

ISSN 2305-9052 (Print)  
ISSN 2410-7034 (Online)

# ВЕСТНИК



ЮЖНО-УРАЛЬСКОГО  
ГОСУДАРСТВЕННОГО  
УНИВЕРСИТЕТА

# BULLETIN

OF THE SOUTH URAL  
STATE UNIVERSITY

**СЕРИЯ**

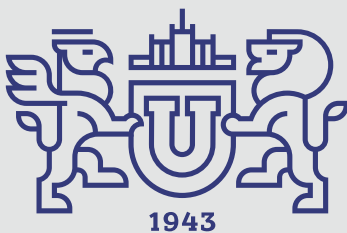
**ВЫЧИСЛИТЕЛЬНАЯ  
МАТЕМАТИКА  
И ИНФОРМАТИКА**

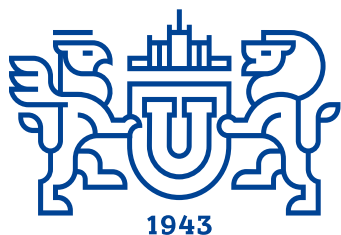
**2021, том 10, № 4**

**SERIES**

**COMPUTATIONAL  
MATHEMATICS  
AND SOFTWARE ENGINEERING**

**2021, volume 10, no. 4**





# ВЕСТНИК

ЮЖНО-УРАЛЬСКОГО  
ГОСУДАРСТВЕННОГО  
УНИВЕРСИТЕТА

2021  
Т. 10, № 4

ISSN 2305-9052

СЕРИЯ

## «ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА И ИНФОРМАТИКА»

Решением ВАК включен в Перечень научных изданий,  
в которых должны быть опубликованы результаты диссертаций  
на соискание ученых степеней кандидата и доктора наук

Учредитель — Федеральное государственное автономное образовательное учреждение  
высшего образования «Южно-Уральский государственный университет  
(национальный исследовательский университет)»

Тематика журнала:

- Вычислительная математика и численные методы
- Математическое программирование
- Распознавание образов
- Вычислительные методы линейной алгебры
- Решение обратных и некорректно поставленных задач
- Доказательные вычисления
- Численное решение дифференциальных и интегральных уравнений
- Исследование операций
- Теория игр
- Теория аппроксимации
- Информатика
- Искусственный интеллект и машинное обучение
- Системное программирование
- Перспективные многопроцессорные архитектуры
- Облачные вычисления
- Технология программирования
- Машинная графика
- Интернет-технологии
- Системы электронного обучения
- Технологии обработки баз данных и знаний
- Интеллектуальный анализ данных

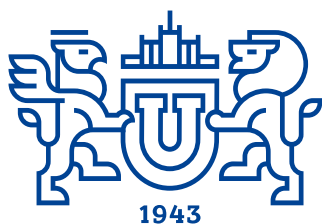
### Редакционная коллегия

**Л.Б. Соколинский**, д.ф.-м.н., проф., *гл. редактор*  
**В.П. Танана**, д.ф.-м.н., проф., *зам. гл. редактора*  
**М.Л. Цымблер**, д.ф.-м.н., доц., *отв. секретарь*  
**Г.И. Радченко**, к.ф.-м.н., доц. (Австрия)  
**Я.А. Краева**, *техн. секретарь*

### Редакционный совет

**С.М. Абдуллаев**, д.г.н., профессор  
**А. Андреев**, PhD, профессор (Германия)  
**В.И. Бердышев**, д.ф.-м.н., акад. РАН, *председатель*  
**В.В. Воеводин**, д.ф.-м.н., чл.-кор. РАН

**Дж. Донгарра**, PhD, профессор (США)  
**С.В. Зыкин**, д.т.н., профессор  
**И.М. Куликов**, д.ф.-м.н.  
**Д. Маллманн**, PhD, профессор (Германия)  
**А.В. Паноков**, д.ф.-м.н., профессор  
**Р. Продан**, PhD, профессор (Австрия)  
**В.И. Ухоботов**, д.ф.-м.н., профессор  
**В.Н. Ушаков**, д.ф.-м.н., чл.-кор. РАН  
**М.Ю. Хачай**, д.ф.-м.н., профессор  
**А. Черных**, PhD, профессор (Мексика)  
**П. Шумяцкий**, PhD, профессор (Бразилия)



# BULLETIN

OF THE SOUTH URAL 2021  
STATE UNIVERSITY vol. 10, no. 4

SERIES

“COMPUTATIONAL  
MATHEMATICS AND SOFTWARE  
ENGINEERING”

ISSN 2305-9052

---

Vestnik Yuzhno-Ural'skogo Gosudarstvennogo Universiteta.  
Seriya “Vychislitel'naya Matematika i Informatika”

---

## South Ural State University

The scope of the journal:

- Numerical analysis and methods
- Mathematical optimization
- Pattern recognition
- Numerical methods of linear algebra
- Reverse and ill-posed problems solution
- Computer-assisted proofs
- Numerical solutions of differential and integral equations
- Operations research
- Game theory
- Approximation theory
- Computer science
- Artificial intelligence and machine learning
- System software
- Advanced multiprocessor architectures
- Cloud computing
- Software engineering
- Computer graphics
- Internet technologies
- E-learning
- Database processing
- Data mining

### Editorial Board

**L.B. Sokolinsky**, South Ural State University (Chelyabinsk, Russia)  
**V.P. Tanana**, South Ural State University (Chelyabinsk, Russia)  
**M.L. Zymbler**, South Ural State University (Chelyabinsk, Russia)  
**G.I. Radchenko**, Silicon Austria Labs (Graz, Austria)  
**Ya.A. Kraeva**, South Ural State University (Chelyabinsk, Russia)

### Editorial Council

**S.M. Abdullaev**, South Ural State University (Chelyabinsk, Russia)  
**A. Andrzejak**, Heidelberg University (Germany)  
**V.I. Berdyshev**, Institute of Mathematics and Mechanics, Ural Branch of the RAS (Yekaterinburg, Russia)  
**J. Dongarra**, University of Tennessee (USA)  
**M.Yu. Khachay**, Institute of Mathematics and Mechanics, Ural Branch of the RAS (Yekaterinburg, Russia)  
**I.M. Kulikov**, Institute of Computational Mathematics and Mathematical Geophysics, Siberian Branch of RAS (Novosibirsk, Russia)  
**D. Mallmann**, Julich Supercomputing Centre (Germany)  
**A.V. Panyukov**, South Ural State University (Chelyabinsk, Russia)  
**R. Prodan**, Alpen-Adria-Universität Klagenfurt (Austria)  
**P. Shumyatsky**, University of Brasilia (Brazil)  
**A. Tchernykh**, CICESE Research Center (Mexico)  
**V.I. Ukhobotov**, Chelyabinsk State University (Chelyabinsk, Russia)  
**V.N. Ushakov**, Institute of Mathematics and Mechanics, Ural Branch of the RAS (Yekaterinburg, Russia)  
**V.V. Voevodin**, Lomonosov Moscow State University (Moscow, Russia)  
**S.V. Zykin**, Sobolev Institute of Mathematics, Siberian Branch of the RAS (Omsk, Russia)

## Содержание

ОБ ОДНОМ МЕТОДЕ ВОССТАНОВЛЕНИЯ ПРОПУЩЕННЫХ ЗНАЧЕНИЙ ПОТОКОВОГО ВРЕМЕННОГО РЯДА В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ М.Л. Цымблер, В.А. Полонский, А.А. Юртин .....	5
АЛГОРИТМ СОЕДИНЕНИЯ ЦИКЛОВ ДЛЯ МЕТРИЧЕСКОЙ ЗАДАЧИ КОММИВОЯЖЕРА НА МАКСИМУМ А.В. Панюков, Ю.Ф. Леонова .....	26
ОРГАНИЗАЦИЯ ОБМЕНА ДАННЫМИ В РАМКАХ ПЛАТФОРМЫ МОБИЛЬНОЙ МЕДИЦИНЫ И.А. Волков, Г.И. Радченко, А.Н. Черных .....	37
СИСТЕМА УЧЕТА ПОСЕЩАЕМОСТИ СТУДЕНТОВ НА ОСНОВЕ РАСПОЗНАВАНИЯ ЛИЦ Е.В. Иванова, А.Ю. Струева .....	60

# Contents

ON ONE METHOD OF IMPUTATION MISSING VALUES OF A STREAMING TIME SERIES IN REAL TIME M.L. Zymbler, V.A. Polonsky, A.A. Yurtin .....	5
CYCLES MERGING ALGORITHM FOR METRIC MAXIMUM TRAVELING SALESMAN PROBLEM A.V. Panyukov, Yu.F. Leonova .....	26
DATA SHARING IN MOBILE HEALTH PLATFORM I.A. Volkov, G.I. Radchenko, A.N. Tchernykh .....	37
THE STUDENT ATTENDANCE CONTROL SYSTEM BASED ON FACE RECOGNITION E.V. Ivanova, A.Yu. Srtueva .....	60



This issue is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

## ОБ ОДНОМ МЕТОДЕ ВОССТАНОВЛЕНИЯ ПРОПУЩЕННЫХ ЗНАЧЕНИЙ ПОТОКОВОГО ВРЕМЕННОГО РЯДА В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ

© 2021 М.Л. Цымблер, В.А. Полонский, А.А. Юртин

*Южно-Уральский государственный университет*

*(454080 Челябинск, пр. им. В.И. Ленина, д. 76)*

*E-mail: mzym@susu.ru, s.polonski@mail.ru, lideor@yandex.ru*

Поступила в редакцию: 03.09.2021

Проблема восстановления пропущенных значений потокового временного ряда в режиме реального времени возникает в широком спектре практических приложений цифровой индустрии и интернета вещей. В статье предложен новый метод восстановления на основе совместного применения технологий интеллектуального анализа временных рядов и искусственных нейронных сетей. Метод предполагает три этапа восстановления: предварительная обработка данных, распознавание и реконструкция. Предварительная обработка предполагает однократную предварительную подготовку обучающих выборок данных. Распознавание и реконструкция реализуются с помощью нейронных сетей, обучаемых на указанных выборках. Предварительной обработке подвергается заранее сохраненный фрагмент потокового временного ряда без пропусков, в котором выполняется поиск набора типичных подпоследовательностей (сниппетов). Распознавание реализуется с помощью сверточной нейронной сети, на вход которой подается вектор из элементов временного ряда, предшествующих пропуску. Распознаватель выдает сниппет, на который более всего похожа входная подпоследовательность. Реконструкция реализуется с помощью рекуррентной нейронной сети, на вход которой подается конкатенация вывода распознавателя и вектора элементов ряда, предшествующих пропуску. Реконструктор выдает восстановленное значение. Представлены результаты экспериментов, показывающих высокую точность восстановления и преимущество предложенного метода перед аналогами.

*Ключевые слова: временной ряд, восстановление пропущенных значений, режим реального времени, сверточная нейронная сеть, рекуррентная нейронная сеть, типичные подпоследовательности.*

### ОБРАЗЕЦ ЦИТИРОВАНИЯ

Цымблер М.Л., Полонский В.А., Юртин А.А. Об одном методе восстановления пропущенных значений потокового временного ряда в режиме реального времени // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 4. С. 5–25. DOI: 10.14529/cmse210401.

### Введение

Временной ряд представляет собой совокупность вещественных значений, взятых в хронологическом порядке. В настоящее время широкий спектр приложений цифровой индустрии и интернета вещей связан с обработкой временных рядов в режиме реального времени: мониторинг технического состояния сложных машин и механизмов [1, 2], интеллектуальное управление системами жизнеобеспечения [3, 4], мониторинг показателей функциональной диагностики организма человека [5], моделирование климата [6], финансовое прогнозирование [7] и др. В силу различных обстоятельств временной ряд может содержать пропущенные значения, например, ввиду сбоя соответствующего датчика или соединительной сети. Однако для указанных выше приложений пропуски недопустимы и должны быть незамедлительно заменены на правдоподобные синтетические значения. В соответствии с этим одной из актуальных задач интеллектуальной обработки данных временных рядов

является создание методов и алгоритмов, которые обеспечивают восстановление пропущенных значений временного ряда в режиме реального времени и являются эффективными в смысле точности и быстродействия восстановления.

В рамках данного исследования нами разработан метод SANNI (Snippet and Neural Network based Imputation), обеспечивающий восстановление пропущенных значений временного ряда в режиме реального времени на основе совместного применения концепции снippetов и технологий искусственных нейронных сетей. Термин снippet (snippet) был предложен Кеогом и др. в работе [8] как уточнение понятия типичной подпоследовательности временного ряда. Снippet неформально может быть определен как подпоследовательность ряда, имеющая заданную длину, на которую похожи многие другие подпоследовательности данного ряда. Набор снippetов имеет существенно меньшую мощность, чем множество подпоследовательностей ряда, имеющих заданную длину, и потому может использоваться для аннотирования исходного временного ряда.

Статья организована следующим образом. Раздел 1 содержит краткий обзор работ по тематике исследования. В разделе 2 представлен метод SANNI для восстановления пропущенных значений временного ряда в режиме реального времени. В разделе 3 приведены результаты вычислительных экспериментов по оценке эффективности предложенного метода. Заключение резюмирует полученные результаты и описывает направления будущих исследований.

## 1. Обзор связанных работ

Метод восстановления Hot Deck [9] заменяет отсутствующий элемент временного ряда значением наиболее близкого к нему по времени присутствующего элемента данного ряда. В случае двух одинаково близких по времени элементов берется элемент с меньшим значением индекса. В общем случае данный метод дает низкую точность восстановления.

Восстановление на основе среднего или медианы [10] предполагает замену пропущенных значений на среднее или наиболее часто встречающееся среди всех известных значений во временном ряде соответственно. Данные техники сравнительно просты для реализации, однако, как правило, дают низкую точность восстановления, поскольку для замены всех отсутствующих значений используется одна и та же константа.

Техники восстановления на основе интерполяции (например, линейная и сплайновая интерполяции) [11] конструируют отсутствующее значение на основе значений, непосредственно предшествующих отсутствующему и идущих за ним. Если количество последовательно отсутствующих значений достаточно большое, указанные техники дают плохие результаты восстановления: например, отсутствующий период синусоиды будет заменен прямой.

Регрессионные методы [12] вычисляют отсутствующее значение временного ряда на основе значений в других временных рядах. Например, в работе [13] описано восстановление показаний метеостанции как среднее арифметическое соответствующих показаний других метеостанций, расположенных поблизости. Линейная регрессия [12] основывается на предположении линейной зависимости между значениями временных рядов, что на практике зачастую не имеет места, в силу чего данный метод не подходит для временных рядов с нелинейной зависимостью между ними.

Модель ARIMA (AutoRegressive Integrated Moving Average, интегрированная модель авторегрессии — скользящего среднего) [14] представляет собой обобщение авторегрессионной

модели, используемой в прогнозировании значений временного ряда. В данной модели предполагается линейная зависимость будущего значения временного ряда от значений данного ряда в прошлом. Недостатком этого алгоритма является большое количество параметров  $p, q, d$ , нахождение которых представляет сложный и трудоемкий процесс. Модель имеет относительно большое количество параметров  $(p, q, d)$ , подбор которых представляет собой кропотливую ручную работу, известную как методология Бокса—Дженкинса [14].

Батиста (Batista) и др. в работе [10] предложили алгоритм восстановления значений временного ряда  $k$ NNI ( $k$ -Nearest Neighbor Imputation), использующий принцип ближайших соседей. В работе исследовалось множество многомерных временных рядов, в котором некоторый ряд  $T$  имеет отсутствующие значения в измерении  $A$ . Предложенный подход предполагает нахождение  $k$  временных рядов, в которых измерения, отличные от  $A$ , являются похожими на соответствующие измерения в ряде  $T$ .

Троянская (Trojanskaya) и др. в работе [15] модифицировали алгоритм  $k$ NNI использованием взвешенных ближайших соседей, где вес соседа прямо пропорционален значению меры схожести с образцом поиска. В качестве меры схожести используется евклидова метрика.

Хаяти (Khayati) и др. предложили алгоритм REBOM (Recovery Blocks of Missing values) [16] который используется для восстановления блоков пропущенных значений в нерегулярном временном ряде (ряде, не имеющем выраженных повторяющихся трендов). Алгоритм выполняет построение матрицы, в которой хранятся значения неполного ряда и  $k$  рядов, имеющих наибольшее значение коэффициента корреляции Пирсона с данным рядом. Отсутствующие значения инициализируются с помощью интерполяции. Затем матрица итеративно подвергается сингулярному разложению и наименее значимые сингулярные числа матрицы отбрасываются. Ввиду квадратичной временной сложности алгоритм REBOM плохо масштабируется в случае временных рядов большой длины. Далее авторы в работе [17] представили улучшение REBOM на базе центроидного разложения, которое является приближением сингулярного разложения и обеспечивает линейную пространственную сложность восстановления. Улучшенный алгоритм тоже предполагает линейную корреляцию ряда, имеющего пропуски значений, и рядов-потенциальных соседей, иначе демонстрирует низкую точность восстановления.

Алгоритм MUSCLES предложен Йи (Yi) и др. в работе [12] и выполняет восстановление пропусков в режиме реального времени на основе множественной авторегрессионной модели. Параметры указанной модели обновляются с помощью рекурсивного метода наименьших квадратов (Recursive Least Squares, RLS), чтобы минимизировать ошибку восстановления пропущенного значения. MUSCLES основывается не только на предыдущих значениях неполного ряда, но и на предыдущих и текущих значениях других временных рядов, имеющих линейную корреляцию с данным рядом. Алгоритм плохо масштабируется для большого количества временных рядов, поскольку имеет квадратичные пространственную и временную сложности.

Пападимитроу (Papadimitriou) и др. предложили алгоритм SPIRIT [18]. SPIRIT использует метод главных компонент (Principal Component Analysis, PCA) [19] для сокращения набора совместно развивающихся и коррелированных потоков с небольшим количеством скрытых переменных, которые суммируют основные тенденции во всей коллекции потоков. Каждой скрытой переменной соответствует одна модель авторегрессии для прошлых значений, которые постепенно обновляются по мере поступления новых данных. Если значение



отсутствует, модель авторегрессии используется для прогнозирования текущего значения каждой переменной, из которой выводится оценка пропущенного значения. Восстановленное значение вместе с пропущенными значениями используется для обновления моделей прогнозирования.

Алгоритм ТКСМ (Top- $k$  Case Matching) [20], предложенный Велленцоном (Wellenzohn) и др., использует набор *связанных временных рядов* (*reference time series*), имеющих в силу особенностей данной предметной области семантическую связь с неполным временным рядом. Алгоритм ТКСМ использует для восстановления отсутствующих значений сходные исторические ситуации в связанных временных рядах. Ситуация описывается *опорной точкой* (*anchor point*) шаблона, который состоит из  $\ell$  последовательных значений временного ряда, связанного с данным. Пропущенное значение во временном ряде вычисляется из исторических значений этого же ряда в опорных точках  $k$  наиболее похожих *шаблонов*. Алгоритм ТКСМ обеспечивает высокую точность восстановления, если шаблон длины  $\ell$  к моменту времени  $t_n$  повторяется как минимум  $k$  раз в связанных временных рядах и соответствующие значения данного временного ряда в опорных точках примерно равны друг другу. Алгоритм способен обрабатывать временные ряды, между которыми отсутствует линейная корреляция (например, имеющие фазовые сдвиги) и показывает в среднем более высокую точность восстановления в случае блоков отсутствующих значений, чем REBOM и MUSCLES. Высокая точность восстановления, однако, обеспечивается за счет существенных временных затрат на стадии поиска шаблонов.

Хсу (Hsu) и др. в работе [21] предложили алгоритм восстановления отсутствующих значений, основанный на использовании принципа  $k$  ближайших соседей и меры схожести DTW [22]. Алгоритм используется для восстановления отсутствующих значений во временном ряде, который представляет собой значения уровня экспрессии генов, полученные с помощью ДНК-микрочипов в серии экспериментов.

Цао (Cao) и др. разработали модель двунаправленной рекуррентной нейронной сети BRITS [23] для восстановления пропусков в многомерных временных рядах. BRITS рассматривает отсутствующие значения как переменные двунаправленного RNN графа, что делает восстановление более точным из-за получения отложенных градиентов в обоих направлениях.

Гуо (Guo) и др. в работе [24] предложили метод восстановления многомерного временного ряда с помощью генеративно-сопоставительной нейронной сети (Generative Adversarial Network, GAN). Подход использует операцию многоканальной свертки в GAN при моделировании распределения многомерных рядов, что позволяет увеличить точность восстановления в сравнении с аналогами.

## 2. Метод восстановления пропусков временного ряда в режиме реального времени

*Потоковый временной ряд* (*streaming time series*) представляет собой набор вещественных значений, поступающих последовательно одно за другим в режиме реального времени:  $T = (\dots, t_{n-2}, t_{n-1}, t_n)$ ,  $t_i \in \mathbb{R}$ . Мы полагаем, что  $n$ -элементное окно ряда  $T$  достаточно велико (сотни тысяч элементов) и его данные могут быть размещены в оперативной памяти. Рассматриваемая в данном исследовании задача предполагает, что значение элемента ряда  $t_n$  в текущий момент времени пропущено (что обозначается нами как  $t_n = \text{NULL}$ ) и вместо него в режиме реального времени должно быть вычислено значение  $\tilde{t}_n$ .

