratory Events with Confidence Using a Smartwatch. – Текст : электронный // Proceedings of Machine Learning Research : [сайт]. – URL: https://proceedings.mlr.press/v91/nguyen18a/nguyen18a.pdf (дата обращения: 22.09.2022).

95. COVID-19 Artificial Intelligence Diagnosis using only Cough Recordings. – Текст : электронный // IEEE Digital Reality : [сайт]. – URL: https://digitalreality.ieee.org/images/files/pdf/COVID-AI-Diagnosis-using-Cough-Recordings.pdf (дата обращения: 22.09.2022).

96. Automatic Non-Invasive Cough Detection based on Accelerometer and Audio Signals / М. Пахар, И. Миранда, А. Дьякон, Т. Нислер. – Текст : непосредственный // Signal Process Systems. – 2022. – № 94. – С. 821-835.

97. Feature Extraction for the Differentiation of Dry and Wet Cough Sounds. – Текст : электронный // Carleton University Research Virtual Environment : [сайт]. – URL: https://curve.carleton.ca/system/files/etd/91d9768a-fdfc-4145-a9a4-a2255cdee360/etd_pdf/0ab37fbde60f27301616e43085ba99fb/chatrzarrin-featureextractionforthedifferentiationofdry.pdf (дата обращения: 22.09.2022).

98. Чикрин Д.Е., Пашин Д.М., Егорчев А.А., Фахрутдинов А.Ф. Использование имитационных манекенов для построения системы биомониторинга состояния здоровья человека // Одиннадцатая всероссийская научно-практическая конференция по имитационному моделированию и его применению в науке и промышленности имитационное моделирование. теория и практика (ИММОД-2023) сборник трудов. - Казань: АН РТ, 2023. - С. 547-554.

99. Yamnet. – Текст : электронный // TensorFlow Hub : [сайт]. – URL: https://tfhub.dev/google/lite-model/yamnet/tflite/1 (дата обращения: 22.09.2022).

100. Орландик, Л. The COUGHVID crowdsourcing dataset, a corpus for the study of large-scale cough analysis algorithms / Л. Орландик, Т. Тейджейро, Д. Атиенза. – Текст : непосредственный // Sci Data. – 2021. – № 8. – С. 156.

101. Detect and Segment Cough. – Текст : электронный // GitHub : [сайт]. – URL: https://github.com/bagustris/detect-segment-cough (дата обращения: 13.10.2022).

102. Вельч, П. The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms / П. Вельч. – Текст : непосредственный // EEE Transactions on Audio and Electroacoustics. – 1967. – № 15. – С. 70-73.

103. Бахвалов Н. С., Жидков Н. П., Кобельков Г. М. Численные методы. – 6-е изд. – М.: БИНОМ. Лаборатория знаний, 2008. – 636 с.

104. Adam: A Method for Stochastic Optimization. – Текст : электронный // arXiv Vanity : [сайт]. – URL: https://www.arxiv-vanity.com/papers/1412.6980/ (дата обращения: 13.10.2022).

105. Cover T. M., Thomas J. A. Elements of Information Theory. – 2nd ed. – Hoboken: John Wiley & Sons, 2006. – 748 p.

106. Вадутов, О. С. Математические основы обработки сигналов: конспект лекций / О. С. Вадутов; Томский политехнический университет. – Томск : Изд-во ТПУ, 2016. — Тема 6. Аналоговые фильтры. — С. 64–76.

107. Trudel-Fitzgerald, C. A healthy mix of emotions: underlying biological pathways linking emotions to physical health / C. Trudel-Fitzgerald, F. Qureshi, A. A. Appleton, L. D. Kubzansky // Current Opinion in Behavioral Sciences. – 2017. – Vol. 15. – P. 16–21.

108. Sentiment-Predictor-for-Stress-Detection-using-Voice. – Текст : электронный // GitHub : [сайт]. – URL: https://github.com/sidmulajkar/sentiment-predictor-for-stress-detection (дата обращения: 10.10.2022).