Ниже приводится детальное описание архитектуры и основных модулей предложенного метода.

## 2.1. Архитектура метода

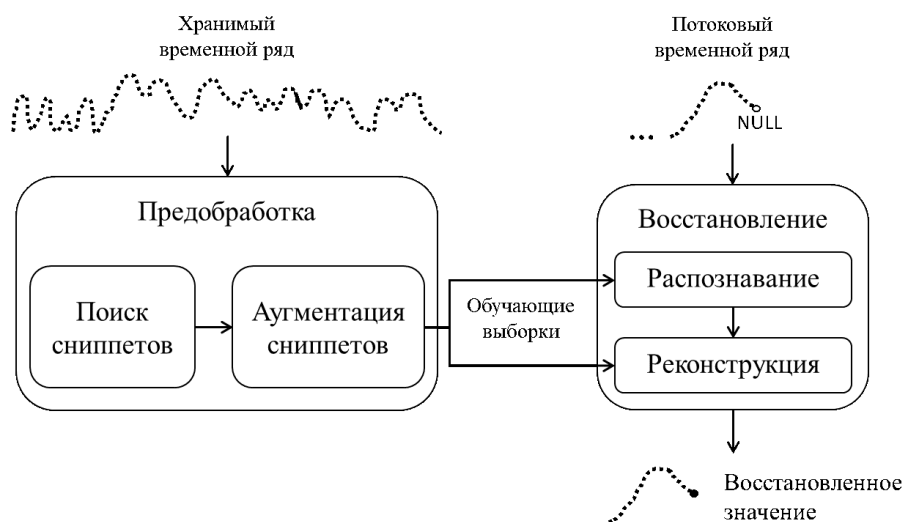


Рис. 1. Архитектура метода SANNI

Метод SANNI предполагает выполнение следующих этапов для восстановления пропущенных значений временного ряда (см. рис. 1): предварительная обработка данных, распознавание и реконструкция. *Предварительная обработка* предполагает однократную предварительную подготовку обучающих выборок данных, обучение на указанных выборках и настройку параметров нейронных сетей, которые реализуют этапы распознавания и реконструкции, выполняющиеся в режиме реального времени. Предварительная обработка данных выполняется регулярно для обновления обучающих выборок и тонкой настройки параметров нейронных сетей.

Для предварительной обработки выбирается заранее сохраненная подпоследовательность исходного потокового временного ряда, которая не содержит пропусков и может быть размещена в оперативной памяти. Мы полагаем, что сохраненная подпоследовательность является достаточно представительной для данной предметной области и охватывает подавляющее большинство потенциально возможных ситуаций при снятии показаний в режиме реального времени. Для удобства дальнейшего изложения и без существенного ограничения общности мы полагаем, что предварительной обработке подвергается временной ряд  $T = (t_1, \dots, t_n)$ . Предварительная обработка состоит из следующих последовательно выполняемых шагов: поиск сниппетов и аугментация сниппетов, рассматриваемых далее в разделах 2.2 и 2.3 соответственно.

*Распознавание* реализуется с помощью сверточной нейронной сети (Convolutional Neural Network, CNN). На вход данной сети подается вектор из  $m$  значений, предшествующих текущему элементу  $t_n = \text{NULL}$ , т.е. подпоследовательность  $T_{n-m, m}$ . Распознаватель выдает сниппет, на который более всего похожа указанная подпоследовательность. Описание методов реализации Распознавателя приведено в разделе 2.4.

*Реконструкция* реализуется с помощью нейронной сети на основе механизма управляемых рекуррентных блоков (Gated Recurrent Units, GRU). На вход данной сети подается

вектор, представляющий собой конкатенацию двух следующих векторов: снippet, являющийся результатом этапа распознавания, и вектор из  $m - 1$  значений, предшествующих текущему элементу  $t_n = \text{NULL}$  (т.е. подпоследовательность  $T_{n-m+1, m}$ ). Реконструктор выдает восстановленное значение  $\tilde{t}_n$ . Описание реализации Реконструктора приведено в разделе 2.5.

## 2.2. Поиск снippetов

Для дальнейшего изложения введем понятие подпоследовательности следующим образом. *Подпоследовательность (subsequence)  $T_{i, m}$*  временного ряда  $T$  представляет собой непрерывное подмножество  $T$  из  $m$  элементов, начиная с позиции  $i$ :  $T_{i, m} = (t_i, \dots, t_{i+m-1})$ ,  $1 \leq m \ll n$ ,  $1 \leq i \leq m - n + 1$ .

Концепция *снippetа (snippet)* [8] уточняет понятие типичной подпоследовательности временного ряда следующим образом. Временной ряд  $T$  может быть разбит на сегменты заданной длины  $m$ , где каждый сегмент  $S_i$  представляет собой подпоследовательность ряда  $T_{m \cdot (i-1) + 1, m}$ , где  $1 \leq i \leq n/m$ . Здесь и далее без существенного ограничения общности мы можем считать, что  $n$  кратно  $m$ , поскольку  $m \ll n$ .

Каждый снippet представляет собой один из сегментов временного ряда. Со снippetом ассоциируются его ближайшие соседи — подпоследовательности ряда, имеющие ту же длину, что и снippet, которые более похожи на данный снippet, чем на другие сегменты. Для вычисления схожести подпоследовательностей используется специализированная мера схожести, основанная на евклидовом расстоянии. Снippetы упорядочиваются по убыванию мощности множества своих ближайших соседей. Формальное определение снippetов выглядит следующим образом.

Обозначим множество снippetов ряда  $T$ , имеющих длину  $m \ll n$ , как  $C_T^m$ , а элементы этого множества как  $C_1, \dots, C_K$ . Число  $K$  ( $1 \leq K \leq n/m$ ) представляет собой параметр, задаваемый прикладным программистом, и отражает соответствующее количество наиболее типичных снippetов. С каждым снippetом ассоциированы следующие атрибуты: индекс снippetа, ближайшие соседи и значимость данного снippetа.

*Индекс снippetа  $C_i \in C_T^m$*  обозначается как  $C_i.index$  и представляет собой номер  $j$  сегмента  $S_j$ , которому соответствует подпоследовательность ряда  $T_{m \cdot (j-1) + 1, m}$ .

*Множество ближайших соседей снippetа  $C_i \in C_T^m$*  обозначается как  $C_i.NN$  (*Nearest Neighbors*) и содержит подпоследовательности ряда, которые более похожи на данный снippet, чем на другие сегменты ряда, в смысле меры схожести MPdist [25]:

$$C_i.NN = \{T_{j, m} \mid S_{C_i.index} = \arg \min_{1 \leq k \leq n/m} \text{MPdist}(T_{j, m}, S_k), 1 \leq j \leq n - m + 1\}. \quad (1)$$

*Значимость снippetа  $C_i \in C_T^m$*  обозначается как  $C_i.frac$  представляет собой долю множества ближайших соседей снippetа в общем количестве подпоследовательностей ряда, имеющих длину  $m$ :

$$C_i.frac = \frac{|C_i.NN|}{n-m+1}. \quad (2)$$

Снippetы упорядочиваются по убыванию их значимости:

$$\forall C_i, C_j \in C_T^m : i < j \Leftrightarrow C_i.frac \geq C_j.frac. \quad (3)$$

Мера MPdist предложена Кеогом и др. в работе [25], и используется для вычисления схожести подпоследовательностей при нахождении сниппетов. MPdist неформально определяется следующим образом. Два временных ряда равной длины  $m$  тем более похожи друг на друга в смысле меры MPdist, чем больше в каждом из них имеется подпоследовательностей заданной длины  $\ell$  ( $3 \leq \ell \leq m$ ), близких друг к другу в смысле евклидова расстояния. Мера MPdist устойчива к выбросам, шумам и пропущенным значениям во временном ряде, а также инвариантна к амплитуде, сдвигу и фазе временного ряда [25]. Отметим также, что MPdist как симметричная неотрицательная функция является мерой, но не метрикой, поскольку для нее выполнены аксиомы тождества и симметрии, но не выполняется аксиома треугольника [25].

### 2.3. Аугментация сниппетов

Найденные множества ближайших соседей сниппетов являются источником данных выборок для обучения нейронных сетей, реализующих этап распознавания. Мощность указанных множеств может существенно различаться, что влечет за собой дисбаланс прецедентов обучающей выборки, соответствующих различным сниппетам, и низкую точность нейронных сетей, реализующих распознавание и восстановление. В соответствии с этим после нахождения сниппетов выполняется *аугментация (искусственное расширение)* множества ближайших соседей каждого малозначимого сниппета.

Введем параметр *порог аугментации*  $\psi$  ( $0 < \psi < 1$ ), показывающий минимальную значимость сниппета, ближайшие соседи которого должны быть включены в обучающую выборку Распознавателя. Множество ближайших соседей сниппета  $C$ , имеющего значимость  $C_i.frac < \psi$ , подвергается аугментации. Если сниппет  $C_i$  имеет значимость  $C_i.frac > \psi$ , то взятая случайным образом  $(C_i.frac - \psi)$ -я часть ближайших соседей данного сниппета не включается в обучающую выборку. В качестве порога аугментации на практике может быть взято значение  $C_1.frac$  (значимость наиболее важного сниппета).

Дополним сниппет атрибутом *SynthNN (Synthetic Nearest Neighbors)*, который представляет собой множество синтетических ближайших соседей, используемых для аугментации малозначимых сниппетов. Мощность указанного множества вычисляется следующим образом:

$$|C_i.SynthNN| = \begin{cases} \lceil (\psi - C_i.frac) \cdot (n - m + 1) \rceil, & C_i.frac < \psi \\ 0, & otherwise. \end{cases} \quad (4)$$

Аугментация ближайшего соседа сниппета выполняется таким образом, чтобы каждый полученный синтетический образец был отличен от любого из ближайших соседей сниппета. Это свойство обеспечивается, если потребовать, чтобы каждый синтетический образец был похож на сниппет больше, чем аугментируемый ближайший сосед сниппета. Основная идея аугментации проиллюстрирована на рис. 2.

Обозначим ближайшего соседа сниппета  $C_i$  как *neighbor*, величину евклидова расстояния от сниппета до ближайшего соседа обозначим как  $\varepsilon$ :  $\varepsilon = ED(C_i, neighbor)$ . Число  $\varepsilon$  может быть разложено на сумму  $m$  неотрицательных слагаемых в соответствии с алгоритмом, описанным в работе [26]. Указанный алгоритм предполагает, что  $\varepsilon$  разлагается на  $k$  слагаемых, каждое из которых равно  $\varepsilon/k$ , и число  $k$  может принимать значения от 1 до  $m$ . Количество разложений равно  $C_m^{m+k-1}$  [26].

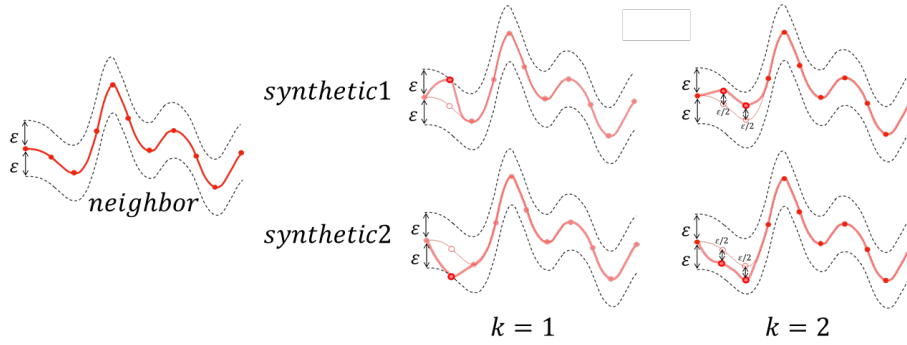


Рис. 2. Аугментация сниппетов

Если мы рассмотрим подпоследовательность *neighbor* и занумерованный набор слагаемых разложения  $\varepsilon$  как два вектора в евклидовом пространстве  $\mathbb{R}^m$ , тогда сумма и разность указанных векторов дает пару синтетических соседей  $S_i$ , отличных от *neighbor*. Обозначим полученные синтетические образцы как *synthetic1* и *synthetic2*, тогда приведенные выше рассуждения могут быть формально описаны следующим образом:

$$\forall neighbor \in C_i.NN \quad \exists c \in \mathbb{R}^m : \sum_{k=1}^m c_k = \varepsilon, \quad (5)$$

$$synthetic1 = neighbor + c, \quad synthetic2 = neighbor - c,$$

$$ED(C_i, synthetic1) < \varepsilon, \quad ED(C_i, synthetic2) < \varepsilon.$$

Без существенного ограничения общности мы можем считать, что ближайшие соседи каждого сниппета  $S_i$  упорядочены по убыванию схожести с данным сниппетом. Для аугментации возьмем  $\lceil \frac{1}{2}|C_i.SynthNN| \rceil$  последних элементов множества ближайших соседей (наименее похожих на сниппет), чтобы обеспечить лучшее обучение нейронной сети распознаванию подпоследовательностей, слабо похожих на сниппет. Вектор  $c \in \mathbb{R}^m$ , упомянутый в формуле (4), формируется следующим образом. Возьмем случайное натуральное число  $q$  ( $1 \leq q \leq m$ ), затем возьмем случайные не повторяющиеся натуральные числа  $p_1, \dots, p_q$  ( $1 \leq p_i \leq m$ ). Тогда координаты вектора  $c \in \mathbb{R}^m$  определяются следующим образом:

$$c \in \mathbb{R}^m, c_i = \begin{cases} \varepsilon/q, & i \in \{p_1, \dots, p_q\} \\ 0, & otherwise. \end{cases} \quad (6)$$

Описанный выше подход к аугментации сниппетов может быть применен для решения широкого спектра конкретных прикладных задач интеллектуального анализа временных рядов с помощью технологий искусственных нейронных сетей, где требуется синтетическая генерация исходных данных для обучающих выборок.

## 2.4. Распознавание

В дальнейшем изложении мы будем обозначать обучающую выборку как множество пар  $D = \{ \langle X; Y \rangle \}$ , где  $X$  представляет собой входные данные, а  $Y$  — соответствующие им выходные данные.

Кортеж обучающей выборки для нейронной сети Распознавателя формируется следующим образом. Входным данным полагается ближайший сосед сниппета, выходным дан-

ным — номер соответствующего снippetsа. Выборка формируется по всем снippetsам и включает в себя как реальных, так и синтетических ближайших соседей снippetsа:

$$D_{Recognizer} = \{ \langle x; i \rangle \mid x \in C_i.NN \cup C_i.SynthNN, 1 \leq i \leq K \}. \quad (7)$$

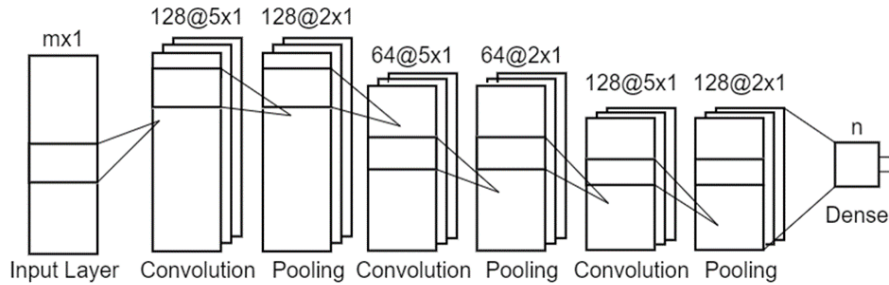


Рис. 3. Топология нейронной сети Распознавателя

На рис. 3 представлена архитектура нейронной сети, реализующей этап распознавания. Нейронная сеть включает в себя три сверточных слоя, каждый из которых чередуется со слоем пулинга на основе усредняющей функции. На каждом слое свертки применяется ядро размера  $5 \times 1$ , на слое пулинга — ядро размера  $2 \times 1$ . Сверточные слои нейронной сети содержат 128, 64 и 128 нейронов соответственно. Для предотвращения переобучения нейронной сети выполняется прореживание (Dropout) 5% нейронов внутренних слоев, за исключением последнего из них, в котором отбрасывается 25% нейронов.

Размерность входного слоя нейронной сети  $m$  совпадает с длиной снippetsа. На вход сети подается вектор из  $m$  значений, предшествующих текущему (восстанавливаемому) элементу временного ряда. Размерность выходного слоя Dense совпадает со значением параметра  $K$  (количество наиболее значимых снippetsов), заданным на этапе предварительной обработки. На выходе нейронной сети формируется  $K$ -мерный вещественный вектор,  $i$ -й элемент которого отражает вероятность схожести входного вектора на снippets  $S_i$ . В качестве результата Распознаватель выдает снippets с максимальной вероятностью схожести.

## 2.5. Реконструкция

Кортеж обучающей выборки для нейронной сети Реконструктора формируется следующим образом. Элементом входных данных полагается вектор, который является конкатенацией двух векторов: снippets и ближайший сосед снippetsа, из которого исключен последний элемент. Элементом выходных данных является исключенный элемент вышеуказанного ближайшего соседа. Выборка формируется по всем снippetsам и включает в себя как реальных, так и синтетических ближайших соседей снippetsа:

$$D_{Reconstructor} = \{ \langle C_i \cdot (x_1, \dots, x_{m-1}); x_m \rangle \mid x \in C_i.NN \cup C_i.SynthNN, 1 \leq i \leq K \}. \quad (8)$$

На рис. 4 представлена архитектура нейронной сети, реализующей Реконструктор. Нейронная сеть включает в себя три слоя: входной слой, скрытый слой и выходной слой. Размерность входного слоя нейронной сети  $m$  совпадает с длиной снippetsа. Скрытый слой состоит 128 нейронов, каждый из которых представляет собой управляемый рекуррент-

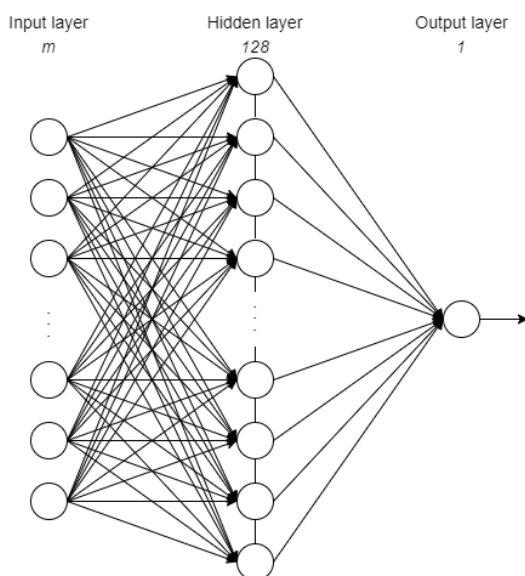


Рис. 4. Топология нейронной сети Реконструктора

ный блок (Gated Recurrent Units, GRU) [27]. Выходной слой состоит из одного нейрона, результатом которого является предсказанное значение  $\tilde{t}_n$ .

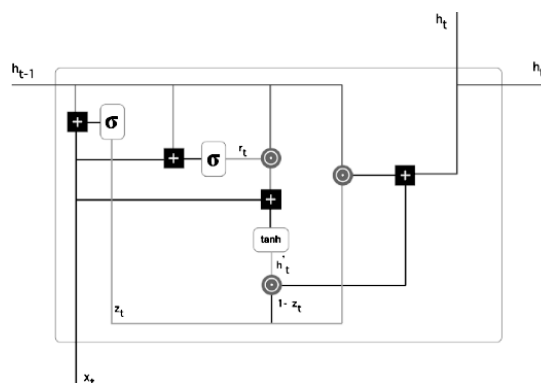


Рис. 5. Структура управляемого рекуррентного блока (операции:  $\odot$  — произведение Адамара,  $\oplus$  — сложение; функции:  $\sigma$  — сигмоида,  $\tanh$  — гиперболический тангенс; векторы (на шаге  $t$ ):  $x_t$  — входной,  $h_t$  — скрытое состояние,  $r_t$  — сброс,  $z_t$  — обновление)

Структура GRU-нейрона представлена на рис. 5. GRU используется для решения проблемы «исчезающего градиента» [28], которая возникает при обучении искусственных нейронных сетей с помощью методов обучения на основе градиента и обратного распространения ошибки. В указанных методах каждый из весов нейронной сети получает обновление, пропорциональное частной производной функции ошибок по текущему весу на каждой итерации обучения. Проблема заключается в том, что градиент может иметь настолько малые значения, что это предотвращает изменение значения веса. В худшем случае это может полностью остановить дальнейшее обучение нейронной сети.

Управляемый рекуррентный блок содержит *вектор сброса* состояний, который позволяет «забыть» информацию, неуместную в будущем:

$$r_t = \sigma(x_t W_{xr} + h_{t-1} W_{hr} + b_r). \quad (9)$$

Для контроля информации, которую необходимо передавать из предыдущего состояния, применяется *вектор обновления*, вычисляемый аналогичным образом:

$$z_t = \sigma(x_t W_{zr} + h_{t-1} W_{hz} + b_z). \quad (10)$$

Далее, исходя из этого, вычисляется *текущее состояние*:

$$\begin{aligned} h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot h'_t, \\ h'_t &= \tanh(x_t W_{xh} + (r_t \odot h_{t-1}) W_{hh} + b_h). \end{aligned} \quad (11)$$

Таким образом, если вектор сброса принимает значение, близкое к нулю, то нейрон «забывает» информацию о предыдущем состоянии и рассчитывает состояние так, как если бы  $x_t$  было бы началом новой подпоследовательности временного ряда. Благодаря наличию векторов сброса и обновления градиент не принимает значения ниже определенной величины, определяемой значением сигнала сохраняемого контента.

### 3. Вычислительные эксперименты

#### 3.1. Цели и наборы данных

Для оценки эффективности метода SANNI нами были проведены вычислительные эксперименты, в которых исследовалась точность восстановления пропущенных значений во временных рядах из различных предметных областей, в том числе в сравнении с аналогичными методами, а также влияние параметров метода на точность восстановления.

Для оценки точности восстановления пропущенных значений нами используется мера среднеквадратичной ошибки *RMSE (Root Mean Square Error)*, определяемая следующим образом:

$$RMSE = \sqrt{\frac{1}{h} \sum_{i=1}^h (t_i - \hat{t}_i)^2}, \quad (12)$$

где  $t_i$  — фактическое значение временного ряда,  $\hat{t}_i$  — восстановленное значение,  $h$  — количество пропущенных значений.

**Таблица 1.** Временные ряды для экспериментов

№ п/п	Временной ряд	Длина ряда $n$	Семантика
1	HumanActivity-2	7 002	Различные виды физической активности человека
2	HumanActivity-12	100 800	
3	AppliancesEnergy	100 000	Энергопотребление бытовых приборов в доме в течение 2 месяцев
4	Household-Power	100 000	Энергопотребление домашнего хозяйства в течение 4 лет
5	Household-Voltage	100 000	
6	Household-Kitchen	100 000	

Для проведения экспериментов были использованы следующие наборы данных, резюмированные в табл. 1. Ряды HumanActivity-2 и HumanActivity-12 получены на основе набора данных РАМАР [29], который представляет собой записи о физической активности различ-



ных людей (бег, ходьба, прыжки и др., всего 12 видов активности). Ряд HumanActivity-2 содержит записи о двух активностях (ходьба, сменяющаяся бегом) одного человека. Ряд HumanActivity-12 соединяет в себе записи о 12 активностях одного человека.

Ряд AppliancesEnergy представляет собой показания общего энергопотребления бытовых приборов дома и получен на основе набора данных, использованного в работе [30].

Мультивариативный ряд Household отражает энергопотребление домашнего хозяйства (посудомоечная машина, микроволновая печь, духовка, стиральная машина, водонагреватель, кондиционер и др.) в течение 4 лет [31] и включает в себя следующие ряды: Power — суммарная мощность потребленной энергии в минуту, Voltage — суммарное напряжение потребленной энергии в минуту, Kitchen — общая мощность энергии, потребленной кухонными приборами.

### 3.2. Сравнение с аналогами

В проведенных экспериментах конкурентами метода SANNI выступали как аналитические алгоритмы, так и методы на основе нейронных сетей. В качестве соперников-аналитических алгоритмов были рассмотрены Mean-, Median-, Mode-imputation (восстановление с помощью среднего, медианы и моды соответственно), ImputeTS (библиотека алгоритмов на языке R для одноименной среды интеллектуального анализа данных, выполняющих восстановление пропусков в одномерных временных рядах) [32] и  $k$ NNI<sup>1</sup> [10]. Нейросетевые методы-конкуренты — BRITS [23] и ANNI. Метод ANNI реализован авторами данного исследования как нейронная сеть из GRU-нейронов и представляет собой метод SANNI, из которого исключены поиск и аугментация сниппетов, а также распознавание подпоследовательностей временного ряда на основе сниппетов. Метод BRITS предназначен для восстановления в мультивариативных временных рядах, поэтому в экспериментах с этим методом нами была выполнена 3-кратная репликация рядов HumanActivity и AppliancesEnergy: на вход BRITS подавался мультивариативный ряд из трех одинаковых значений.

Таблица 2. Параметры экспериментов

№ п/п	Временной ряд	Длина сегмента $m$	Количество сниппетов $K$
1	HumanActivity-2	100	2
2	HumanActivity-12	300	12
3	AppliancesEnergy	1 440	2
4	Household-Power	60	2
5	Household-Voltage	60	2
6	Household-Kitchen	60	2

В табл. 2 приведены параметры метода SANNI длина сегмента  $m$  и количество наиболее значимых сниппетов  $K$ , использованные в экспериментах. В экспериментах в качестве порога аугментации (см. раздел 2.3) было взято значение  $\psi = 0.01$ . Для формирования обучающей и тестовой выборок использовались 70% и 30% случайно выбранных подпоследовательностей исходного временного ряда соответственно.

<sup>1</sup>В экспериментах взято значение параметра  $k = 10$ , обеспечившее наибольшую точность восстановления.

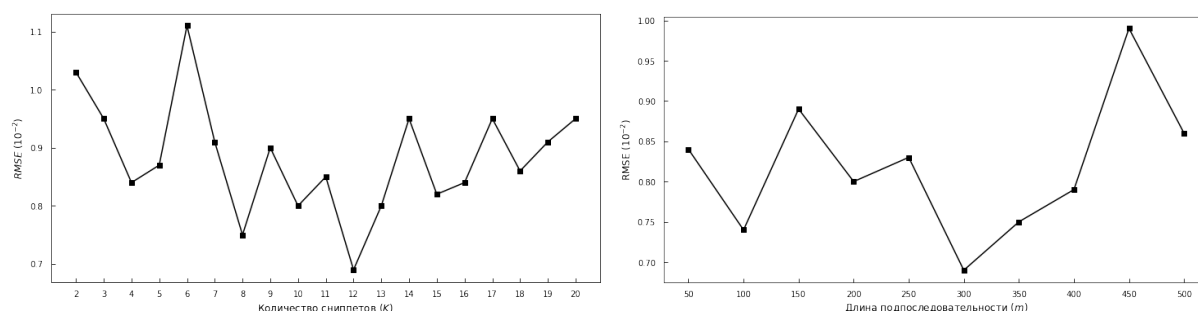
**Таблица 3.** Точность восстановления ( $RMSE \cdot 10^{-2}$ ) приведены параметры метода SANNI длина сегмента  $m$  и количество наиболее значимых сниппетов  $K$ , использованные в экспериментах

Метод	Human Activity-2	Human Activity-12	Appliances Energy	Household		
				Power	Voltage	Kitchen
Аналитические методы						
Mean	10.29	3.85	14.58	13.37	12.24	8.76
Median	10.56	3.92	14.77	13.44	12.28	8.87
Mode	10.56	3.92	21.21	19.86	12.44	8.77
$k$ NNI	3.93	1.88	4.99	5.95	3.23	3.42
imputeTS	11.05	4.16	20.36	18.38	11.04	9.15
Нейросетевые методы						
ANNI	3.53	0.98	<b>0.69</b>	<b>4.36</b>	<b>2.60</b>	<b>2.20</b>
BRITS	8.94	9.63	6.57	3.74		
<b>SANNI</b>	<b>2.84</b>	<b>0.69</b>	0.72	4.38	2.67	2.44

Результаты экспериментов по сравнению точности восстановления метода SANNI с аналогами приведены в табл. 3. Можно видеть, что на временных рядах HumanActivity предлагаемый в данной работе метод SANNI опережает как аналитические алгоритмы, так и нейросетевые методы. Для остальных временных рядов лучшую точность восстановления показывает метод ANNI.

### 3.3. Влияние параметров

В данном разделе представлены результаты экспериментов по исследованию влияния параметров метода SANNI, количества сниппетов  $K$  и длина сниппета  $m$ , на точность восстановления. Эксперименты проведены на временном ряде HumanActivity-12 (см. табл. 1). В экспериментах в качестве порога аугментации (см. раздел 2.3) было взято значение  $\psi = 0.01$ . Для формирования обучающей и тестовой выборки использовались 70% и 30% случайно выбранных подпоследовательностей исходного временного ряда соответственно.



**Рис. 6.** Зависимость точности восстановления от параметров метода SANNI

Зависимость точности восстановления с помощью метода SANNI от количества сниппетов представлена на рис. 6. Можно видеть, что наиболее высокую точность метод показывает при  $K = 12$ , когда количество сниппетов совпадает с фактическим значением различных видов физической активности человека, представленных в ряде (см. табл. 2).

Метод SANNI показывает наиболее высокую точность, когда параметр равен фактическому значению ( $m = 300$ ), а при отклонении от такового точность уменьшается.

## Заключение

В статье рассмотрена проблема восстановления пропущенных значений потокового временного ряда в режиме реального времени. Данная задача возникает в широком спектре практических приложений цифровой индустрии и интернета вещей, в которых пропуски в показаниях датчиков недопустимы и должны быть незамедлительно заменены на правдоподобные синтетические значения.

Авторами предложен новый метод восстановления пропущенных значений потокового временного ряда в режиме реального времени, названный SANNI (Snippet and Neural Network based Imputation). Метод SANNI предполагает следующие этапы восстановления пропусков: предварительная обработка данных, распознавание и реконструкция. Предварительная обработка предполагает однократную подготовку обучающих выборок данных. Распознавание и реконструкция реализуются с помощью нейронных сетей, обучаемых на указанных выборках.

Предварительная обработка включает в себя поиск сниппетов и аугментацию сниппетов. Сниппет [8] представляет собой уточнение понятия типичной подпоследовательности временного ряда. Временной ряд логически разбивается на сегменты (непересекающиеся подпоследовательности) заданной длины, сниппет — один из сегментов ряда. Со сниппетом ассоциируются его ближайшие соседи — подпоследовательности ряда, имеющие ту же длину, что и сниппет, которые более похожи на данный сниппет, чем на другие сегменты. Для вычисления схожести подпоследовательностей используется специализированная мера схожести MPdist [25], основанная на евклидовом расстоянии. Сниппеты упорядочиваются по убыванию мощности множества своих ближайших соседей. Далее выполняется аугментация (искусственное расширение) каждого маломощного множества ближайших соседей сниппета, которая обеспечивает сбалансированность обучающих выборок нейронных сетей, реализующих распознавание и реконструкцию.

Распознавание реализуется с помощью сверточной нейронной сети. На вход данной сети подается вектор из элементов временного ряда, предшествующих пропущенному элементу; длина указанного вектора совпадает с длиной сниппета. Распознаватель выдает сниппет, на который более всего похожа входная подпоследовательность. Реконструкция реализуется с помощью нейронной сети из управляемых рекуррентных блоков (Gated Recurrent Units, GRU). На вход данной сети подается вектор, представляющий собой конкатенацию двух следующих векторов: сниппет, являющийся результатом этапа распознавания, и вектор элементов ряда, предшествующих пропуску (длина указанного вектора на единицу меньше длины сниппета). Реконструктор выдает восстановленное значение.

Представлены результаты экспериментов на реальных временных рядах, показывающих высокую точность восстановления метода SANNI и его преимущество перед аналогами.

В качестве возможного направления будущих исследований мы рассматриваем внедрение в метод SANNI адаптивного автоматизированного подбора параметра, отвечающего за длину сниппета.

*Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (грант № 20-07-00140) и Министерства науки и высшего образования РФ (государственное задание FENU-2020-0022).*

## Литература

1. Kumar S., Tiwari P., Zymbler M. Internet of Things is a revolutionary approach for future technology enhancement: a review // *J. Big Data*. 2019. Vol. 6. P. 111. DOI: 10.1186/s40537-019-0268-2.
2. Xu L.D., Duan L. Big Data for cyber physical systems in Industry 4.0: A survey // *Enterp. Inf. Syst.* 2019. Vol. 13, no. 2. P. 148–169. DOI: 10.1080/17517575.2018.1442934.
3. Цымблер М.Л., Краева Я.А., Латыпова Е.А. и др. Очистка сенсорных данных в интеллектуальных системах управления отоплением зданий // *Вестник ЮУрГУ. Серия: Вычислительная математика и информатика*. 2021. Т. 10, № 3. С. 16–36. DOI: 10.14529/cmse210302.
4. Иванов С.А., Никольская К.Ю., Радченко Г.И. и др. Концепция построения цифрового двойника города // *Вестник ЮУрГУ. Серия: Вычислительная математика и информатика*. 2020. Т. 9, № 4. С. 5–23. DOI: 10.14529/cmse200401.
5. Епишев В.В., Исаев А.П., Минахметов Р.М. и др. Система интеллектуального анализа данных физиологических исследований в спорте высших достижений // *Вестник ЮУрГУ. Серия: Вычислительная математика и информатика*. 2013. Т. 2, № 1. С. 44–54. DOI: 10.14529/cmse130105.
6. Абдуллаев С.М., Ленская О.Ю., Гаязова А.О. и др. Алгоритмы краткосрочного прогноза с использованием радиолокационных данных: оценка траектории и композиционный дисплей жизненного цикла // *Вестник ЮУрГУ. Серия: Вычислительная математика и информатика*. 2014. Т. 3, № 1. С. 17–32. DOI: 10.14529/cmse140102.
7. Дышаев М.М., Соколинская И.М. Представление торговых сигналов на основе адаптивной скользящей средней Кауфмана в виде системы линейных неравенств // *Вестник ЮУрГУ. Серия: Вычислительная математика и информатика*. 2013. Т. 2, № 4. С. 103–108. DOI: 10.14529/cmse130408.
8. Imani S., Madrid F., Ding W. *et al.* Introducing time series snippets: a new primitive for summarizing long time series // *Data Min. Knowl. Discov.* 2020. Vol. 34, no. 6. P. 1713–1743. DOI: 10.1007/s10618-020-00702-y.
9. Sande I.G. Hot-deck imputation procedures // *Incomplete data in sample surveys*. 1983. Vol. 3. P. 339–349.
10. Batista G.E.A.P.A., Monard M.C. An Analysis of Four Missing Data Treatment Methods for Supervised Learning // *Appl. Artif. Intell.* 2003. Vol. 17, no. 5-6. P. 519–533. DOI: 10.1080/713827181.
11. de Carvalho Jr. O.A., Guimarães R.F., Gomes R.A.T., da Silva N.C. Time series interpolation // *IEEE International Geoscience & Remote Sensing Symposium, IGARSS 2007, Barcelona, Spain, July 23–28, 2007. Proceedings. IEEE, 2007*. P. 1959–1961. DOI: 10.1109/IGARSS.2007.4423211.
12. Yi B., Sidiropoulos N.D., Johnson T. *et al.* Online Data Mining for Co-Evolving Time Sequences // *Proceedings of the 16th International Conference on Data Engineering, San Diego*,

- California, USA, February 28 – March 3, 2000 / Ed. by Lomet D.B., Weikum G. IEEE Computer Society, 2000. P. 13–22. DOI: 10.1109/ICDE.2000.839383.
13. Paulhus J., Kohler M. Interpolation of missing precipitation records // Monthly Weather Review. 1952. Vol. 80, no. 8. P. 129–133. DOI: 10.1175/1520-0493(1952)080<0129:IOMPR>2.0.CO;2.
14. Box G.E., Jenkins G.M., Reinsel G.C., Ljung G.M. Time Series Analysis: Forecasting and Control, 5th Edition. John Wiley & Sons, 2015. 712 p.
15. Troyanskaya O.G., Cantor M.N., Sherlock G. *et al.* Missing value estimation methods for DNA microarrays // Bioinform. 2001. Vol. 17, no. 6. P. 520–525. DOI: 10.1093/bioinformatics/17.6.520.
16. Khayati M., Böhlen M.H. REBOM: Recovery of Blocks of Missing Values in Time Series // Proceedings of the 18th International Conference on Management of Data, COMAD 2012, Pune, India / Ed. by Sahasrabuddhe C., Abbadi A.E., Murthy K., Bhattacharya A. Computer Society of India, 2012. P. 44–55. URL: <http://comad.in/comad2012/pdf/khayati.pdf>.
17. Khayati M., Böhlen M.H., Gamper J. Memory-efficient centroid decomposition for long time series // IEEE 30th International Conference on Data Engineering, Chicago, ICDE 2014, IL, USA, March 31 – April 4, 2014 / Ed. by Cruz I.F., Ferrari E., Tao Y. *et al.* IEEE Computer Society, 2014. P. 100–111. DOI: 10.1109/ICDE.2014.6816643.
18. Papadimitriou S., Sun J., Faloutsos C., Yu P.S. Dimensionality Reduction and Filtering on Time Series Sensor Streams // Managing and Mining Sensor Data / Ed. by Aggarwal C.C. Springer, 2013. P. 103–141. DOI: 10.1007/978-1-4614-6309-2\_5.
19. Pearson K. On lines and planes of closest fit to systems of points in space // Philosophical Magazine. 1901. Vol. 2. P. 559–572. DOI: 10.1080/14786440109462720.
20. Wellenzohn K., Böhlen M.H., Dignös A. *et al.* Continuous Imputation of Missing Values in Streams of Pattern-Determining Time Series // Proceedings of the 20th International Conference on Extending Database Technology, EDBT 2017, Venice, Italy, March 21–24, 2017 / Ed. by Markl V., Orlando S., Mitschang B. *et al.* OpenProceedings.org, 2017. P. 330–341. DOI: 10.5441/002/edbt.2017.30.
21. Hsu H., Yang A.C., Lu M. KNN-DTW Based Missing Value Imputation for Microarray Time Series Data // J. Comput. 2011. Vol. 6, no. 3. P. 418–425. DOI: 10.4304/jcp.6.3.418-425.
22. Berndt D.J., Clifford J. Using Dynamic Time Warping to Find Patterns in Time Series // Knowledge Discovery in Databases: Papers from the 1994 AAAI Workshop, Seattle, Washington, USA, July 1994. Technical Report WS-94-03 / Ed. by Fayyad U.M., Uthurusamy R. AAAI Press, 1994. P. 359–370.
23. Cao W., Wang D., Li J. *et al.* BRITS: Bidirectional Recurrent Imputation for Time Series // Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, Montréal, Canada, December 3–8, 2018 / Ed. by Bengio S., Wallach H.M., Larochelle H. *et al.* 2018. P. 6776–6786. URL: <https://proceedings.neurips.cc/paper/2018/hash/734e6bfcd358e25ac1db0a4241b95651-Abstract.html>.

24. Guo Z., Wan Y., Ye H. A data imputation method for multivariate time series based on generative adversarial network // *Neurocomputing*. 2019. Vol. 360. P. 185–197. DOI: 10.1016/j.neucom.2019.06.007.
25. Gharghabi S., Imani S., Bagnall A.J. *et al.* An ultra-fast time series distance measure to allow data mining in more complex real-world deployments // *Data Min. Knowl. Discov.* 2020. Vol. 34, no. 4. P. 1104–1135. DOI: 10.1007/s10618-020-00695-8.
26. Reingold E., Nievergelt J., Deo N. *Combinatorial Algorithms: Theory and Practice*. Prentice Hall, 1977. 433 p.
27. Cho K., van Merriënboer B., Gülçehre Ç. *et al.* Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation // *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, Doha, Qatar, October 25–29, 2014, A meeting of SIGDAT, a Special Interest Group of the ACL / Ed. by Moschitti A., Pang B., Daelemans W.* ACL, 2014. P. 1724–1734. DOI: 10.3115/v1/d14-1179.
28. Hochreiter S., Bengio Y., Frasconi P., Schmidhuber J. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies // *A Field Guide to Dynamical Recurrent Neural Networks / Ed. by Kremer S., Kolen J.* Wiley-IEEE Press, 2001. P. 237–243. DOI: 10.1109/9780470544037.ch14.
29. Reiss A., Stricker D. Introducing a New Benchmarked Dataset for Activity Monitoring // *16th International Symposium on Wearable Computers, ISWC 2012, Newcastle, United Kingdom, June 18–22, 2012.* IEEE Computer Society, 2012. P. 108–109. DOI: 10.1109/ISWC.2012.13.
30. Candanedo L., Feldheim V., Deramaix D. Data driven prediction models of energy use of appliances in a low-energy house // *Energy and Buildings*. 2017. Vol. 140. P. 81–97. DOI: 10.1016/j.enbuild.2017.01.083.
31. Individual household electric power consumption data set. UCI Machine Learning Repository. URL: <https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption> (дата обращения: 03.09.2021).
32. Moritz S., Bartz-Beielstein T. imputeTS: Time Series Missing Value Imputation in R // *R Journal*. 2017. Vol. 9, no. 1. P. 207. DOI: 10.32614/rj-2017-009.