109. Audio Emotion | Part 1 - Explore data. – Текст : электронный // Kaggle : [сайт]. – URL: https://www.kaggle.com/code/ejlok1/audio-emotion-part-1-explore-data (дата обращения: 10.10.2022).

110. Surrey Audio-Visual Expressed Emotion (SAVEE) Database // The Centre for Vision, Speech and Signal Processing URL: http://kahlan.eps.surrey.ac.uk/savee/ (дата обращения: 29.11.2022).

111. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English // PLOS URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0196391 (дата обращения: 28.11.2022).

112. CREMA-D: Crowd-sourced Emotional Multimodal Actors Dataset // The National Center for Biotechnology Information URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4313618/ (дата обращения: 30.11.2022).

113. opensmile 2.4.2 // PyPI - Указатель пакетов Python'a URL: https://pypi.org/project/opensmile/ (дата обращения: 12.12.2022).

114. Аналитический обзор и классификация методов выделения признаков акустического сигнала в речевых системах // Известия Кабардино-Балкарского научного центра РАН URL: https://www.mathnet.ru/links/67c9001b85e58a9ffa0ce8598d4c31e1/izkab421.pdf (дата обращения: 07.07.2025).

115. Оптимизация комплексного подхода к коррекции голоса при эндокринопатиях (аналитический обзор) // ЭНЦ URL: https://www.probl-endojournals.ru/jour/article/view/12822?locale=ru_RU (дата обращения: 07.07.2025).

116. Кашель // Omnidoctor URL: https://omnidoctor.ru/upload/Kashel_monografiya.pdf (дата обращения: 07.07.2025).

117. Степанова Юлия Евгеньевна, Мохотаева Маргарита Владимировна, Корнеенков Алексей Александрович. АКУСТИЧЕСКИЕ ХАРАКТЕРИСТИКИ ГОЛОСА У ПРЕДСТАВИТЕЛЕЙ ГОЛОСОРЕЧЕВЫХ ПРОФЕССИЙ С ФУНКЦИОНАЛЬНОЙ ДИСФОНИЕЙ ПО ГИПОТОНУСНОМУ ТИПУ // РО. 2021. №4 (113). URL: https://cyberleninka.ru/article/n/akusticheskie-harakteristiki-golosa-u-predstaviteley-golosorechevyh-professiy-s-funktsionalnoy-disfoniey-po-gipotonusnomu-tipu (дата обращения: 07.09.2025).

118. TensorFlow Lite Example On-device Model Personalization // GitHub URL: https://github.com/tensorflow/examples/tree/master/lite/examples/model_personalization (дата обращения: 07.07.2025).

119. Гомоморфное шифрование // ИСП РАН URL: https://www.ispras.ru/proceedings/docs/2007/12/isp_12_2007_27.pdf?ysclid=mpwaly1h8133304358 (дата обращения: 15.04.2026).

120. Kullback S., Leibler R. A. On Information and Sufficiency // Annals of Mathematical Statistics. – 1951. – Vol. 22, No. 1. – P. 79–86.

121. Егорчев А.А., Чикрин Д.Е., Пашин Д.М., Фахрутдинов А.Ф., Кокунин П.А. Алгоритм детектирования дефектов мелкой моторики на основе информации от сенсоров смартфона // Известия ЮФУ. Технические науки. №4. - Таганрог: Южный Федеральный Университет, 2023. - С. 14-25.

122. Welltroy // Welltroy URL: https://welltory.com/ (дата обращения: 07.07.2025).

123. A review of Cardiio: an app to accurately measure heart rate at rest and after exercise using imaging photoplethysmography // BMJ URL: https://blogs.bmj.com/bjsm/2017/02/17/review-cardiio-app-accurately-measure-heart-rate-rest-exercise-using-imaging-photoplethysmography/ (дата обращения: 07.07.2025).