Цымблер Михаил Леонидович, д.ф.-м.н., доцент, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Полонский Вячеслав Александрович, магистрант, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Юртин Алексей Артемьевич, магистрант, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

## ON ONE METHOD OF IMPUTATION MISSING VALUES OF A STREAMING TIME SERIES IN REAL TIME

© 2021 M.L. Zymbler, V.A. Polonsky, A.A. Yurtin

*South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)*

*E-mail: mzym@susu.ru, s.polonski@mail.ru, lideor@yandex.ru*

Received: 03.09.2021

The problem of the imputation of missing values in a streaming time series arises in a wide range of Industry 4.0 and Internet of Things applications. In the article, we propose a novel imputation method based on time series mining techniques and artificial neural networks. The method involves three steps of imputation: data preprocessing, recognition, and reconstruction. Preprocessing is a one-time preparation of training data samples. Recognition and reconstruction are implemented through two neural networks trained on the samples above. Preprocessing supposes the discovery of a set of typical subsequences (snippets) in a pre-stored fragment of the streaming time series without misses. Recognition is implemented through a Convolutional Neural Network, and its input is a vector of the elements preceding the current (missing) value. The Recognizer outputs the snippet that the input subsequence is most similar to. Reconstruction is implemented through a Recurrent Neural Network, and its input is a concatenation of the Recognizer's output and the vector of the elements preceding the missing value. The Reconstructor outputs the value to be imputed. The experimental results show high accuracy and the advantage of the proposed method over analogs.

*Keywords: time series, imputation of missing values, online mode, artificial neural networks, CNN, RNN, time series snippets.*

### FOR CITATION

Zymbler M.L., Polonsky V.A., Yurtin A.A. On One Method of Imputation Missing Values of a Streaming Time Series in Real Time. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2021. Vol. 10, no. 4. P. 5–25. (in Russian) DOI: 10.14529/cmse210401.

*This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.*

### References

1. Kumar S., Tiwari P., Zymbler M. Internet of Things is a revolutionary approach for future technology enhancement: a review. J. Big Data. 2019. Vol. 6. P. 111. DOI: 10.1186/s40537-019-0268-2.
2. Xu L.D., Duan L. Big Data for cyber physical systems in Industry 4.0: A survey. Enterp. Inf. Syst. 2019. Vol. 13, no. 2. P. 148–169. DOI: 10.1080/17517575.2018.1442934.
3. Zymbler M.L., Kraeva Y.A., Latypova E.A. *et al.* Cleaning sensor data in intelligent heating control system. Bulletin of the South Ural State University. Series: Computational Mathematics and Computational Mathematics and Software Engineering. 2021. Vol. 10, no. 3. P. 16–36. (in Russian) DOI: 10.14529/cmse210302.
4. Ivanov S.A., Nikolskaya K.Y., Radchenko G.I. *et al.* Digital twin of a city: concept overview. Bulletin of the South Ural State University. Series: Computational Mathematics and Com-

- putational Mathematics and Software Engineering. 2020. Vol. 9, no. 4. P. 5–23. (in Russian) DOI: 10.14529/cmse200401.
5. Epishev V.V., Isaev A.P., Miniakhmetov R.M. *et al.* Physiological data mining system for elite sports. Bulletin of the South Ural State University. Computational Mathematics and Software Engineering. 2013. Vol. 2, no. 1. P. 44–54. (in Russian) DOI: 10.14529/cmse130105.
  6. Abdullaev S.M., Lenskaya O.Y., Gayazova A.O. *et al.* Short-range forecasting algorithms using radar data: translation estimate and life-cycle composite display. Bulletin of the South Ural State University. Series: Computational Mathematics and Computational Mathematics and Software Engineering. 2014. Vol. 3, no. 1. P. 17–32. (in Russian) DOI: 10.14529/cmse140102.
  7. Dyshaev M.M., Sokolinskaya I.M. Representation of trading signals based on the Kaufman's Adaptive Moving Average in the form of a system of linear inequalities. Bulletin of the South Ural State University. Series: Computational Mathematics and Computational Mathematics and Software Engineering. 2013. Vol. 2, no. 4. P. 103–108. (in Russian) DOI: 10.14529/cmse130408.
  8. Imani S., Madrid F., Ding W. *et al.* Introducing time series snippets: a new primitive for summarizing long time series. Data Min. Knowl. Discov. 2020. Vol. 34, no. 6. P. 1713–1743. DOI: 10.1007/s10618-020-00702-y.
  9. Sande I.G. Hot-deck imputation procedures. Incomplete data in sample surveys. 1983. Vol. 3. P. 339–349.
  10. Batista G.E.A.P.A., Monard M.C. An Analysis of Four Missing Data Treatment Methods for Supervised Learning. Appl. Artif. Intell. 2003. Vol. 17, no. 5-6. P. 519–533. DOI: 10.1080/713827181.
  11. de Carvalho Jr. O.A., Guimarães R.F., Gomes R.A.T., da Silva N.C. Time series interpolation. IEEE International Geoscience & Remote Sensing Symposium, IGARSS 2007, Barcelona, Spain, July 23–28, 2007. Proceedings. IEEE, 2007. P. 1959–1961. DOI: 10.1109/IGARSS.2007.4423211.
  12. Yi B., Sidiropoulos N.D., Johnson T. *et al.* Online Data Mining for Co-Evolving Time Sequences. Proceedings of the 16th International Conference on Data Engineering, San Diego, California, USA, February 28 – March 3, 2000 / Ed. by Lomet D.B., Weikum G. IEEE Computer Society, 2000. P. 13–22. DOI: 10.1109/ICDE.2000.839383.
  13. Paulhus J., Kohler M. Interpolation of missing precipitation records. Monthly Weather Review. 1952. Vol. 80, no. 8. P. 129–133. DOI: 10.1175/1520-0493(1952)080<0129:IOMPR>2.0.CO;2.
  14. Box G.E., Jenkins G.M., Reinsel G.C., Ljung G.M. Time Series Analysis: Forecasting and Control, 5th Edition. John Wiley & Sons, 2015. 712 p.
  15. Troyanskaya O.G., Cantor M.N., Sherlock G. *et al.* Missing value estimation methods for DNA microarrays. Bioinform. 2001. Vol. 17, no. 6. P. 520–525. DOI: 10.1093/bioinformatics/17.6.520.



16. Khayati M., Böhlen M.H. REBOM: Recovery of Blocks of Missing Values in Time Series. Proceedings of the 18th International Conference on Management of Data, COMAD 2012, Pune, India / Ed. by Sahasrabuddhe C., Abbadi A.E., Murthy K., Bhattacharya A. Computer Society of India, 2012. P. 44–55. URL: <http://comad.in/comad2012/pdf/khayati.pdf>.
17. Khayati M., Böhlen M.H., Gamper J. Memory-efficient centroid decomposition for long time series. IEEE 30th International Conference on Data Engineering, Chicago, ICDE 2014, IL, USA, March 31 – April 4, 2014 / Ed. by Cruz I.F., Ferrari E., Tao Y. *et al.* IEEE Computer Society, 2014. P. 100–111. DOI: 10.1109/ICDE.2014.6816643.
18. Papadimitriou S., Sun J., Faloutsos C., Yu P.S. Dimensionality Reduction and Filtering on Time Series Sensor Streams. Managing and Mining Sensor Data / Ed. by Aggarwal C.C. Springer, 2013. P. 103–141. DOI: 10.1007/978-1-4614-6309-2\_5.
19. Pearson K. On lines and planes of closest fit to systems of points in space. Philosophical Magazine. 1901. Vol. 2. P. 559–572. DOI: 10.1080/14786440109462720.
20. Wellenzohn K., Böhlen M.H., Dignös A. *et al.* Continuous Imputation of Missing Values in Streams of Pattern-Determining Time Series. Proceedings of the 20th International Conference on Extending Database Technology, EDBT 2017, Venice, Italy, March 21–24, 2017 / Ed. by Markl V., Orlando S., Mitschang B. *et al.* OpenProceedings.org, 2017. P. 330–341. DOI: 10.5441/002/edbt.2017.30.
21. Hsu H., Yang A.C., Lu M. KNN-DTW Based Missing Value Imputation for Microarray Time Series Data. J. Comput. 2011. Vol. 6, no. 3. P. 418–425. DOI: 10.4304/jcp.6.3.418-425.
22. Berndt D.J., Clifford J. Using Dynamic Time Warping to Find Patterns in Time Series. Knowledge Discovery in Databases: Papers from the 1994 AAAI Workshop, Seattle, Washington, USA, July 1994. Technical Report WS-94-03 / Ed. by Fayyad U.M., Uthurusamy R. AAAI Press, 1994. P. 359–370.
23. Cao W., Wang D., Li J. *et al.* BRITS: Bidirectional Recurrent Imputation for Time Series. Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, Montréal, Canada, December 3–8, 2018 / Ed. by Bengio S., Wallach H.M., Larochelle H. *et al.* 2018. P. 6776–6786. URL: <https://proceedings.neurips.cc/paper/2018/hash/734e6bfcd358e25ac1db0a4241b95651-Abstract.html>.
24. Guo Z., Wan Y., Ye H. A data imputation method for multivariate time series based on generative adversarial network. Neurocomputing. 2019. Vol. 360. P. 185–197. DOI: 10.1016/j.neucom.2019.06.007.
25. Gharghabi S., Imani S., Bagnall A.J. *et al.* An ultra-fast time series distance measure to allow data mining in more complex real-world deployments. Data Min. Knowl. Discov. 2020. Vol. 34, no. 4. P. 1104–1135. DOI: 10.1007/s10618-020-00695-8.
26. Reingold E., Nievergelt J., Deo N. Combinatorial Algorithms: Theory and Practice. Prentice Hall, 1977. 433 p.

27. Cho K., van Merriënboer B., Gülçehre Ç. *et al.* Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, Doha, Qatar, October 25–29, 2014, A meeting of SIGDAT, a Special Interest Group of the ACL / Ed. by Moschitti A., Pang B., Daelemans W. ACL, 2014. P. 1724–1734. DOI: 10.3115/v1/d14-1179.
28. Hochreiter S., Bengio Y., Frasconi P., Schmidhuber J. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. A Field Guide to Dynamical Recurrent Neural Networks / Ed. by Kremer S., Kolen J. Wiley-IEEE Press, 2001. P. 237–243. DOI: 10.1109/9780470544037.ch14.
29. Reiss A., Stricker D. Introducing a New Benchmarked Dataset for Activity Monitoring. 16th International Symposium on Wearable Computers, ISWC 2012, Newcastle, United Kingdom, June 18–22, 2012. IEEE Computer Society, 2012. P. 108–109. DOI: 10.1109/ISWC.2012.13.
30. Candanedo L., Feldheim V., Deramaix D. Data driven prediction models of energy use of appliances in a low-energy house. Energy and Buildings. 2017. Vol. 140. P. 81–97. DOI: 10.1016/j.enbuild.2017.01.083.
31. Individual household electric power consumption data set. UCI Machine Learning Repository. URL: <https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption> (accessed: 03.09.2021).
32. Moritz S., Bartz-Beielstein T. imputeTS: Time Series Missing Value Imputation in R. R Journal. 2017. Vol. 9, no. 1. P. 207. DOI: 10.32614/rj-2017-009.

# АЛГОРИТМ СОЕДИНЕНИЯ ЦИКЛОВ ДЛЯ МЕТРИЧЕСКОЙ ЗАДАЧИ КОММИВОЯЖЕРА НА МАКСИМУМ

© 2021 А.В. Панюков, Ю.Ф. Леонова

Южно-Уральский государственный университет

(454080 Челябинск, пр. им. В.И. Ленина, д. 76)

E-mail: paniukovav@susu.ac.ru, yuliya.igosheva@gmail.com

Поступила в редакцию: 22.03.2021

Задача коммивояжера на максимум имеет ряд практических приложений, например, при сжатии произвольных данных и анализе последовательностей ДНК. При том, что задача коммивояжера на максимум является менее разработанной, чем задача коммивояжера на минимум, для ее решения существуют эффективные приближенные алгоритмы. В статье приведены оценки точности лучших на сегодняшний день алгоритмов для приближенного решения метрической задачи коммивояжера на максимум, и предлагается еще один алгоритм приближенного решения задачи коммивояжера на максимум, состоящий из поиска 2-фактора максимального веса в заданном графе, а затем применения операции оптимального соединения циклов в один гамильтонов цикл. Приведено доказательство, что для метрической задачи коммивояжера на максимум отношение длины найденного алгоритмом гамильтонова цикла к максимально возможной длине гамильтонова цикла не менее  $5/6$ . Вычислительная сложность алгоритма не превышает  $O(|V|^3)$ . Проведено тестирование качества алгоритма на случайно сгенерированных матрицах стоимостей с евклидовой метрикой. Аналитическое и численное исследование алгоритма объединения циклов позволило выдвинуть гипотезу об асимптотической точности алгоритма на классе метрических задач коммивояжера на максимум.

*Ключевые слова:* алгоритм, асимптотическая точность, вычислительная сложность, вычислительный эксперимент, задача коммивояжера.

## ОБРАЗЕЦ ЦИТИРОВАНИЯ

Панюков А.В., Леонова Ю.Ф. Алгоритм соединения циклов для метрической задачи коммивояжера на максимум // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 4. С. 26–36. DOI: 10.14529/cmse210402.

## Введение

Пусть  $G = (V, E)$  полный граф,  $W: E \rightarrow \mathbb{Z}^{\geq 0}$  — заданная весовая функция,  $W(E') = \sum_{e \in E'} W(e)$  для любого  $E' \subset E$ . Гамильтонов цикл — это цикл, проходящий через каждую вершину графа ровно один раз. Задача коммивояжера на максимум (MAX TSP) состоит в нахождении гамильтонова цикла  $H \in G$  с максимальным весом  $W(H)$ .

Задача коммивояжера на максимум нашла применение в комбинаторике и вычислительной биологии: в частности, для понимания взаимодействий РНК [1] и обеспечения алгоритмов сжатия результатов секвенирования ДНК [2]; в задаче нахождения треугольного покрытия графа максимального веса [3] и к комбинаторной задаче bandpass-2 [4], в которой необходимо найти наилучшую перестановку строк в матрице с булевыми значениями, при которой взвешенная сумма структур, называемых полосовыми пропусками, максимальна.

Особый интерес представляют геометрические экземпляры задачи коммивояжера, в которых вершины графа  $G$  соответствуют точкам в  $\mathbb{R}^d$  для некоторого  $d \geq 1$ , а расстояния вычисляются в соответствии с некоторой геометрической нормой. Подобные задачи

принято называть метрическими [5]. Метрическая задача коммивояжера на максимум является SNP-трудной [6, 21], что обуславливает актуальность изучения эвристических алгоритмов решения задачи.

В данной статье предлагается алгоритм соединения циклов для метрической задачи коммивояжера на максимум. Статья организована следующим образом. В первом разделе статьи отражено текущее состояние исследований приближенных алгоритмов решения задачи MAX TSP. Во втором разделе дано описание алгоритма соединения циклов для решения задачи коммивояжера на максимум. В третьем разделе приведено исследование качественных характеристик алгоритма. Результаты эмпирического исследования на основе разработанного программного обеспечения приведены в разделе 4. В заключении приводится краткая сводка результатов, полученных в работе, и указаны направления дальнейших исследований.

## 1. Обзор состояния проблемы

Алгоритм называется  $C$ -приближенным, если при любых исходных данных он находит допустимое решение со значением целевой функции, отличающимся от оптимума не более чем в  $C$  раз.

Первые приближенные алгоритмы для задачи MAX TSP были разработаны Фишером, Немхаузером и Уолси [7]. Они предложили несколько алгоритмов с коэффициентом аппроксимации  $1/2$  и один  $2/3$ -приближенный алгоритм. В [8] Косараджу, Парк и Штейн представили улучшенный  $19/27$ -приближенный алгоритм. Этот результат, в свою очередь, был улучшен Хассином и Рубинштейном, которые предложили  $5/7$ -приближенный алгоритм [9]. В 1984 году Сердюков [10] представил простой и элегантный  $3/4$ -приближенный алгоритм. Алгоритм является детерминированным (выдает уникальный и предопределенный результат для заданных входных данных) и имеет вычислительную сложность  $O(V^3)$ , где  $V$  — множество вершин графа. Впоследствии Хассин и Рубинштейн [12] предложили рандомизированный (использует степень случайности как часть своей логики) алгоритм, имеющий ожидаемый коэффициент аппроксимации не менее  $\frac{25(1-\varepsilon)}{33-32\varepsilon}$  и вычислительную сложность  $o(|V|^2(|V| + 2^{1/\varepsilon}))$ , где  $\varepsilon$  — произвольно малая константа. Первый детерминированный приближенный алгоритм с оценкой лучше, чем  $3/4$ , был предложен в [13] Ченом, Окамото и Вангом:  $61/81$ -приближенный алгоритм с вычислительной сложностью  $O(|V|^3)$ . В 2009 году Палуч, Муха и Мадри [14] предложили  $7/9$ -приближенный алгоритм с вычислительной сложностью  $O(|V|^3)$ . В 2017 году Дудич, Марцинковский, Палуч и Рыбицкий предложили  $4/5$ -приближенный алгоритм [15] с вычислительной сложностью  $O(|V|^3)$ .

Для метрической задачи коммивояжера на максимум известны детерминированный  $5/6$ -приближенный алгоритм Косточки—Сердюкова [16] с вычислительной сложностью  $O(|V|^4)$ ,  $7/9$ -приближенный алгоритм Панюкова—Тычинина [11] с вычислительной сложностью  $O(|V|^3)$ ,  $7/8$ -приближенный алгоритм Хассина и Рубинштейна [17], имеющий асимптотическую оценку точности и вычислительную сложность  $O(|V|^3)$ ,  $17/20$ -приближенный рандомизированный алгоритм, предложенный Ченом, Окамото и Вангом [13] с вычислительной сложностью  $O(|V|^3)$ .

## 2. Описание алгоритма соединения циклов

Суть предлагаемого алгоритма соединения циклов (Cycles Merging Algorithm — СМА) для решения задачи коммивояжера состоит в предварительном нахождении в заданном графе 2-регулярного суграфа максимального веса, т.е. покрытия этого графа циклами. Данную конструкцию принято называть 2-фактором максимального веса, он определяется на первом шаге алгоритма.

На втором шаге проверяется условие единственности цикла полученного решения. Если 2-фактор представлен единственным циклом, то этот цикл является решением задачи. Если 2-фактор представлен несколькими циклами, то перебираются попарно различные циклы  $r$  и  $t$ , в каждом цикле выбранной пары перебираются по одному ребру, пусть выбраны  $e_{\{r \oplus t\}}^r = [v_1, v_2]$  и  $e_{\{r \oplus t\}}^t = [u_1, u_2]$ , для них находятся две пары сопряженных ребер ( $f = [v_1, u_1]$ ,  $g = [v_2, u_2]$ ) и ( $f = [v_1, u_2]$ ,  $g = [v_2, u_1]$ ), используемых для соединения циклов (см. рис. 1). Выполняется поиск набора  $\{r, t, e_{\{r \oplus t\}}^r, e_{\{r \oplus t\}}^t, f, g\}$  из перечисленных элементов такой, что соединенный цикл имеет максимальный вес.

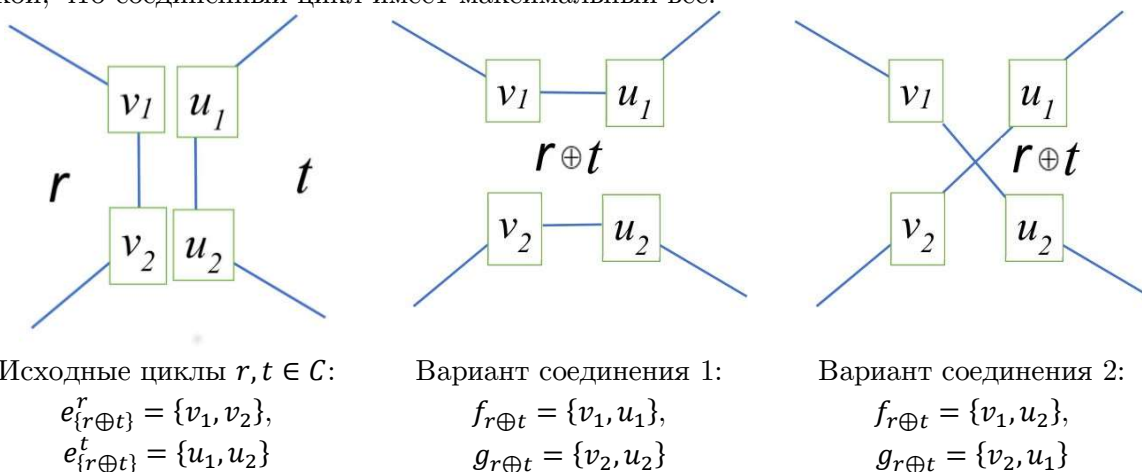


Рис. 1. Варианты соединения циклов

Затем найденная пара циклов заменяется объединенным циклом с максимальной стоимостью. Алгоритм завершает работу, когда текущий 2-фактор содержит один цикл.

Формально СМА для задачи MAX TSP может быть записан следующим образом.

### Алгоритм соединения циклов

**Вход:** полный граф  $G = (V, E)$ ,

весовая функция  $W: E \rightarrow R$ .