124. EHRV // EHRV URL: https://elitehrv.com (дата обращения: 07.07.2025).

125. OURA // OURA Ring URL: https://ouraring.com (дата обращения: 07.07.2025).

126. HealthKit // Developer Apple URL: https://developer.apple.com/documentation/healthkit (дата обращения: 07.07.2025).

127. Observer // In-Kotlin URL: https://in-kotlin.com/design-patterns/observer/ (дата обращения: 20.06.2023).

128. PyQt5 Signals, Slots & Events / Martin Fitzpatrick. – Текст : электронный // Python GUIs : [сайт]. – URL: https://www.pythonguis.com/tutorials/pyqt-signals-slots-events/ (дата обращения: 20.06.2023).

129. Facade // Habr URL: https://habr.com/ru/articles/688546/ (дата обращения: 20.06.2023).

130. Factory Method Python Design Patterns // GeeksForGeeks URL: https://www.geeksforgeeks.org/factory-method-python-design-patterns/ (дата обращения: 20.06.2023).

131. Moxy. – Текст : электронный // Moxy Community : [сайт]. – URL: https://moxy- community.github.io/Moxy/ (дата обращения: 20.06.2023).

132. Фильтр Калмана // Habr URL: https://habr.com/ru/articles/166693/ (дата обращения: 25.06.2023).

133. Г. Р. Сайпулаев, М. Р. Сайпулаев, И. В. Меркурьев, Б. И. Адамов, Р. Б. Гарсиа. Применение блока инерциальных датчиков для оценивания положения и управления движением экзоскелета нижних конечностей // Advanced Engineering Research (Rostov-on-Don). 2022. №3. URL: https://cyberleninka.ru/article/n/primenenie-bloka-inertsialnyh-datchikov-dlya-otsenivaniya-polozheniya-i-upravleniya-dvizheniem-ekzoskeleta-nizhnih-konechnostey (дата обращения: 03.04.2024).

134. Чикрин Д. Е., Егорчев А. А., Пашин Д. М., Сарамбаев Н. А. Построение системы определения частоты поверхностного дыхания по данным виброметрических сенсоров смартфона // Известия ЮФУ. Технические науки.. - 2023. - №4. - С. 35-42.

135. Hernandez J., McDuff D.J., Picard R.W. Biophone: Physiology monitoring from peripheral smartphone motions, 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2015,pp. 7180-7183.

136. Чикрин Д. Е., Егорчев А. А., Пашин Д. М., Сарамбаев Н. А. Построение системы определения частоты поверхностного дыхания по данным виброметрических сенсоров смартфона // Известия ЮФУ. Технические науки. - 2023. - №4. - С. 35-42.

137. Kanwa A., Sharma C., Deb S. Determination of SpO2 and Heart-rate using Smartphone Camera // Proceedings of the 2014 International Conference on Control, Instrumentation, Energy and Communication (CIEC). - 2014. - С. 40-49.

138. Wijaya N., Fauzi F., Helmy E., Nguyen P., Atmoko R. The Design of Heart Rate Detector and Body Temperature Measurement Device Using ATMega16 // Journal of Robotics and Control (JRC). - 2020. - №2. - С. 40-43.

139. Morsi I., Gawad Y. Measurement of Blood Pressure and Heart Beat Based on Sensors and Microcontrollers // Applied Mechanics and Materials. - 2013. - №249. - С. 193-201.

140. Optical blood pressure estimation with photoplethysmography and FFT-based neural networks // ResearchGate URL: https://www.researchgate.net/publication/305223445_Optical_blood_pressure_estimation_with_photoplethysmography_and_FFT-based_neural_networks (дата обращения: 02.10.2022).

141. Abay T., Kyriacou P. Photoplethysmography for blood volumes and oxygenation changes during intermittent vascular occlusions // Journal of Clinical Monitoring and Computing. - 2017. - №32. - С. 447-455.

142. Чикрин Д.Е., Егорчев А.А., Фахрутдинов А.Ф., Шарипов М.Р., Бурнашев Р.А. Методы получения информации биомедицинского мониторинга уровня оксигенации и артериального давления с использованием встроенных сенсоров смартфонной техники // Известия Саратовского университета. Новая серия. Серия: Математика. Механика. Информатика. 2024. Т. 24, вып. 3. С. 421–429.

143. Scully C., Lee J., Meyer J., Gorbach A., Granquist-Fraser D., Mendelson Y., Chon K. Physiological Parameter Monitoring from Optical Recordings with a Mobile Phone // IEEE Transactions on Biomedical Engineering. - 2012. - №59. - С. 303-306.

144. Setting–up of PPG Scaling Factors for SpO2% Evaluation by Smartphone // IEEE URL: https://ieeexplore.ieee.org/document/7533775 (дата обращения: 24.10.2022).

145. Cheng Q., Juen J., Schatz B. Using mobile phones to simulate pulse oximeters: gait analysis predicts oxygen saturation // Proceedings of the 5th ACM Conference on Bioinformatics, Computational Bioligy and Health Informatics. - 2014. - №1. - С. 331-340.

146. Фахрутдинов, А. Ф. Адаптация архитектуры мобильного комплекса мониторинга жизнедеятельности на основе федеративного

обучения и управления ресурсами / А.Ф. Фахрутдинов // Форум : электронный научно-практический журнал Волгоградского филиала АНО ВО МГЭУ. – 2026. – № 2 (36). – Статья принята к публикации.

ПРИЛОЖЕНИЕ А

Акты внедрения результатов диссертационной работы

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
(КФУ)

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Кремлевская ул., д. 35, корпус 2, Казань, 420008
тел. (843) 233-70-37, факс (843) 233-70-37
эл. почта: vmk.dep@kpfu.ru

12.10.2026 № 1.1.2.09.1.01-25/365/2026

На №_____от _____

АКТ О ВНЕДРЕНИИ
об использовании результатов диссертационного исследования в НИЦ

Настоящий акт составлен о том, что результаты диссертационной работы Фахрутдинова Аделя Фердинандовича «Программный комплекс анализа показателей жизнедеятельности человека с использованием самоадаптирующейся микросервисной архитектуры», представленной на соискание ученой степени кандидата технических наук по специальности 2.3.5 – «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей», использованы в научно-исследовательском центре Центр превосходства Специальная робототехника и искусственный интеллект ФГАОУ ВО КФУ (далее — НИЦ) при выполнении ряда научно-исследовательских и опытно-конструкторских работ по направлениям построения самоадаптирующихся программных комплексов, контроля устойчивости моделей машинного обучения и распределенного обучения.

Внедрение затронуло построение архитектуры программного комплекса с микросервисным взаимодействием компонентов и процессы поддержания актуальности моделей машинного обучения в условиях изменения распределения входных данных.

В основу процесса внедрения положены разработанные автором:

архитектурная модель самоадаптирующегося программного комплекса, основанная на микросервисном взаимодействии компонентов, обнаружении дрейфа данных и федеративном обновлении моделей машинного обучения;

методика обнаружения дрейфа данных и верификации качества моделей машинного обучения, обеспечивающая запуск процедуры федеративного дообучения без передачи исходных данных.

Применение разработанной методики подтвердило эффективность управления дрейфом данных: расхождения обнаружены в 35% случаев изменения распределения входных данных, что позволило своевременно фиксировать устаревание моделей машинного обучения; после проведения федеративного дообучения мера F1 глобальной модели машинного обучения, обрабатывающей акустические сигналы, повысилась с 0,81 до 0,88 за счет обновления моделей по данным отдельных клиентских устройств без их централизованной передачи; архитектура программного комплекса обеспечила масштабирование и равномерное распределение вычислительной нагрузки при подключении дополнительных клиентских устройств без остановки процессов обработки данных, что повысило устойчивость сервисов к пиковым нагрузкам и сократило время развертывания обновлений. Обеспечена прослеживаемость обновлений моделей машинного обучения, что сократило время подготовки отчетности по достоверности результатов в рамках НИР и НИОКР.