**Выход:** гамильтонов цикл максимального веса.

**Шаг 1.** Найти 2-фактор  $C = \{c_1, c_2, \dots, c_n\}$  максимального веса в графе  $G$ .

**Шаг 2.** Если  $n = 1$ , то вернуть  $C$  — найденное решение. Переход на Шаг 5.

**Шаг 3.** Для всех попарно различных циклов  $r, t \in C$  найти замену ребер, обеспечивающую оптимальное соединение циклов, и стоимость этого соединения.

$$\begin{pmatrix} e_{\{r \oplus t\}}^r \\ e_{\{r \oplus t\}}^t \\ f_{\{r \oplus t\}} \\ g_{\{r \oplus t\}} \end{pmatrix} = \arg \max_{\substack{e^r = \{u_1, u_2\} \in E(c_r) \\ e^t = \{u_1, u_2\} \in E(c_t) \\ f, g \in E(G): f \cup g = e^r \cup e^t}} [W(f) + W(g) - W(e^r) - W(e^t)], \quad (1)$$

$$\tilde{W}(r \oplus t) = W(f_{r \oplus t}) + W(g_{r \oplus t}) - W(e_{r \oplus t}^r) - W(e_{r \oplus t}^t). \quad (2)$$

**Шаг 4.** Пока  $n > 1$ , выполнять шаги 4.1–4.4.

**Шаг 4.1.** Пусть

$$(r^*, t^*) = \arg \max_{r, t \in C} \tilde{W}(r \oplus t). \quad (3)$$

**Шаг 4.2.** Построить цикл

$$\begin{aligned} s &= r^* \oplus t^*: V(s) = V(r^*) \cup V(t^*), \\ E(s) &= (E(r^*) \cup E(t^*) \cup \{f_{r \oplus t}, g_{r \oplus t}\}) \setminus \{e_{\{r \oplus t\}}^r, e_{\{r \oplus t\}}^t\}. \end{aligned} \quad (4)$$

**Шаг 4.3.** Модифицировать 2-фактор

$$C = (C \cup \{s\}) \setminus \{r, t\}; n := n - 1. \quad (5)$$

**Шаг 4.4.** Для всех  $t \in C: t \neq s$  найти замену ребер, обеспечивающую оптимальное соединение циклов  $t$  и  $s$ , и стоимость этого соединения:

$$\begin{pmatrix} e_{\{r \oplus t\}}^r \\ e_{\{r \oplus t\}}^t \\ f_{\{r \oplus t\}} \\ g_{\{r \oplus t\}} \end{pmatrix} = \arg \max_{\substack{e^r = \{u_1, u_2\} \in E(c_r) \\ e^t = \{u_1, u_2\} \in E(c_t) \\ f, g \in E(G): f \cup g = e^r \cup e^t}} [W(f) + W(g) - W(e^r) - W(e^t)], \quad (6)$$

$$\tilde{W}(r \oplus t) = W(f_{s \oplus t}) + W(g_{s \oplus t}) - W(e_{s \oplus t}^s) - W(e_{s \oplus t}^t). \quad (7)$$

**Шаг 5.**  $C$  — найденное решение. Останов.

### 3. Аналитическое исследование качества алгоритма соединения циклов для метрической задачи MAX TSP

#### 3.1. Вычислительная сложность алгоритма соединения циклов

**Теорема 1.** Вычислительная сложность алгоритма СМА соединения циклов не превосходит величины  $O(|V|^3)$ .

*Доказательство.* Вычислительная сложность Шага 1 не превосходит величины  $O(|V|^3)$ . Вычислительная сложность Шага 2 не превосходит величины  $O(1)$ .

Поскольку 2-фактор неориентированного графа является объединением не более  $O(|V|/3)$  циклов и содержит  $|V|$  ребер, то количество возможных замен ребер для соединения циклов не превосходит величины  $O(|V|^2)$ , а вычислительная сложность нахождения всех оптимальных замен, т.е. вычислительная сложность шага 3, также не превосходит величины  $O(|V|^3)$ .

Цикл Шага 4 выполняется не более  $O(|V|/3)$  раз. Наиболее трудоемкими в теле цикла являются Шаг 4.1 и Шаг 4.4. Очевидно, что их вычислительная сложность не превосходит величины  $O(|V|)$ . Следовательно, вычислительная сложность Шага 4 не превосходит величины  $O(|V|^2)$ , а всего алгоритма соединения циклов не превосходит величины  $O(|V|^3)$ . Теорема доказана.

#### 3.2. Точность алгоритма соединения циклов для метрической задачи MAX TSP

**Теорема 2.** Пусть  $W_{opt}$  — оптимальное значение метрической задачи MAX TSP,  $W_C$  — вес максимального 2-фактора данной задачи,  $W_{alg}$  — вес цикла, построенного алгоритмом СМА. Тогда  $W_{alg}/W_{opt} \geq 5/6$ . Оценка  $W_{alg}/W_{opt} \geq W_{alg}/W_C = 5/6$  достижима.

*Доказательство.* Так как весовая функция  $W$  метрической задачи удовлетворяет неравенству треугольника

$$W\{u, v\} + W\{v, w\} \leq W\{u, w\}, \quad u, v, w \in V, \quad (8)$$

то имеем:

$$\begin{aligned} W\{v_1, v_2\} &\leq W\{v_1, u_1\} + W\{u_1, v_2\}, \\ W\{v_1, v_2\} &\leq W\{v_1, u_2\} + W\{u_2, v_2\}, \\ W\{u_1, u_2\} &\leq W\{u_1, v_1\} + W\{v_1, u_2\}, \\ W\{u_1, u_2\} &\leq W\{u_1, v_2\} + W\{v_2, u_2\}, \\ v_1, v_2, u_1, u_2 &\in V. \end{aligned}$$

Суммирование этих неравенств дает:

$$\begin{aligned} 2 \cdot (W\{v_1, v_2\} + W\{u_1, u_2\}) &\leq 2 \cdot W\{v_1, u_1\} + 2 \cdot W\{u_2, v_2\} + 2 \cdot W\{v_2, u_1\} + 2 \cdot W\{u_1, v_2\} \leq \\ &\leq 4 \cdot \max\{W\{v_1, u_1\} + W\{u_2, v_2\}, W\{v_2, u_1\} + W\{u_1, v_2\}\}, \quad v_1, v_2, u_1, u_2 \in V. \end{aligned} \quad (9)$$

Пусть  $C$  — 2-фактор максимального веса. Рассмотрим соединение циклов  $r, t \in C$  с использованием ребер  $\{v_1, v_2\} \in E(r)$ ,  $\{u_1, u_2\} \in E(t)$  (рис. 2).

Из оптимальности 2-фактора  $C$  следует, что

$$W(e_{\{r \oplus t\}}^r) + W(e_{\{r \oplus t\}}^t) = W\{v_1, v_2\} + W\{u_1, u_2\} \geq W\{v_1, u_1\} + W\{u_2, v_2\}, \quad (10)$$

$$W(e_{\{r \oplus t\}}^r) + W(e_{\{r \oplus t\}}^t) = W\{v_1, v_2\} + W\{u_1, u_2\} \geq W\{v_1, u_2\} + W\{v_2, u_1\}. \quad (11)$$

Следовательно, с учетом (9) имеем

$$\begin{aligned} \frac{[W(e_{\{r \oplus t\}}^r) + W(e_{\{r \oplus t\}}^t)]}{2} &\leq \\ &\leq \max\{W\{v_1, u_1\} + W\{u_2, v_2\}, W\{v_2, u_1\} + W\{u_1, v_2\}\} \leq [W(f_{\{r \oplus t\}}) + W(g_{\{r \oplus t\}})], \end{aligned} \quad (12)$$

то есть полусумма веса удаленных ребер не превосходит суммы весов введенных ребер.

Учитывая (12), получаем нижнюю оценку стоимости соединенных циклов  $r, t \in C$ :

$$\begin{aligned} \tilde{W}(r \oplus t) &= W(f_{\{r \oplus t\}}) + W(g_{\{r \oplus t\}}) - W(e_{\{r \oplus t\}}^r) - W(e_{\{r \oplus t\}}^t) \geq \\ &\quad - \min_{\substack{\{v_1, v_2\} \in E(r), \\ \{u_1, u_2\} \in E(t)}} \frac{W\{v_1, v_2\} + W\{u_1, u_2\}}{2} = \\ &= -\frac{1}{2} \left[ \min_{\{v_1, v_2\} \in E(r)} W\{v_1, v_2\} + \min_{\{u_1, u_2\} \in E(t)} W\{u_1, u_2\} \right] \geq -\frac{1}{2} \cdot \left[ \frac{W(r)}{|r|} + \frac{W(t)}{|t|} \right]. \end{aligned} \quad (13)$$

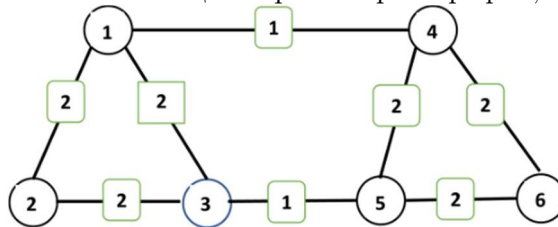
В данной цепочке первое неравенство есть следствие (12), равенство есть следствие отсутствия пересечения  $r, t \in C$ , последнее неравенство следует из того, что средний вес ребра в цикле не меньше минимального из весов входящих в него ребер. Следовательно,

$$\begin{aligned} \frac{W_{alg}}{W_{opt}} &\geq \frac{W_{alg}}{W(C)} \geq \frac{W(C) - \sum_{c \in C} \frac{W(c)}{2 \cdot |c|}}{W(C)} = 1 - \frac{1}{W(C)} \cdot \sum_{c \in C} \frac{W(c)}{2|c|} \geq \\ &\geq 1 - \frac{1}{2 \cdot W(C)} \cdot \max \left\{ \sum_{i=1}^{|C|} \frac{y_i}{x_i} \left| \begin{array}{l} \sum_{i=1}^{|C|} y_i = W(C), \\ \sum_{i=1}^{|C|} x_i = |V|, \\ x, y \geq 0 \end{array} \right. \right\} = 1 - \frac{|C|}{2 \cdot |V|} \geq \frac{5}{6}. \end{aligned} \quad (14)$$

Первое неравенство является следствием  $W_{opt} \leq W(C)$ . Второе неравенство является следствием (1)–(5) и (13). В третьем неравенстве сумма средних значений весов ребер из элементов 2-фактора  $C$  заменена численной задачей на максимум с допустимым множеством, включающим средние значения весов ребер из каждого элемента 2-фактора  $C$ .

Оптимальное решение полученной задачи равно  $x_i = |V|/|C|$ ,  $y_i = W(C)/|C|$ ,  $i = 1, 2, \dots, |C|$ , оптимальное значение равно  $(W(C) \cdot |C|)/|V|$ . Равенство очевидно. Четвертое неравенство является следствием неравенства  $|C| \leq |V|/3$ .

Для доказательства достижимости оценки рассмотрим граф  $G$ , приведенный на рис. 2.



**Рис. 2.** Пример графа для доказательства достижимости оценки  $W_{alg}/W_C = 5/6$

Чтобы не загромождать рисунок, изображены только ребра с ненулевым весом. Оптимальным решением задачи MAX TSP для данного графа является гамильтонов цикл  $H = \{1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 6 \rightarrow 4 \rightarrow 1\}$ . Его вес  $W_{opt} = 2 + 2 + 1 + 2 + 2 + 1 = 10$ . 2-фактор максимального веса содержит циклы  $C_1 = \{1 \rightarrow 2 \rightarrow 3 \rightarrow 1\}$  и  $C_2 = \{4 \rightarrow 5 \rightarrow 6 \rightarrow 4\}$ . Его вес  $W(C) = 2 + 2 + 2 + 2 + 2 + 2 = 12$ . В результате объединения циклов  $C_1$  и  $C_2$  имеем  $C_1 \oplus C_2 = H$ . Таким образом, в рассматриваемом примере  $W_{alg}/W_{opt} > W_{alg}/W_C = 5/6$ .

Теорема доказана.

#### 4. Эмпирическое исследование алгоритма соединения циклов для метрической задачи коммивояжера на максимум

Для проверки полученных теоретических результатов алгоритм соединения циклов для решения задачи коммивояжера на максимум был протестирован на случайно сгенерированных матрицах стоимостей с евклидовой метрикой. Количество вершин  $n$  варьируется от 100 до 3000 с шагом 100. Для  $100 < n < 1000$  все результаты являются средними по 10 испытаниям каждое, а для  $1000 < n < 3000$  результаты являются средними по трем испытаниям каждое.

Такая схема тестирования применяется в известных работах [18, 19] для анализа качества работы алгоритмов решения задачи коммивояжера.

Так как уже для большинства задач с числом городов  $n > 13$  невозможно получить точное решение за приемлемое время [20], то в качестве оценки относительной погрешности использовано отношение разности верхней оценки и полученного значения к полученному значению. В качестве верхней оценки принимается сумма весов ребер, полученная на первой итерации цикла. Даже при такой грубой оценке результаты эксперимента (см. табл. 1) показывают, что при небольшом числе вершин среднее значение оценки относительной погрешности составляет менее 0.1 доли процента. С ростом числа вершин точность алгоритма быстро растет.

Полученные эмпирические результаты для метрической MAX TSP значительно превосходят теоретическую оценку точности. Проведенное исследование дает основание выдвинуть гипотезу об асимптотической точности алгоритма СМА на классе метрических задач коммивояжера на максимум.



**Таблица 1.** Данные об эмпирической оценке относительной погрешности

Число вершин	Оценка точности $\frac{W_{alg}}{W_{opt}}$	Число вершин	Оценка точности $\frac{W_{alg}}{W_{opt}}$
<b>100</b>	0.9990217	<b>1600</b>	0.9999727
<b>200</b>	0.9995204	<b>1700</b>	0.9999747
<b>300</b>	0.9997492	<b>1800</b>	0.9999738
<b>400</b>	0.9997926	<b>1900</b>	0.9999752
<b>500</b>	0.9998432	<b>2000</b>	0.9999790
<b>600</b>	0.9998711	<b>2100</b>	0.9999810
<b>700</b>	0.9998988	<b>2200</b>	0.9999801
<b>800</b>	0.9999048	<b>2300</b>	0.9999833
<b>900</b>	0.9999343	<b>2400</b>	0.9999809
<b>1000</b>	0.9999340	<b>2500</b>	0.9999830
<b>1100</b>	0.9999506	<b>2600</b>	0.9999867
<b>1200</b>	0.9999594	<b>2700</b>	0.9999846
<b>1300</b>	0.9999593	<b>2800</b>	0.9999855
<b>1400</b>	0.9999691	<b>2900</b>	0.9999860
<b>1500</b>	0.9999753	<b>3000</b>	0.9999862

## Заключение

Предложенный в данной работе алгоритм соединения циклов для метрической задачи коммивояжера на максимум, заключается в нахождении 2-фактора максимального веса и последующем соединении входящих в него циклов в гамильтонов цикл. Проведенное аналитическое и эмпирическое исследование алгоритма соединения циклов для приближенного решения задачи коммивояжера на максимум подтверждают справедливость доказанных в работе теорем.

Предложенный алгоритм имеет вычислительную сложность  $O(|V|^3)$ , как в алгоритме Панюкова—Тычина, однако, его точность  $W_{alg}/W_{opt} \geq 5/6$  аналогична точности алгоритма Косточки—Сердюкова, лучшего из известных на сегодняшний день детерминированных приближенных алгоритмов для метрической задачи MAX TSP, который имеет сложность  $O(|V|^4)$ . Таким образом, предложенный алгоритм имеет более высокие качественные характеристики.

Проведенное исследование дает основание выдвинуть гипотезу об асимптотической точности алгоритма соединения циклов на классе метрических задач MAX TSP. Полное статистическое исследование данной гипотезы представляет предмет отдельной статьи.

## Литература

1. Tong W., Goebel R., Liu T., Lin G. Approximation algorithms for the maximum multiple RNA interaction problem // International Conference on Combinatorial Optimization and Applications, Chengdu, China, December 12–14, 2013. Springer, Cham, 2013. P. 49–59. DOI: 10.1007/978-3-319-03780-6\_5.

2. Sichen Z., Zhao L., Liang Y. *et al.* Optimizing read reversals for sequence compression // International Workshop on Algorithms in Bioinformatics, Atlanta, GA, USA, September 10–12, 2015. Springer, Berlin, Heidelberg, 2015. P. 189–202. DOI: 10.1007/978-3-662-48221-6\_14.
3. Hassin R., Rubinstein S. An approximation algorithm for maximum triangle packing // *Discrete applied mathematics*. 2006. Vol. 154, no. 6. P. 971–979. DOI: 10.1016/j.dam.2005.11.003.
4. Chen Z.Z., Wang L. An improved approximation algorithm for the bandpass-2 problem // International Conference on Combinatorial Optimization and Applications, Banff, AB, Canada, August 5–9, 2012. Springer, Berlin, Heidelberg, 2012. P. 188–199. DOI: 10.1007/978-3-642-31770-5\_17.
5. Харари Ф. Теория графов. Глава 9. Факторизация. М.: Едиториал УРСС, 2003. 296 с.
6. Barvinok A.I., Johnson D.S., Woeginger G.J., Woodroffe R. Finding maximum length tours under polyhedral norms // *Proceedings of IPCO VI, Lecture Notes in Computer Science*, Houston, Texas, June 22–24, 1998. 1998. Vol. 1412. P. 195–201. DOI: 10.1007/3-540-69346-7\_15.
7. Fisher M.L., Nemhauser G.L., Wolsey L.A. An analysis of approximations for finding a maximum weight Hamiltonian circuit // *Operations Research*. 1979. Vol. 27, no. 4. P. 799–809. DOI: 10.1287/opre.27.4.799.
8. Kosaraju S.R., Park J.K., Stein C. Long tours and short superstrings // *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Santa Fe, New Mexico, November 20–22, 1994. IEEE, 1994. P. 166–177. DOI: 10.1109/sfcs.1994.365696.
9. Hassin R., Rubinstein S. An approximation algorithm for the maximum traveling salesman problem // *Information Processing Letters*. 1998. Vol. 67, no. 3. P. 125–130. DOI: 10.1016/s0020-0190(98)00102-1.
10. Сердюков А.И. Алгоритм с оценкой для задачи коммивояжера на максимум // *Дискретный анализ и исследование операций*. 1984. № 25. С. 80–86.
11. Панюков А.В., Тычинин С.А. Применение дополнений паросочетаниями для решения задачи MAX TSP // *Вестник Южно-Уральского государственного университета. Серия: Математическое моделирование и программирование*. 2008. № 15 (115). С. 54–63.
12. Hassin R., Rubinstein S. Better approximations for MAX TSP // *Information Processing Letters*. 2000. Vol. 75, no. 4. P. 181–186. DOI: 10.1016/s0020-0190(00)00097-1.
13. Chen Z.Z., Okamoto Y., Wang L. Improved deterministic approximation algorithms for MAX TSP // *Information Processing Letters*. 2005. Vol. 95, no. 2. P. 333–342. DOI: 10.1016/j.ipl.2005.03.011.
14. Paluch K., Mucha M., Madry A. A 7/9-approximation algorithm for the maximum traveling salesman problem // *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, Berkeley, CA, USA, August 21–23, 2009. Springer, Berlin, Heidelberg, 2009. P. 298–311. DOI: 10.1007/978-3-642-03685-9\_23.
15. Dudycz S., Marcinkowski J., Paluch K., Rybicki B. A 4/5-approximation algorithm for the maximum traveling salesman problem // *International Conference on Integer Programming and Combinatorial Optimization*, Waterloo, ON, Canada, June 26–28, 2017. Springer, Cham, 2017. P. 173–185. DOI: 10.1007/978-3-319-59250-3\_15.
16. Косточка А.В., Сердюков А.И. Полиномиальные алгоритмы с оценками 3/4 и 5/6 для задачи коммивояжера на максимум // *Управляемые системы*. 1985. № 26. С. 55–59

17. Hassin R., Rubinstein S. A 7/8-approximation algorithm for metric MAX TSP // Information processing letters. 2002. Vol. 81, no. 5. P. 247–251. DOI: 10.1016/s0020-0190(01)00234-4.
18. Glover F., Gutin G., Yeo A., Zverovich A. Construction heuristics for the asymmetric TSP // European Journal of Operational Research. 2001. Vol. 129, no. 3. P. 555–568. DOI: 10.1016/s0377-2217(99)00468-3.
19. Goldengorin B., Jäger G., Molitor P. Tolerance based contract-or-patch heuristic for the asymmetric TSP // Workshop on Combinatorial and Algorithmic Aspects of Networking, Chester, UK, July 2, 2006. Springer, Berlin, Heidelberg, 2006. P. 86–97. DOI: 10.1007/11922377\_8.
20. Кузюрин Н.Н., Фомин С.А. Эффективные алгоритмы и сложность вычислений : учебное пособие. Москва: МФТИ, 2007. 311 с.