Таким образом, внедрение результатов диссертационной работы позволило улучшить качество проводимых исследований и поддержать актуальность моделей машинного обучения в составе выполняемых НИР и НИОКР.

Факт использования научных результатов подтверждается настоящим актом.

Директор НИЦ Кокунин П.А.

«12» 10 2026 г. М.П.

ЕГОРЧЕВ АНТОН АЛЕКСАНДРОВИЧ
+7 917 237 30 47
CONTACT@НПО-ПРОМРОБОТ.РФ
HTTPS://НПО-ПРОМРОБОТ.РФ
ООО «НПО ПРОМЫШЛЕННАЯ РОБОТОТЕХНИКА»
ОГРН: 1171690116773
ИНН: 1655394887 /КПП: 166001001

НПО ПРОМЫШЛЕННАЯ РОБОТОТЕХНИКА

АКТ О ВНЕДРЕНИИ

об использовании результатов в научно-производственной деятельности

Настоящий акт составлен о том, что результаты диссертационной работы Фахрутдинова Аделя Фердинандовича «Программный комплекс анализа показателей жизнедеятельности человека с использованием самоадаптирующейся микросервисной архитектуры», представленной на соискание ученой степени кандидата технических наук по специальности 2.3.5 – «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей», внедрены в научно-производственную деятельность ООО «НПО Промышленная робототехника» при разработке продукта для анализа показателей жизнедеятельности человека.

Результаты диссертационной работы реализованы в виде программного комплекса, включающего мобильное приложение и серверный узел. Внедрение затронуло процессы многопараметрической обработки сенсорных данных мобильных устройств и планирования автономной работы устройств.

В основу процесса внедрения положены разработанные автором:

алгоритм многопараметрической обработки акустических, кинематических и визуальных сенсорных данных мобильного устройства, обеспечивающий формирование информационного статуса 17 показателей жизнедеятельности человека;

алгоритм адаптивного управления режимами работы сенсорных и вычислительных компонентов мобильного приложения, обеспечивающий рациональное использование ресурсов устройства.

Применение разработанных решений позволило добиться значительных улучшений при разработке продукта для анализа показателей жизнедеятельности человека: применение алгоритма многопараметрической обработки сенсорных данных обеспечило формирование информационного статуса по 17 показателям жизнедеятельности с гибким выбором необходимых для использования показателей в зависимости от задач применения, что сократило время на ручную обработку и интерпретацию результатов; применение алгоритма адаптивного управления режимами работы сенсорных и вычислительных компонентов увеличило время автономной работы мобильного устройства в среднем на 14% при сохранении требуемой точности оценок, что снизило частоту подзарядки устройства и повысило продолжительность непрерывного мониторинга; применение программного комплекса в целом обеспечило устойчивую работу серверного узла при пиковых нагрузках и возможность масштабирования обслуживания подключающихся пользователей без остановки процессов обработки данных.

Таким образом, внедрение результатов диссертационной работы позволило улучшить качество разрабатываемого продукта для анализа показателей жизнедеятельности человека.

Внедрение подтверждается настоящим актом.

Генеральный директор / Егорчев А.А./

«09» 02 2026 г. М. П.



РЕКВИЗИТЫ:
Р/С 40702810710000238281
В БАНКЕ АО «ТИНЬКОФФ БАНК»,
БИК 044525974
К/С 30101810145250000974

ЮРИДИЧЕСКИЙ АДРЕС:
ИНДЕКС: 420061
РЕСПУБЛИКА ТАТАРСТАН Г. КАЗАНЬ
УЛ. НОВАТОРОВ Д. 2В ПОМЕЩЕНИЕ ОФ35

ПРИЛОЖЕНИЕ Б

СВИДЕТЕЛЬСТВО РЕГИСТРАЦИИ ПРОГРАММЫ



РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2023666549

Программа клиентского модуля комплекса централизованного дистанционного мониторинга основных показателей здоровья сотрудников

Правообладатели: ***федеральное государственное автономное образовательное учреждение высшего образования «Казанский (Приволжский) федеральный университет» (ФГАОУ ВО КФУ) (RU), Акционерное общество "Радиокомпания "ВЕКТОР" (АО "РАДИОКОМПАНИЯ "ВЕКТОР") (RU)***

Авторы: ***Чикрин Дмитрий Евгеньевич (RU), Егорчев Антон Александрович (RU), Фахрутдинов Адель Фердинандович (RU), Мусин Фоат Минегереевич (RU), Шарипов Марсель Ривкатович (RU), Сарамбаев Никита Андреевич (RU)***

Заявка № **2023665812**

Дата поступления **27 июля 2023 г.**

Дата государственной регистрации в Реестре программ для ЭВМ ***02 августа 2023 г.***

Руководитель Федеральной службы по интеллектуальной собственности

Ю.С. Зубов

ПРИЛОЖЕНИЕ В
ПРОТОКОЛЫ ИСПЫТАНИЙ ПРОГРАММНОГО КОМПЛЕКСА

Приложение содержит индивидуальные протоколы испытаний программного комплекса неинвазивного мониторинга показателей жизнедеятельности человека. Таблица В.1 определяет соответствие испытуемых и тестовых устройств. Таблицы В.2 и В.3 содержат индивидуальные значения меры F1 по моделям машинного обучения до и после обновления в результате федеративного обучения для каждого испытуемого. Значения метрик в таблицах протокола приведены в процентах.

Таблица В.1 – Соответствие испытуемых и тестовых устройств

№ испытуемого	Возрастная группа	Устройство (модель)	Класс производительности
1	25–40	Honor X7c	Средний
2	25–40	Xiaomi Redmi 13	Начальный
3	25–40	Samsung Galaxy A26	Средний
4	25–40	Realme 14T	Средний
5	25–40	Samsung Galaxy A06	Начальный
6	25–40	Huawei nova Y72S	Средний
7	25–40	POCO M7 Pro	Средний
8	25–40	Samsung Galaxy A36	Средний
9	25–40	Tecno SPARK 20 Pro	Средний
10	25–40	Samsung Galaxy A26	Средний
11	25–40	Honor X7c	Средний
12	25–40	Xiaomi Redmi 13	Начальный
13	25–40	Realme 14T	Средний
14	25–40	POCO M7 Pro	Средний
15	25–40	Huawei nova Y72S	Средний
16	60–76	Huawei nova 8i	Начальный
17	60–76	Samsung A8 (2018)	Средний
18	60–76	Xiaomi Mi 9 Lite	Средний
19	60–76	Redmi Note 9S	Средний
20	60–76	POCO X3 Pro	Средне-высокий
21	60–76	Samsung SM-A515F	Средний
22	60–76	Samsung A8 (2018)	Средний
23	60–76	Xiaomi Mi 9 Lite	Средний
24	60–76	Huawei nova 8i	Начальный
25	60–76	Redmi Note 9S	Средний
26	60–76	Samsung SM-A515F	Средний
27	60–76	POCO X3 Pro	Средне-высокий
28	60–76	Huawei nova 8i	Начальный
29	60–76	Samsung A8 (2018)	Средний
30	60–76	Xiaomi Mi 9 Lite	Средний

Таблица В.2 – Индивидуальные результаты испытаний моделей определения кашля и чихания по испытуемым