Панюков Анатолий Васильевич, д.ф.-м.н., профессор, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Леонова Юлия Федоровна, аспирант, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

---

DOI: 10.14529/cmse210402

## CYCLES MERGING ALGORITHM FOR METRIC MAXIMUM TRAVELING SALESMAN PROBLEM

© 2021 A.V. Panyukov, Yu.F. Leonova

*South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)*

*E-mail: paniukovav@susu.ac.ru, yuliya.igosheva@gmail.com*

Received: 22.03.2021

The traveling salesman problem is an important combinatorial optimization problem that involves finding the optimal path between given vertices. The maximum traveling salesman problem has several practical applications, for example, when compressing arbitrary data and analyzing DNA sequences. Even though maximum traveling salesman problem is less developed than minimum traveling salesman problem, there are effective approximate algorithms for solving this problem. The article presents estimates of the accuracy of the best algorithms for the approximate solution of the metric maximum traveling salesman problem. The paper proposes a new algorithm for the approximate solution of the traveling salesman problem to the maximum, consisting of finding the 2-factor of the extreme weight in each graph, and then applying the operation of the optimal connection of cycles into one Hamiltonian cycle. The computational complexity of the algorithm does not exceed  $O(|V|^3)$ . We present a proof of the theorem that for the metric traveling salesman problem, the maximum accuracy of the algorithm is at least 5/6. The quality of the algorithm has been tested on randomly generated cost matrices with the Euclidean metric. An analytical and numerical study of the algorithm for combining cycles has allowed us to move the hypothesis about the asymptotic accuracy of the algorithm on the class of metric traveling salesman problems to the maximum.

*Keywords: algorithm, asymptotic accuracy, computational complexity, computational experiment, traveling salesman problem.*

### FOR CITATION

Panyukov A.V., Leonova Yu.F. Cycles Merging Algorithm for Metric Maximum Traveling Salesman Problem. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2021. Vol. 10, no. 4. P. 26–36. (in Russian) DOI: 10.14529/cmse210402.

---

*This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.*

## References

1. Tong W., Goebel R., Liu T., Lin G. Approximation algorithms for the maximum multiple RNA interaction problem. International Conference on Combinatorial Optimization and Applications, Chengdu, China, December 12–14, 2013. Springer, Cham, 2013. P. 49–59. DOI: 10.1007/978-3-319-03780-6\_5.
2. Sichen Z., Zhao L., Liang Y. *et al.* Optimizing read reversals for sequence compression. International Workshop on Algorithms in Bioinformatics, Atlanta, GA, USA, September 10–12, 2015. Springer, Berlin, Heidelberg, 2015. P. 189–202. DOI: 10.1007/978-3-662-48221-6\_14.
3. Hassin R., Rubinstein S. An approximation algorithm for maximum triangle packing. Discrete applied mathematics. 2006. Vol. 154, no. 6. P. 971–979. DOI: 10.1016/j.dam.2005.11.003.
4. Chen Z.Z., Wang L. An improved approximation algorithm for the bandpass-2 problem. International Conference on Combinatorial Optimization and Applications, Banff, AB, Canada, August 5–9, 2012. Springer, Berlin, Heidelberg, 2012. P. 188–199. DOI: 10.1007/978-3-642-31770-5\_17.
5. Harari F. Graph Theory. Chapter 9. Factorization. Moscow: Editorial URSS, 2003. 296 p. (in Russian)
6. Barvinok A.I., Johnson D.S., Woeginger G.J., Woodroffe R. Finding maximum length tours under polyhedral norms. Proceedings of IPCO VI, Lecture Notes in Computer Science, Houston, Texas, June 22–24, 1998. 1998. Vol. 1412. P. 195–201. DOI: 10.1007/3-540-69346-7\_15.
7. Fisher M.L., Nemhauser G.L., Wolsey L.A. An analysis of approximations for finding a maximum weight Hamiltonian circuit. Operations Research. 1979. Vol. 27, no. 4. P. 799–809. DOI: 10.1287/opre.27.4.799.
8. Kosaraju S.R., Park J.K., Stein C. Long tours and short superstrings. Proceedings 35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, November 20–22, 1994. IEEE, 1994. P. 166–177. DOI: 10.1109/sfcs.1994.365696.
9. Hassin R., Rubinstein S. An approximation algorithm for the maximum traveling salesman problem. Information Processing Letters. 1998. Vol. 67, no. 3. P. 125–130. DOI: 10.1016/s0020-0190(98)00102-1.
10. Serdyukov A.I. An algorithm with an estimate for the traveling salesman problem for a maximum. Discrete analysis and operations research. 1984. No. 25. P. 80–86. (in Russian)
11. Panyukov A.V., Tychinin S.A. The use of matching additions for solving MAX TSP. Bulletin of South Ural State University. Series: Mathematical modeling and programming. 2008. No. 15(115). P. 54–63. (in Russian)
12. Hassin R., Rubinstein S. Better approximations for MAX TSP. Information Processing Letters. 2000. Vol. 75, no. 4. P. 181–186. DOI: 10.1016/s0020-0190(00)00097-1.
13. Chen Z.Z., Okamoto Y., Wang L. Improved deterministic approximation algorithms for MAX TSP. Information Processing Letters. 2005. Vol. 95, no. 2. P. 333–342. DOI: 10.1016/j.ipl.2005.03.011.

14. Paluch K., Mucha M., Madry A. A 7/9-approximation algorithm for the maximum traveling salesman problem. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, Berkeley, CA, USA, August 21–23, 2009. Springer, Berlin, Heidelberg, 2009. P. 298–311. DOI: 10.1007/978-3-642-03685-9\_23.
15. Dudycz S., Marcinkowski J., Paluch K., Rybicki B. A 4/5-approximation algorithm for the maximum traveling salesman problem. *International Conference on Integer Programming and Combinatorial Optimization*, Waterloo, ON, Canada, June 26–28, 2017. Springer, Cham, 2017. P. 173–185. DOI: 10.1007/978-3-319-59250-3\_15.
16. Kostochka A.V., Serdyukov A.I. Polynomial algorithms with the estimates  $3/4$  and  $5/6$  for the Traveling Salesman Problem of the maximum. *Managed systems*. 1985. No. 26. P. 55–59. (in Russian)
17. Hassin R., Rubinstein S. A 7/8-approximation algorithm for metric MAX TSP. *Information processing letters*. 2002. Vol. 81, no. 5. P. 247–251. DOI: 10.1016/s0020-0190(01)00234-4.
18. Glover F., Gutin G., Yeo A., Zverovich A. Construction heuristics for the asymmetric TSP. *European Journal of Operational Research*. 2001. Vol. 129, no. 3. P. 555–568. DOI: 10.1016/s0377-2217(99)00468-3.
19. Goldengorin B., Jäger G., Molitor P. Tolerance based contract-or-patch heuristic for the asymmetric TSP. *Workshop on Combinatorial and Algorithmic Aspects of Networking*, Chester, UK, July 2, 2006. Springer, Berlin, Heidelberg, 2006. P. 86–97. DOI: 10.1007/11922377\_8.
20. Kuzyurin N.N., Fomin S.A. *Efficient algorithms and computational complexity: a tutorial*. Moscow: MIPT, 2007. 311 p. (in Russian)

## ОРГАНИЗАЦИЯ ОБМЕНА ДАННЫМИ В РАМКАХ ПЛАТФОРМЫ МОБИЛЬНОЙ МЕДИЦИНЫ

© 2021 И.А. Волков<sup>1</sup>, Г.И. Радченко<sup>1</sup>, А.Н. Черных<sup>1,2</sup>

<sup>1</sup>Южно-Уральский государственный университет  
(454080 Челябинск, пр. им. В.И. Ленина, д. 76),

<sup>2</sup>Научно-исследовательский центр Энсенады  
(22860 Энсенада, Мексика, Carretera Ensenada – Tijuana No. 3918)  
E-mail: volkovia@susu.ru, gleb.radchenko@susu.ru, chernykh@cicese.mx

Поступила в редакцию: 03.11.2021

Мобильная медицина является важной составляющей для реализации постоянного мониторинга здоровья, который открывает для врачей, пациентов и исследователей новые возможности. Экспоненциальный рост сферы привел к увеличению скорости появления новых решений на рынке здравоохранения и, соответственно, к увеличению количества информации о здоровье населения. Однако сегодня практически вся собираемая сервисами мобильной медицины информация изолирована друг от друга, так как распределена по разным сайтам и мобильным приложениям. Отсутствие у пользователя возможности удобно управлять своими медицинскими данными и делиться ими является существенной проблемой на текущий момент. Создание платформы мобильной медицины позволит решить данную проблему, предлагая механизмы на основе которых сторонние разработчики могут размещать свои сервисы и организовывать меж-сервисный обмен данными пользователей. В рамках нашей работы мы провели анализ существующих решений на рынке мобильной медицины, описали варианты использования сервисов в сфере мобильной медицины. Нами предложена концепция платформы мобильной медицины и проведено сравнение способов организации обмена медицинскими данными. На основе проведенного анализа нами был выбран и реализован прототип системы обмена медицинскими данными между мобильными клиентами на основе однорангового (peer-to-peer) соединения.

*Ключевые слова:* умное здравоохранение, мобильная медицина, мониторинг здоровья, платформа, PaaS, интернет вещей, мобильные приложения.

### ОБРАЗЕЦ ЦИТИРОВАНИЯ

Волков И.А., Радченко Г.И., Черных А.Н. Организация обмена данными в рамках платформы мобильной медицины // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 4. С. 37–59. DOI: 10.14529/cmse210403.

### Введение

Современная медицина сталкивается со множеством проблем, связанных с увеличением потребности населения в использовании услуг системы здравоохранения [1]. Данные проблемы вызваны ростом числа населения; повышением средней продолжительности жизни; ростом числа пожилых людей, за которыми требуется уход и наблюдение [2]; ростом стоимости медицинских услуг, что особенно сильно сказывается на больных хроническими заболеваниями [3]; а также урбанизацией городов, что увеличивает шансы на возникновение в них эпидемий, приводящих к резким скачкам количества пациентов, которым требуется медицинская помощь [4].

Считается, что данные проблемы возможно решить, применяя современные технологии и подходы к лечению пациентов. Предоставление пациентам возможности принимать активное участие в отслеживании своего здоровья и управления им, потенциально может

помочь децентрализовать уход за здоровьем, что позволит снизить нагрузку врачей и увеличить эффективность проводимого лечения [5].

Мобильная медицина сегодня является одним из ключевых подходов, обеспечивающих решение задач в области здравоохранения. В настоящее время ведется большое количество исследований в области *мобильной медицины* (*Mobile health, mHealth*), ориентированных на методики использования мобильных технологий для постоянного мониторинга и воздействия на состояние пациента [6]. Решения мобильной медицины генерируют большое количество данных, которые могут быть использованы как самими пациентами, так и их лечащими врачами, родственниками или независимыми исследовательскими группами. Данные, генерируемые одним таким решением могут достигать объема в несколько гигабайт за один день только с одного устройства [1].

Существующие сегодня решения в области мобильной медицины не предоставляют универсальных кросс-платформенных механизмов организации обмена медицинскими данными из различных источников. Большое число приложений и сервисов мобильной медицины сегодня собирают данные в своих проприетарных форматах, без возможности универсального обмена этими данными как с другими приложениями так и с заинтересованными сторонами. Растущее количество медицинских данных и их источников вызывает необходимость в платформе, которая могла бы обеспечить удобную и безопасную обработку и обмен такими данными.

Активный рост рынка информационных технологий в сфере здравоохранения показывает, что исследования в данной сфере являются актуальными. Согласно [7], данный рынок должен был вырасти с 99.6 миллиардов долларов в 2010 году до 162.2 миллиардов долларов к 2015 году. Сейчас, возвращаясь к тому же источнику мы можем видеть, что на 2019 год данный рынок составляет 187.6 миллиардов и ожидает роста до 390.7 миллиардов долларов к 2024 году [8]. Таким образом, мы можем наблюдать экспоненциальный рост данного рынка в предыдущие десять лет, и данный тренд только сохранится в следующем десятилетии.

Авторами данной статьи реализовано и развивается решение в области мобильной медицины, обеспечивающее поддержку пациентов, больных сахарным диабетом: “DiaMeter — Ваш дневник диабета” [9, 10]. Анализ пожеланий пользователей данного решения также дает основания для того, чтобы считать задачу организации обмена и обработки медицинских данных актуальной.

Целью данной работы является исследование методов организации обмена данными в контексте мобильной медицины, с учетом требований к приватности и других ограничений, накладываемых спецификой предметной области. Для реализации цели нам необходимо решить следующие задачи:

- провести анализ существующих решений на рынке мобильной медицины, в том числе в контексте возможных механизмов обмена данными;
- выявить ключевых акторов и варианты использования механизмов обмена данными в мобильной медицине;
- провести анализ возможных вариантов организации обмена данными с учетом функциональных требований;
- спроектировать и реализовать прототип решения обмена медицинскими данными.

Дальнейший текст статьи организован следующим образом. В разделе 1 мы проводим обзор существующих решений в области мобильной медицины, рассматриваем примеры использования мобильной медицины компаниями и анализируем варианты стандартизации

решений обмена данными мобильной медицины. В разделе 2 мы даем определение платформе мобильной медицины, обсуждаем концепцию работы платформы, приводим анализ вариантов использования решений в области мобильной медицины и выводим из него варианты использования платформы мобильной медицины. В разделе 3 нами представлен анализ подходов по организации обмена данными в рамках платформы мобильной медицины. В разделе 4 мы приводим описание прототипа решения, обеспечивающего обмен медицинскими данными между мобильными клиентами на основе однорангового (peer-to-peer) соединения. В заключении представлены обобщенные результаты работы, а также перспективные направления развития данной темы.

## 1. Обзор решений в области мобильной медицины

### 1.1. Мобильная медицина

В настоящее время ведется большое количество научно-исследовательских работ в области мобильной медицины, ориентированных на методики использования мобильных технологий для постоянного мониторинга и воздействия на состояние пациента [6]. В статье [11] мы представили анализ предметной области и пришли к заключению, что мобильная медицина включает в себя не только технологии, связанные с мобильными приложениями, но и технологии интернета вещей, периферийных устройств, компьютерного зрения и телемедицины. Мобильная медицина включает в себя множество областей, среди которых выделяют [12]:

- *носимые сенсоры* — браслеты, часы, головные повязки, пластыри, наушники и одежда, обеспечивающие пассивный и постоянный мониторинг биометрических показателей человека;
- *лаборатории на чипе* (микросистемы полного анализа) — это миниатюрные приборы, позволяющие осуществлять один или несколько многостадийных (био) химических процессов на одном чипе площадью от нескольких  $mm^2$  до нескольких  $cm^2$  и использующие микро- или наноскопические количества образцов для пробоподготовки и проведения реакций;
- *интеллектуальный анализ изображений* — высокое качество камер смартфонов позволило использовать их для диагностики фотометрических показателей как с использованием дополнительных приборов (например, для распознавания ушной инфекции с помощью отоскопа), так и без них.

### 1.2. Постоянный мониторинг здоровья

*Интернет медицинских вещей (Internet of Medical Things, IoMT)* — это отдельная категория Интернета Вещей, отличительной особенностью которой является использование сенсоров с целью мониторинга и контроля здоровья пациентов [13]. Такие сенсоры позволяют собирать и обрабатывать важные биометрические данные о здоровье человека в режиме реального времени.

В статье [14] основной современной технологией, обеспечивающей рост сферы мониторинга состояния здоровья и возможность его массового внедрения выделяют *незаметные (или ненавязчивые) сенсорные и носимые устройства (Unobtrusive sensing and wearable systems)*. Наиболее часто измеряемые с помощью таких устройств жизненные показатели включают в себя: ЭКГ, баллистокardiограмма (БКГ), частота сердечных сокращений, артериальное давление (АД), насыщение крови кислородом (SpO<sub>2</sub>), уровень глюкозы в крови,



температура тела, осанка и физическая активность. Основное отличие данных медицинских устройств заключается в возможности интегрировать их в повседневную жизнь пользователя так, что данные устройства не только не будут нарушать обычный распорядок дня человека, а наоборот давать ему преимущества в виде отображения детальной информации о разных аспектах его здоровья на экране мобильного устройства. Благодаря этому появляется возможность в автономном и регулярном режиме не только собирать и хранить важные данные о здоровье человека на протяжении длительного промежутка времени, но и анализировать их, сравнивать с данными других пациентов и помогать врачам принимать быстрые решения в зависимости от текущих показателей.

В статье [15] с помощью такого подхода авторы предлагают решение, позволяющее следить за распорядком дня и поведением пожилых людей с целью выявления у них аномального поведения без вмешательства в их жизнь. Такое решение может помочь в уходе за пожилыми людьми с деменцией или болезнью Альцгеймера, которые живут одни.

### 1.3. Проблемы технологий в сфере мобильной медицины

В настоящий момент уровень внедрения носимых устройств не может предоставить достаточный уровень детализации информации о здоровье человека для получения точных интерпретаций. Хотя уже были зафиксированы случаи, когда собираемая с носимых устройств информация помогала спасти человека [16], для реализации подробного анализа состояния здоровья человека необходим сбор более специфичных данных.

Основной проблемой в данной сфере на текущий момент является *отсутствие механизмов согласования разнородной информации*, собираемой различными приложениями с различных сенсоров. Пользователи устанавливают на устройство несколько приложений, каждое из которых хранит и обрабатывает данные в своем формате без возможности обмена этой информацией друг с другом.

С помощью приложения Apple Health (“Здоровье”) компания Apple частично решает эту проблему, однако это коммерческий продукт, разработанный под экосистему компании Apple, что ограничивает возможность использования данного сервиса сторонними организациями. Сама компания Apple при этом не обязывает разработчиков медицинских приложений реализовать интерфейсы для обмена данными с HealthKit [17]. Разработчики при этом не получают больших преимуществ от использования Apple Health, так как он не предоставляет пользователю глубокого анализа хранящихся в приложении данных, а разработчики не имеют возможность передавать эти данные куда-либо кроме этого приложения.

Так как сейчас большинство приложений разрабатываются одновременно как под систему iOS, так и под систему Android, отсутствие возможности обмениваться данными между этими платформами ведет к необходимости разработки собственных веб-сервисов, которые будут отвечать за хранение и обработку получаемых данных. К тому же, решение Apple накладывает на пользователей ограничение в выборе используемых ими смартфонов и носимых устройств и не позволяет в реальном времени получать доступ к этим данным лечащим врачам или родственникам.

При этом нужно учитывать, что различные носимые устройства могут генерировать существенные объемы данных, вплоть до нескольких гигабайт за один день только с одного устройства [1]. Конечно, не все полученные данные необходимо передавать по сети в чистом формате, однако увеличение числа носимых устройств и генерируемой ими инфор-

мации ведет к необходимости создания соответствующей сетевой архитектуры, которая бы могла стабильно, надежно и своевременно передавать полученные данные заинтересованным лицам.

Так же, учитывая стремительный рост и появление новых технологий сферы мобильной медицины, важно учитывать и влияние доступности этих технологий на самих пациентов. Так, усиливающееся желание людей делиться своими персональными данными, в том числе и медицинскими, может привести к непредвиденным последствиям и является чрезвычайно рискованным. До появления мобильных технологий невозможно было так легко собирать ежедневную информацию о состоянии пациента. Существуют риски, что mHealth-платформа может быть использована не для тех целей, которые планировались ее создателями [18]. Поэтому к предоставляемой возможности и данным, получаемым с ее помощью, нужно относиться с максимальной осторожностью.

#### 1.4. Стандартизация подхода к реализации решений мобильной медицины

На данный момент существует множество решений в сфере мобильной медицины и большинство разрабатывается с нуля. Такой подход к разработке приводит к отсутствию согласованности разрабатываемых систем и сложности их интеграции друг с другом.

Существуют исследования, которые ориентированы на решение данной проблемы. Авторы статьи [19] предлагают архитектуру для создания систем по мониторингу пациентов посредством внешних устройств с целью получения уведомлений о тревожных состояниях здоровья и составления отчетов о состоянии пациента. Они разделяют компоненты системы на четыре уровня: сенсорный уровень, уровень восприятия, гибридный уровень обработки и уровень уведомлений (см. рис. 1). Авторы данной архитектуры разделяют модули не по функционалу, а по порядку, в котором эти модули должны друг с другом взаимодействовать. Основным преимуществом такой архитектуры выделяют возможность настройки системы под требования для конкретного случая и возможность использования с различным набором правил и сенсоров.



Рис. 1. Архитектура системы мониторинга пациентов

## 2. Анализ вариантов использования для платформы Мобильной медицины

### 2.1. Понятие платформы мобильной медицины

В рамках нашей работы мы рассматриваем платформу мобильной медицины как программную инфраструктуру, обеспечивающую взаимодействие между конечными потребителями медицинских услуг (пациентами), поставщиками медицинских услуг (медиками) и исследовательскими коллективами, занимающимися анализом медицинских данных в различных целях (исследователи). Платформа должна обеспечивать возможность следующих вариантов использования

- анонимизированный сбор, хранение и анализ медицинских данных, полученных с использованием средств мобильной медицины и медицинского интернета вещей, как на краевых устройствах конечных пользователей, так и средствами туманных и облачных вычислительных систем;
- поддержку взаимодействия между пациентом и медиком, предоставляя возможности индивидуализации путем предоставления исторической информации о медицинских данных пациента;
- предоставление механизмов сбора анонимизированных медицинских больших данных для проведения различного рода медицинских исследований на большом числе пациентов;
- предоставление API для подключения новых мобильных приложений и средств мобильной медицины и медицинского интернета вещей.

### 2.2. Концепция маркетплейса

В статье [11] мы рассмотрели существующие технологии и платформы, используемые в области мобильной медицины. На основании проведенного анализа мы пришли к выводу, что на данный момент на рынке не существует решения, предоставляющего универсальные кроссплатформенные возможности по созданию собственного сервиса в сфере мобильной медицины для разработчиков. В качестве модели взаимодействия пользователей с данным сервисом можно выделить принцип двустороннего маркетплейса, где сервис предоставляет пользователям возможности платформы с заранее определенной структурой построения данных и взаимодействия с ними, с которой компании могут интегрироваться для решения своих внутренних задач или для предоставления своих услуг, а пациенты могут получать услуги от компаний, не зависящих друг от друга, но предоставляющих свои услуги в одном месте и в заранее согласованном системой формате. Аналогом данного решения можно привести такие платформы, как:

- Coursera, на которой независимые преподаватели размещают свои курсы, доступные для всех посетителей сайта;
- AppStore, на котором независимые разработчики размещают свои приложения, доступные для всех пользователей iOS;
- Alexa Skills Kit [20], позволяющая независимым разработчикам создавать свои сценарии разговора с голосовым помощником, которыми могут воспользоваться владельцы колонок Amazon Echo.

Платформа мобильной медицины может предлагать аналогичный сценарий использования (см. рис. 2), являясь основой, на которой сторонние разработчики и компании могут раз-

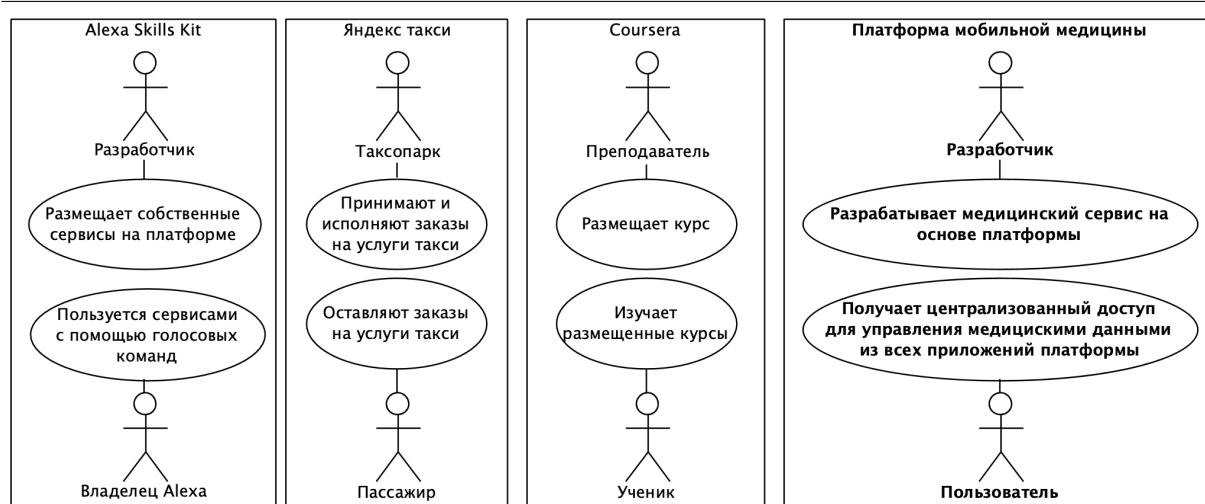


Рис. 2. Концепция работы платформы

мещать свои сервисы, доступные всем посетителям платформы. Для разработки такого сервиса необходимо подробно рассмотреть вопрос безопасной передачи и хранения данных на платформе, так как храниться и передаваться будут медицинские данные пользователей.

### 2.3. Пользователи мобильной медицины

В концепции мобильной медицины основным выделяется подход, ориентированный на пациентов. В рамках сферы информационных технологий в нем можно выделить такие типы пользователей, как пациенты, доверенные лица, медицинские работники, фармацевтические и клинические исследователи. Так как конечные пользователи могут использовать сервисы на платформе на ежедневной основе, пользователю должно быть удобно манипулировать базовыми задачами на платформе, такими как обмен данными и управление доступом. Для определения функциональных требований к платформе для каждого из перечисленных пользователей мы выявили основные сценарии использования приложений в области мобильной медицины (см. рис. 3).



Рис. 3. Варианты использования платформы мобильной медицины конечными пользователями

**Пациенты** рассчитывают получить широкий спектр медицинских услуг по доступной цене с персонализированными рекомендациями. Помимо получения клинического диагноза врача, у них есть возможность получить больше медицинских знаний с помощью цифровых платформ и общаться с похожими людьми для получения информации, например, о симптомах заболевания, побочных эффектах, госпитализации, информации о лекарствах, клинических отчетах и сценариях развития. Пациент является основным поставщиком информации о собственном здоровье с помощью мобильного устройства.

**Доверенные лица** рассчитывают на возможность получения информации о состоянии здоровья пациента, за которым они наблюдают. Хотя доверенные лица и не участвуют в лечении пациента напрямую, для них важны возможности получения обучающей информации о болезни пациента и коммуникации с пациентом.

**Медицинские работники.** Данные, полученные на различных этапах диагностики и лечения пациентов, дают медицинским работникам реальное представление о предлагаемом ходе лечения. С помощью результатов лабораторных исследований, клинических заметок и данных с сенсорных устройств врачи могут следить за состоянием здоровья пациента в любой момент времени, что помогает улучшить наблюдение за общественным здравоохранением и обеспечить быстрое реагирование благодаря эффективному анализу закономерностей развития заболеваний.

**Фармацевтические и клинические исследователи.** Использование клинических данных помогает построить прогностические модели для понимания биологических и медикаментозных процессов, которые способствуют достижению высокого уровня эффективности при разработке лекарственных препаратов. Анализ медицинских данных помогает фармацевтическим компаниям измерять результаты разработки лекарственных средств даже при малочисленных и быстрых испытаниях.

## 2.4. Акторы платформы мобильной медицины

В нашей платформе мы ориентируемся в первую очередь на разработчиков сторонних приложений в области мобильной медицины, поскольку они будут основными бенефициартами сервисов, предоставляемых платформой. Кроме того, мы должны помнить, что наша архитектура должна быть достаточно гибкой, чтобы предоставить разработчикам возможность создавать новую функциональность из базовых модулей платформы. В связи с этим, в рамках нашей платформы мы выделяем только одного ключевого актора: **стороннее приложение** для управления медицинскими данными, созданное на базе нашей платформы.

Учитывая, что платформа должна предоставлять функциональные возможности стороннему сервису, а не обычному пользователю, мы разработали диаграмму вариантов использования системы, в которой главным актором является разрабатываемое независимым разработчиком стороннее приложение (см. рис. 4).

Стороннее приложение может взаимодействовать с платформой следующим образом:

- **Получить спецификацию обмена данными с платформой** для конфигурации приложения.
- **Авторизовать пользователя** стороннего приложения для получения пользователем доступа к своим данным через API.
- **Обменять данные между устройствами** для синхронной передачи медицинских данных.



Рис. 4. Варианты использования платформы мобильной медицины сторонними приложениями

- **Обменять данные между пользователями** для асинхронной передачи медицинских данных.
- **Собрать общедоступные данные** для их дальнейшей обработки и анализа.
- **Изменить права доступа** для изменения списка пользователей, которые имеют доступ к данным.
- **Настроить уведомления**, которые будет отправлять пользователям стороннего приложения платформа.

### 3. Сравнение подходов к обмену данными в рамках платформы мобильной медицины

#### 3.1. Требования к безопасности

При разработке продукта, взаимодействующего с конфиденциальными медицинскими данными, одним из важнейших критериев для анализа является обеспечение безопасности данных, хранящихся на платформе. Чтобы решить проблему безопасности, необходимо учесть следующие важные детали при реализации платформы:

- платформа должна максимально усложнить возможность утечки конфиденциальных данных с платформы в случае ее взлома. В идеале, в принципе не должно быть механизмов централизованной обработки таких данных;
- хранение и обмен данными пользователя должен быть обеспечен в зашифрованном виде, таким образом, чтобы в случае перехвата их злоумышленником, возможность их интерпретации была максимально затруднена;
- платформа должна предоставлять пользователю возможность управлять настройками доступа к своим данным;
- платформа должна предоставлять доступ к данным пользователя только в том случае, если пользователь самостоятельно установил разрешение на доступ.

В то же время, чтобы сделать нашу платформу удобной для конечных пользователей, мы должны реализовать следующие возможности:

- легкое управление своими данными без сторонних сервисов;
- синхронизация данных между устройствами пользователя;
- обмен данными между пациентом и родственниками или пациентом и врачом.

### 3.2. Подходы к синхронизации данных

Сегодня можно выделить три наиболее распространенных варианта для организации обмена данными в мобильных приложениях.

- **Стороннее облачное хранилище.** В этом случае платформа обеспечивает возможность хранения и синхронизации данных пользователей через сторонние облачные сервисы хранения данных, такие как Google Drive или Dropbox.
- **Собственное облачное хранилище.** При таком подходе все данные пользователей хранятся в облачном хранилище, управляемым разработчиками платформы.
- **P2P-обмен данными.** При таком подходе, медицинские данные хранятся на конечных устройствах пользователей и передаются непосредственно между конечными устройствами.

Предложенные подходы имеют свои преимущества и недостатки в вопросах сохранения конфиденциальности, возможностей по обмену данными и затрат на развертывание и поддержку. В рамках нашей работы мы анализируем три основных варианта использования:

- **Персональная синхронизация.** Может ли пользователь синхронизировать данные между собственными устройствами?
- **Совместный доступ к данным.** Может ли пользователь обмениваться данными с другими учетными записями?
- **Анонимный обмен данными.** Может ли пользователь анонимно делиться своими данными в исследовательских целях?

В рамках данного исследования мы не фокусируемся на вопросах реализации алгоритма шифрования данных в хранилищах, генерации, хранения и обмена ключами, а также специфики реализации передачи зашифрованных данных, так как описание данных процессов выходит за рамки данной работы. Все приведенные способы авторизации, шифрования и передачи данных предполагают использование стандартных средств, предоставляемых выбранной технологией для разработки. Передача данных выполняется только по протоколу по HTTPS, а доступ к данным на локальных устройствах требует авторизации пользователя средствами операционной системы.

Далее в качестве типового примера рассмотрим сценарий организации обмена данными между пользователем на платформе iOS и доверенным лицом на платформе Android.

#### 3.2.1. Стороннее облачное хранилище

При использовании сторонних облачных сервисов мы должны руководствоваться методами хранения и обмена данными файловых систем. В этом случае медицинские данные должны находиться в специальной папке внутри стороннего облачного хранилища, которой он может поделиться с другими пользователями. При таком подходе доверенное лицо должно принять приглашение на доступ к папке, подключиться к выбранному облачному хранилищу, после чего оно сможет просматривать данные пользователя.

В случае использования стороннего облачного хранилища процесс синхронизации включает в себя следующие шаги:

- подключить облачное хранилище в приложении iOS;
- отправить приглашение для доверенного лица;
- подключить облачное хранилище в приложении Android;
- обменяться ключами шифрования;
- асинхронно синхронизировать данные.

Важным преимуществом данного подхода является возможность значительно снизить затраты платформы на хранение данных, так как каждый пользователь самостоятельно отвечает за свои учетные записи в сторонних облачных хранилищах. К тому же, с таким подходом снижается ответственность за обеспечение безопасности хранения и передачи данных, так как данные хранятся и передаются только через устройства и сервисы пользователя.

Однако главным недостатком данного подхода являются ограничения, вызванные механизмами совместного использования и функциональностью файловой системы выбранного облачного хранилища. Для общего доступа к платформе, оба пользователя должны иметь учетные записи в одном и том же стороннем облачном хранилище, сама настройка обмена данными довольно трудоемка и может быть недостаточно комфортной, а пользователям придется самостоятельно следить за оставшимся свободным дисковым пространством, что сильно ограничивает доступность платформы.

Для исследовательских целей данный подход не предоставляет никаких возможностей, так как в нем отсутствуют механизмы удобного получения данных от большого количества анонимных пользователей.

### *3.2.2. Собственное облачное хранилище*

В этом случае медицинские данные хранятся в облачном хранилище, которым управляют владельцы платформы мобильной медицины. Для того чтобы использовать платформу, пользователи должны пройти авторизацию, после чего их данные будут передаваться и сохраняться в веб-сервисе.

Чтобы реализовать обмен данными с помощью этого подхода, нам необходим уникальный ключ шифрования для каждого из получателей. Данные, предназначенные для конкретного получателя, шифруются с помощью этого ключа, а затем отправляются в веб-сервис, откуда получатель может их скачать.

В случае организации обмена данными с доверенным лицом посредством собственного облачного хранилища платформы, процесс синхронизации состоит из шести шагов:

- авторизовать пользователя в приложении iOS;
- авторизовать доверенное лицо в приложении Android;
- предоставить доступ к данным для учетной записи доверенного лица;
- принять доступ к учетной записи пользователя;
- обменяться ключами шифрования;
- асинхронно синхронизировать данные.

В случае использования собственного облачного хранилища данных, мы можем предоставить нашим конечным пользователям гораздо больше функциональных возможностей, поскольку все данные хранятся внутри нашей платформы и мы получаем возможность реализации любой необходимой нам функциональности. С таким подходом мы можем легко



интегрировать различные способы синхронизации между любыми типами конечных пользователей, включая возможность организации сбора данных большого числа конечных пользователей для организации анализа медицинских данных для независимых исследовательских групп.

Однако, поскольку все данные хранятся на облачной платформе ответственность за обеспечение их безопасности полностью ложится на платформу. Также в случае выбора данного подхода необходимо рассмотреть возможные способы финансовой поддержки облачного хранилища, так как увеличение числа пользователей резко повысит стоимость хранения данных и обслуживания облачного решения.

### 3.2.3. Peer-to-peer обмен данными

В случае организации обмена данными посредством однорангового (peer-to-peer) соединения (см. рис. 5), медицинские данные хранятся только на устройствах пользователей и передаются непосредственно между ними.

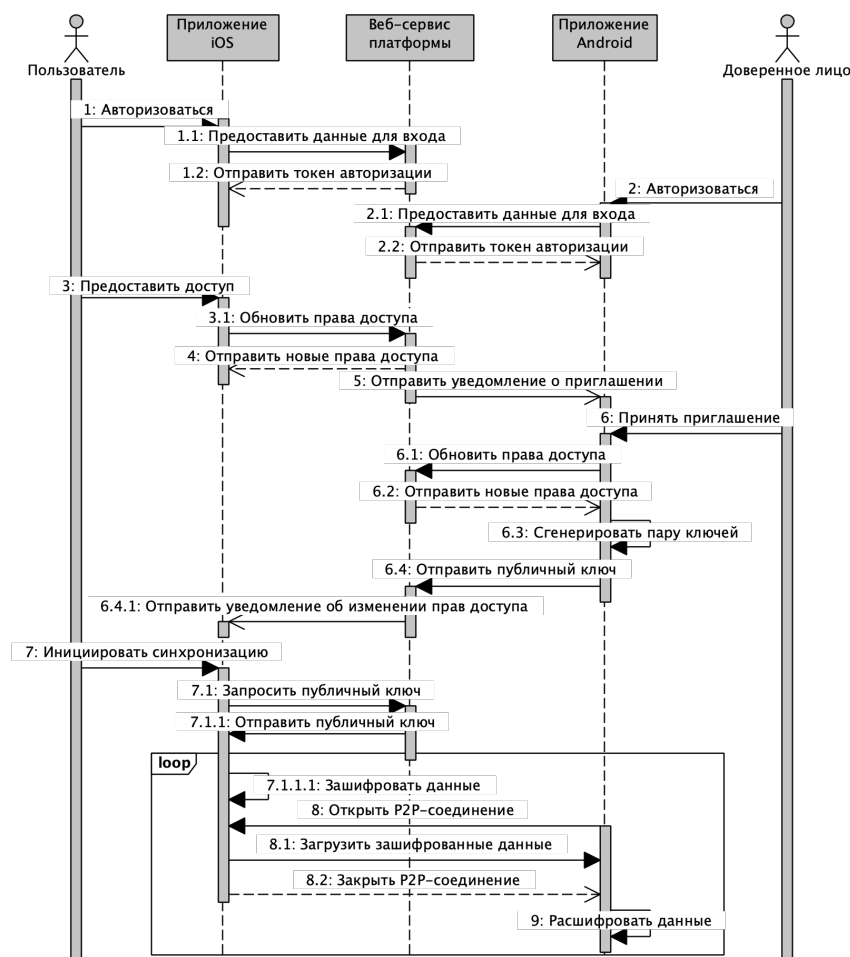


Рис. 5. Обмен данными с использованием peer-to-peer соединения

При реализации этого подхода мы отказываемся от идеи централизованного хранения медицинских данных пользователей в веб-сервисе нашей платформы, ограничиваясь хранением информации об учетных записях пользователей и их настройках доступа.

Данный подход снижает риски, связанные с возможным раскрытием медицинских пользовательских данных, в связи с децентрализацией их хранения и обмена. Также значительно-

но снижаются затраты на хранение данных, поскольку для этого используются устройства конечных пользователей.

Однако этот подход ограничивает нас в функциональности и гибкости платформы, поскольку у нас нет прямого доступа к данным пользователей и обмен данными происходит напрямую между двумя устройствами а синхронизация возможна только тогда, когда оба устройства находятся в сети, что затрудняет асинхронную синхронизацию данных между устройствами. Также, необходимо учитывать факт, что не у всех пользователей может быть достаточно дискового пространства на конечном устройстве. Еще одной особенностью данного решения является необходимость отдельной настройки резервных копий локальных данных средствами операционной системы либо сторонними сервисами.

### 3.2.4. Сравнение подходов обмена данными

Сравнив эти подходы (см. таблицу), можно сделать следующие выводы.

**Таблица 1.** Сравнение подходов обмена данными

	Персональная синхронизация	Совместный доступ	Анонимный обмен данными
Стороннее облачное хранилище	+	±	—
Собственное облачное хранилище	+	+	+
Peer-to-peer соединение	+	+	±

**Стороннее облачное хранилище** позволяет уменьшить стоимость реализации решения, но не сможет обеспечить нужную функциональность нашей платформы и не позволит нам ее масштабировать. Кроме того, механизмы синхронизации на основе сторонних сервисов будет сложно поддерживать в долгосрочной перспективе.

**Собственное облачное хранилище** является наиболее перспективным с точки зрения функциональности, которую мы можем предоставить конечным пользователям. Но вызывает вопрос возможность финансовой поддержки такого подхода, учитывая цену облачных решений.

Обмен с использованием **peer-to-peer связи** является компромиссным, так как предоставляет возможность создания практически всей необходимой функциональности, а также решает вопрос финансовой стоимости обмена, хотя и обладает некоторыми особенностями, связанными с необходимостью реализации синхронного обмена данными.

## 4. Реализация прототипа системы для обмена медицинскими данными

Для демонстрации механизмов обмена данными между двумя устройствами с помощью peer-to-peer соединения мы разработали прототип веб-сервиса и мобильного приложения, обеспечивающих эту функциональность. Рассмотрим процесс обмена данными между пациентом и его доверенным лицом с использованием представленного решения (см. рис. 6).