№ испытуемого	Возрастная группа	Кашель: F1 до обновления	Кашель: F1 после обновления	Чихание: F1 до обновления	Чихание: F1 после обновления
1	25–40	92,5	93,0	89,5	90,5
2	25–40	91,4	91,9	90,5	92,0
3	25–40	88,7	89,3	85,7	86,1
4	25–40	90,4	90,4	88,1	89,1
5	25–40	91,9	92,0	83,2	84,1
6	25–40	93,1	93,1	93,0	93,0
7	25–40	88,0	88,0	87,7	87,7
8	25–40	88,8	89,1	91,5	91,5
9	25–40	87,8	87,8	91,9	92,7
10	25–40	92,3	92,3	89,2	89,7
11	25–40	92,2	93,6	85,2	86,0
12	25–40	87,3	87,9	84,3	84,3
13	25–40	91,2	91,7	89,1	90,6
14	25–40	90,5	90,5	83,9	85,5
15	25–40	97,4	97,4	87,2	87,7
16	60–76	83,0	86,1	76,4	79,6
17	60–76	83,2	88,9	82,1	88,4
18	60–76	87,5	92,2	83,1	89,0
19	60–76	80,8	84,7	86,4	86,9
20	60–76	89,1	91,6	80,8	84,4
21	60–76	83,8	84,8	73,6	84,5
22	60–76	84,2	84,5	78,3	85,6
23	60–76	82,3	83,3	82,3	90,8
24	60–76	77,9	81,3	88,4	93,9
25	60–76	79,8	80,8	77,7	82,5
26	60–76	83,8	84,8	70,9	74,2
27	60–76	82,8	88,1	80,0	82,6
28	60–76	82,2	83,2	75,5	82,2
29	60–76	88,1	91,5	78,6	82,7
30	60–76	81,0	82,7	81,4	87,7

Таблица В.3 – Индивидуальные результаты испытаний моделей определения эмоций по испытуемым

№ испытуемого	Возрастная группа	Негативная эмоция: F1 до	Негативная эмоция: F1 после	Позитивная эмоция: F1 до	Позитивная эмоция: F1 после	Нейтральная эмоция: F1 до	Нейтральная эмоция: F1 после
1	25–40	90,6	90,6	88,7	90,8	80,9	88,0
2	25–40	98,8	98,8	85,0	85,6	85,2	92,9
3	25–40	97,7	97,9	90,3	91,2	75,1	79,1
4	25–40	95,1	95,2	90,4	92,0	80,7	86,7
5	25–40	94,9	94,9	83,7	86,0	78,8	83,7
6	25–40	96,0	96,0	87,5	89,8	81,1	86,9
7	25–40	99,0	99,0	88,8	90,2	77,0	84,8
8	25–40	90,6	90,9	89,4	91,2	81,5	85,8
9	25–40	95,5	95,5	89,9	92,4	82,5	87,2
10	25–40	93,6	93,6	87,7	87,7	80,5	83,9
11	25–40	99,5	99,5	92,4	93,1	79,3	82,9
12	25–40	94,2	94,5	89,6	90,4	76,1	81,0
13	25–40	97,4	97,4	88,0	88,8	85,9	93,3
14	25–40	96,5	97,1	87,6	87,6	71,9	82,5
15	25–40	96,1	96,1	91,5	94,7	73,0	88,3
16	60–76	73,5	78,8	87,9	88,6	79,3	83,7
17	60–76	83,4	84,6	82,4	84,0	93,2	99,6
18	60–76	71,8	78,8	88,9	90,1	84,0	89,7
19	60–76	66,9	77,9	88,9	88,9	86,5	91,7
20	60–76	61,3	64,9	88,3	88,9	82,8	86,1
21	60–76	68,9	79,6	91,3	94,9	82,5	83,8
22	60–76	68,1	76,2	83,1	85,5	89,2	94,8
23	60–76	79,7	86,3	85,0	87,1	90,4	95,0
24	60–76	73,3	81,1	88,9	92,3	81,3	84,2
25	60–76	88,9	90,0	88,3	90,6	83,9	88,1
26	60–76	58,2	65,6	81,8	83,7	84,9	88,0
27	60–76	83,6	88,2	83,0	84,9	80,8	85,3
28	60–76	56,5	60,7	93,5	94,5	87,7	90,8
29	60–76	63,1	66,7	89,1	90,4	88,7	95,0
30	60–76	75,3	81,6	82,9	85,9	82,8	86,7