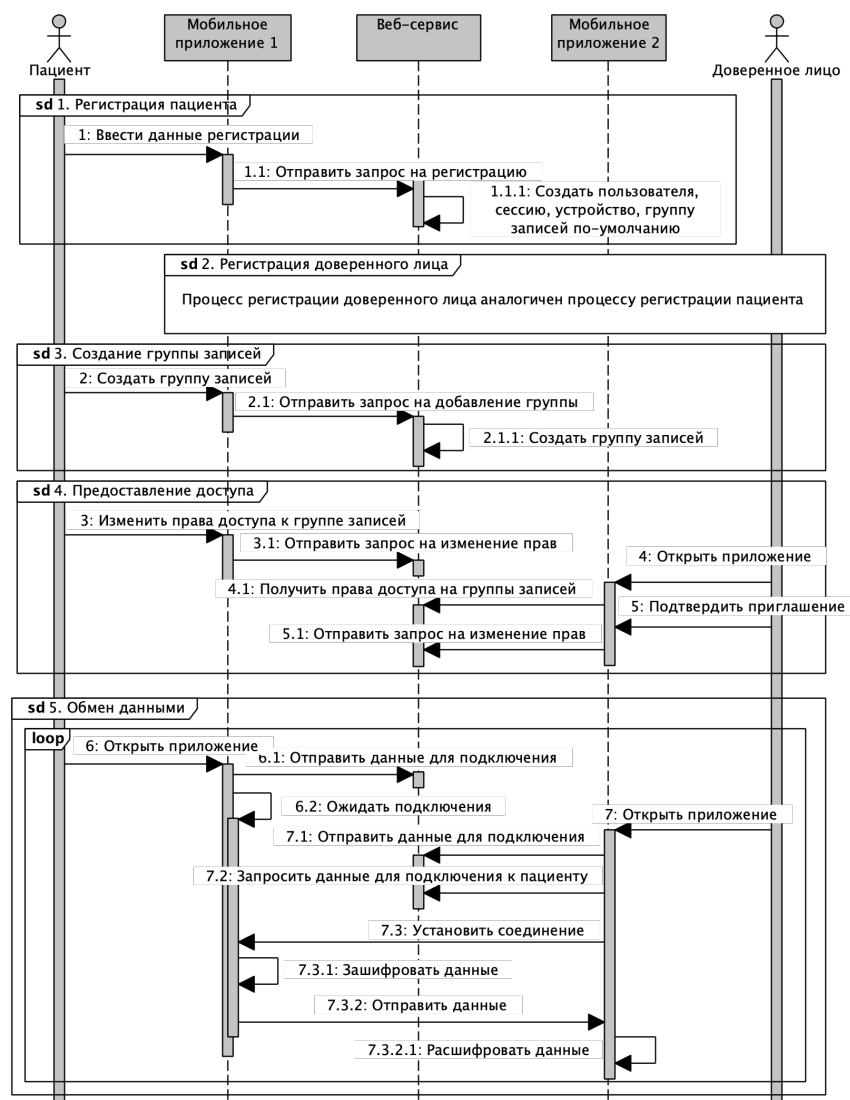


Рис. 6. Диаграмма последовательности процесса обмена данными между пациентом и доверенным лицом в рамках разработанной платформы

Организации обмена данными с доверенным лицом в рамках нашего прототипа состоит из пяти шагов:

- **Регистрация Пациента.** После запуска приложения Пациент может зарегистрироваться на платформе для создания учетной записи.
- **Регистрация доверенного лица.** Для получения данных пациента доверенному лицу также необходимо зарегистрироваться на платформе.
- **Создание группы записей.** На данном этапе пациент может создать для своих медицинских показателей группу записей, настройки которой будут синхронизироваться с веб-сервисом.
- **Предоставление доступа.** После создания группы записей пользователь может отправить приглашение на доступ к данным внутри группы любому другому пользователю платформы. В данном случае пользователь отправляет приглашение на группу по email доверенного лица. Доступ к данным у доверенного лица появляется только после принятия им прав на доступ. После принятия приглашения на сервис отправляется событие о изменении прав. Хотя в рамках данной диаграммы последовательности

- мы не рассматриваем процесс отклонения доступа, стоит отметить, что после принятия прав пользователь может самостоятельно в любой момент отказаться от доступа.
- **Обмен данными.** Данный этап может повторяться сколько угодно раз после прохождения предыдущих шагов. В случае открытия доверенным лицом приложения устройство отправляет на веб-сервис запрос на получение данных для подключения к устройству пациента. В случае успешного установления прямого соединения данные на устройстве пациента передаются на устройство доверенного лица посредством peer-to-peer подключения. Перед отправкой на устройстве пациента данные предварительно шифруются публичным ключом доверенного лица. После получения данных доверенное лицо может расшифровать их с помощью своего приватного ключа.

Для разработки веб-сервиса был выбран язык программирования Python с использованием фреймворка Django. Мобильное приложение реализовано для iOS нативными средствами на языке программирования Swift. Нами было принято решение разработать приложение только для операционной системы iOS, так как концептуально в рамках нашей платформы разница между обменом с iOS на iOS и с iOS на Android отсутствует.

#### 4.1. Схема серверной базы данных

Для работы веб-сервиса была разработана схема базы данных (см. рис. 7), содержащая информацию о пользователях системы, сессиях, группах записей и правах доступа. Рассмотрим таблицы и представленные в них атрибуты подробнее.

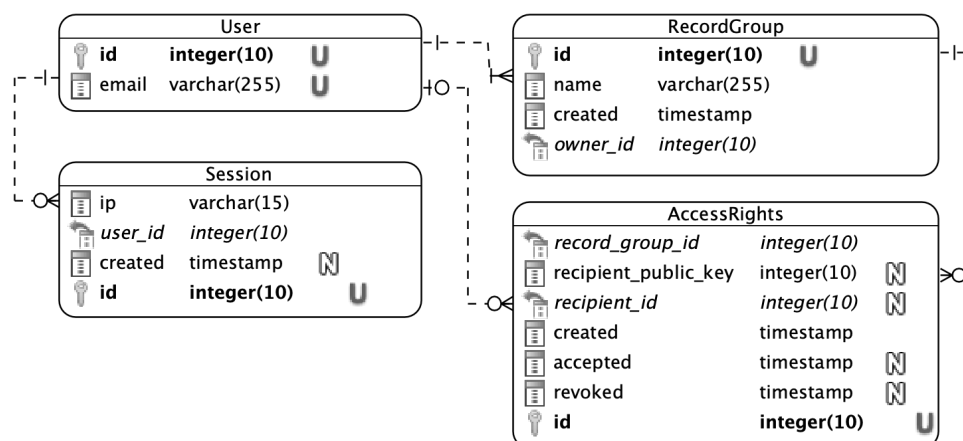


Рис. 7. Схема серверной базы данных

**User** — хранит информацию о существующих на платформе пользователях. В качестве универсального имени пользователя было выбрано использовать поле электронной почты.

**Session** — хранит все сессии пользователей, взаимодействующих с платформой. Содержит в себе информацию об ip-адресе пользователя на момент создания сессии. У каждого пользователя может быть неограниченное количество сессий.

**RecordGroup** — хранит информацию о всех имеющихся у пользователя группах данных. Группы представляют собой именованные категории данных, у которых отсутствует связь с самими данными на сервере, но которые стороннее приложение может использовать для управления доступом к данным локально. У каждого пользователя может быть неограниченное количество групп.

**AccessRights** — хранит информацию о всех правах доступа к группе данных пользователя. В нашем примере права доступа реализованы без детализации конкретных прав на управление данными. Для реализации такой возможности таблица может быть расширена дополнительными параметрами в будущем. Хранит в себе ссылку на группу записей к которой предоставляется доступ и публичный ключ получателя прав. У одной группы данных можно задавать неограниченное число прав доступа.

## 4.2. API веб-сервиса

Для взаимодействия с веб-сервисом был разработан REST API. Реализованные запросы можно разделить на четыре группы: авторизация, управление данными и управление доступом.

В **группе авторизации** были реализованы запросы регистрации, входа и обновления токена. При корректно введенных данных сервис возвращает JSON-объект описывающий авторизованного пользователя, его текущую сессию и пару токенов.

Для реализации механизма авторизации нами был выбран подход с использованием пары токенов *refresh* и *access*. При предоставлении данных авторизации пользователю будет возвращаться:

- *access* — токен доступа, в дальнейшем передается клиентом в заголовке запроса и по нему сервер определяет все ли еще активна ли сессия пользователя.
- *refresh* — токен обновления, используется после окончания срока действия *access*-токена для получения нового *access*-токена.

Время жизни *access*-токена должно быть меньше времени жизни *refresh*-токена, чтобы клиент мог получить новый *access*-токен без необходимости повторного ввода логина и пароля. В рамках данной работы время жизни *access*-токена было установлено в 30 минут, а *refresh*-токена — в 365 дней.

После получения клиентом *access*-токена он должен отправляться в заголовке каждого запросе пользователя в формате “Authorization”: “Bearer **token**”, где **token** — значение *access*-токена.

В **группе управления данными** реализованы запросы на создание новой группы, удаление группы и получение списка групп пользователя.

После создания пользователя ему по умолчанию создается группа данных с наименованием “Общая”, в которой по-умолчанию располагаются все данные пользователя.

В **группе управления доступом** реализованы запросы на отправку доступа к группе, принятие доступа, отклонение доступа и на получение данных для инициализации обмена. При подтверждении или отклонении доступа, API автоматически записывает в соответствующее поле *accepted* или *revoked* текущую дату. При принятии прав доступа пользователь дополнительно отправляет свой публичный ключ шифрования данных, который будет необходим владельцу группы для последующего шифрования данных.

## 4.3. Прототип мобильного приложения

Для демонстрации процесса обмена данными мы разработали прототип мобильного приложения, позволяющего пользователю зарегистрироваться и войти в систему, управлять своими группами данных, правами доступа к ним, и обмениваться данными с другими пользователями (см. рис. 8).

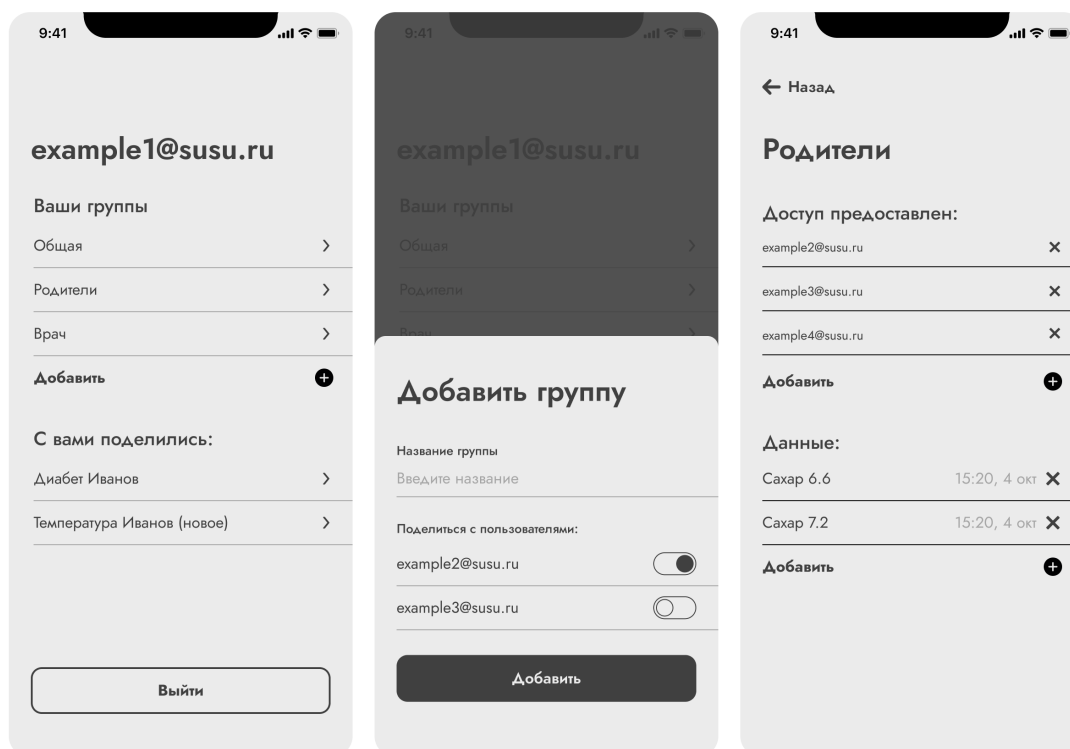


Рис. 8. Главный экран, добавление группы и просмотр группы

На данном примере мы можем показать процесс обмена данными между пациентом и его доверенным лицом. Мобильное приложение состоит из нескольких разделов, рассмотрим подробнее каждый из них.

#### 4.3.1. Регистрация/вход

При открытии приложения пользователь попадает на экран Входа, с помощью которого может войти в систему или перейти на экран Регистрации. Поля для входа и регистрации совпадают, так как для демонстрационных целей профиль пользователя не содержит в себе дополнительной информации. После успешного входа или регистрации в системе приложение запоминает полученного пользователя и его access и refresh токены.

#### 4.3.2. Главный экран и группы данных

После авторизации в приложении пользователь попадает на главный экран. Все последующие перезапуски приложения будут выводить пользователя на главный экран до момента, пока он не выйдет из своего аккаунта, либо пока не истечет его access и refresh токены.

Сразу после открытия приложения: при открытии данного экрана, приложение отправляет на веб-сервис запрос на создание у пользователя новой сессии, благодаря чему сервер запоминает актуальный IP-адрес устройства пользователя.

На данном экране пользователь может видеть свои группы данных и группы данных, которыми с ним поделились. Распределение групп мобильное приложение выполняет по полю `owner_email`.

При выборе пользователем одной из групп данных открывается экран с ее деталями, на котором отображаются все данные, находящиеся в этой группе и пользователи, которым предоставлен доступ к этой группе. При добавлении пользователем новой записи в группу данных приложение предлагает пользователю поделиться этой записью сразу и с другими группами. По умолчанию все записи также отправляются в основную группу пользователя “Общая”.

#### 4.3.3. Наблюдатели

При просмотре группы данных, к которым пользователю был предоставлен доступ, интерфейс приложения отличается. Наблюдатели не могут добавлять новые данные в группы, а могут только просматривать содержащуюся там информацию. До принятия приглашения пользователь не видит содержащиеся в группе данные. После принятия приглашения пользователь может в любой момент самостоятельно отписаться от группы.

## Заключение

В рамках данной статьи, нами был проведен анализ существующих проблем, связанных с организацией обмена данными в области мобильной медицины. К сожалению, сегодня отсутствуют единые стандартизованные механизмы организации такого обмена, что приводит к фрагментации и отсутствию синергии между мобильными приложениями, обеспечивающими сбор и обработку данных мобильного интернета вещей. Нами предлагается понятие платформы мобильной медицины, которая позволила бы решить данную задачу. На основе анализа вариантов использования такой платформы, мы определили и проанализировали три возможных способа организации обмена данными с учетом ограничений, накладываемых предметной областью мобильной медицины. По результатам сравнения данных решений, наиболее перспективным был признан вариант организации однорангового обмена данными. Нами был реализован прототип мобильного приложения, обеспечивающий возможность синхронизации, при котором медицинские данные хранятся на конечных устройствах пользователя, а веб-сервис отвечает исключительно за управление авторизацией пользователей и распределением прав доступа.

Для эффективной реализации платформы мобильной медицины в дальнейших работах мы планируем детальнее проработать вопрос безопасности хранения и передачи данных, разработать архитектуру других модулей платформы и определить общую механику подключения сторонних приложений к платформе мобильной медицины.

*Секции 1, 3 исследования была выполнена при финансовой поддержке РФФИ и Челябинской области в рамках научного проекта № 20-47-740005, разделы 2, 4 были выполнены при финансовой поддержке Министерства науки и высшего образования РФ (государственное задание FENU-2020-0022).*

## Литература

1. Farahani B., Firouzi F., Chang V. *et al.* Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare // Future Generation Computer Systems. 2018. Vol. 78. P. 659–676. DOI: 10.1016/j.future.2017.04.036.

2. United Nations. World population ageing 2013. United Nations, 2014. 114 p. DOI: 10.18356/30d0966c-en.
3. Yang W., Dall T.M., Halder P. *et al.* Economic costs of diabetes in the U.S. in 2012 // *Diabetes Care*. 2013. Vol. 36. P. 1033–1046. DOI: 10.2337/dc12-2625.
4. Carozzi F., Provenzano S., Roth S. Urban Density and Covid-19 // SSRN. 2020. DOI: 10.2139/ssrn.3643204.
5. Qudah B., Luetsch K. The influence of mobile health applications on patient - healthcare provider relationships: A systematic, narrative review // *Patient Education and Counseling*. 2019. Vol. 102, no. 6. P. 1080–1089. DOI: 10.1016/j.pec.2019.01.021.
6. Free C., Phillips G., Watson L. *et al.* The Effectiveness of Mobile-Health Technologies to Improve Health Care Service Delivery Processes: A Systematic Review and Meta-Analysis // *PLoS Medicine*. 2013. Vol. 10. P. e1001363. DOI: 10.1371/journal.pmed.1001363.
7. Healthcare IT Market: Healthcare Information Technology Market Trends & Global Forecast (2010–2015) | MarketsandMarkets. URL: <https://www.marketsandmarkets.com/Market-Reports/healthcare-information-technology-market-136.html> (дата обращения: 12.01.2021).
8. Healthcare IT Market Size, Share and Trends forecast to 2026 by Products & Services, Components, End User | COVID-19 Impact Analysis | MarketsandMarkets. URL: <https://www.marketsandmarkets.com/Market-Reports/healthcare-it-252.html> (дата обращения: 12.01.2021).
9. Волков И.А. Мобильное приложение для ведения сахарного диабета - DiaMeter // Материалы XV Итоговой научно-практической конференции молодых ученых и специалистов, Челябинск, Россия, 26 апреля, 2017. 2017. С. 25–28.
10. Volkov I., Radchenko G. DiaMeter: a Mobile Application and Web Service for Monitoring Diabetes Mellitus // 2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBREIT), Yekaterinburg, Russia, May 14–15, 2020. IEEE, 2020. P. 0384–0387. DOI: 10.1109/USBREIT48449.2020.9117654.
11. Volkov I., Radchenko G., Tchernykh A. Digital Twins, Internet of Things and Mobile Medicine: a Review of Current Platforms to Support Smart Healthcare // *CoRR*. 2021. Vol. abs/2106.11728. URL: <https://arxiv.org/abs/2106.11728>.
12. Steinhubl S.R., Muse E.D., Topol E.J. The emerging field of mobile health // *Science Translational Medicine*. 2015. Vol. 7. P. 283rv3–283rv3. DOI: 10.1126/scitranslmed.aaa3487.
13. Mastoi Q.U.A., Wah T.Y., Raj R.G., Lakhan A. A novel cost-efficient framework for critical heartbeat task scheduling using the internet of medical things in a fog cloud system // *Sensors*. 2020. Vol. 20, no. 2. P. 441. DOI: 10.3390/s20020441.
14. Zheng Y.L., Ding X.R., Poon C.C.Y. *et al.* Unobtrusive sensing and wearable devices for health informatics // *IEEE Transactions on Biomedical Engineering*. 2014. Vol. 61, no. 5. P. 1538–1554. DOI: 10.1109/TBME.2014.2309951.



15. Meng L., Miao C., Leung C. Towards online and personalized daily activity recognition, habit modeling, and anomaly detection for the solitary elderly through unobtrusive sensing // *Multimedia Tools and Applications*. 2017. Vol. 76, no. 8. P. 10779–10799. DOI: 10.1007/s11042-016-3267-8.
16. Rudner J., McDougall C., Sailam V. *et al.* Interrogation of Patient Smartphone Activity Tracker to Assist Arrhythmia Management // *Annals of Emergency Medicine*. 2016. Vol. 68. P. 292–294. DOI: 10.1016/j.annemergmed.2016.02.039.
17. HealthKit | Apple Developer Documentation. URL: <https://developer.apple.com/documentation/healthkit> (дата обращения: 12.01.2021).
18. Jardine J., Fisher J., Carrick B. Apple’s ResearchKit: smart data collection for the smartphone era? // *Journal of the Royal Society of Medicine*. 2015. Vol. 108, no. 8. P. 294–296. DOI: 10.1177/0141076815600673.
19. Esposito M., Minutolo A., Megna R. *et al.* A smart mobile, self-configuring, context-aware architecture for personal health monitoring // *Engineering Applications of Artificial Intelligence*. 2018. Vol. 67. P. 136–156. DOI: 10.1016/j.engappai.2017.09.019.
20. Create Alexa Skills Kit | Amazon Alexa Voice Development. URL: <https://developer.amazon.com/en-US/alexa/alexa-skills-kit> (дата обращения: 12.01.2021).

Волков Иван Алексеевич, аспирант, кафедра электронно-вычислительных машин, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Радченко Глеб Игоревич, к.ф.-м.н., доцент, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация).

Черных Андрей Николаевич, к.т.н., доцент, заведующий научно-исследовательской лабораторией проблемно-ориентированных облачных сред, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация), профессор, научно-исследовательский центр Энсенады (Энсенада, Мексика).

## DATA SHARING IN MOBILE HEALTH PLATFORM

© 2021 I.A. Volkov<sup>1</sup>, G.I. Radchenko<sup>1</sup>, A.N. Tchernykh<sup>1,2</sup>

<sup>1</sup>South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia),

<sup>2</sup>Ensenada Research Center

(Carretera Ensenada – Tijuana No. 3918, Ensenada, 22860 Mexico)

E-mail: volkovia@susu.ru, gleb.radchenko@susu.ru, chernykh@cicese.mx

Received: 03.11.2021

Mobile health is an important tool for continuous health monitoring, which opens up new opportunities for physicians, patients, and researchers. The exponential growth of the field resulted in an increase rate of appearance of new solutions in the healthcare market and, consequently, an increase in the amount of information about the health of the population. However, nearly all of the information collected by these services is isolated from each other, as it is distributed across different sites, services and mobile apps. The lack of a user's ability to conveniently manage and share their medical data is the major problem of the sphere at the moment. The creation of the Mobile Medicine platform can be a solution to this problem, offering a framework on which third-party developers and companies can host their services, available to all visitors of the platform. In our work we analyze existing solutions in the mobile medicine market, provide use cases for Mobile medicine services, propose the concept of the platform, develop use cases for Mobile medicine platform, highlight and compare three ways of data exchange variants and based on the comparison select and implement a prototype of data exchange within Mobile Medicine platform.

*Keywords: smart healthcare, mobile medicine, health monitoring, platform, PaaS, internet of things, mobile applications.*

### FOR CITATION

Volkov I.A., Radchenko G.I., Tchernykh A.N. Data Sharing in Mobile Health Platform. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2021. Vol. 10, no. 4. P. 37–59. (in Russian) DOI: 10.14529/cmse210403.

*This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.*

### References

1. Farahani B., Firouzi F., Chang V. *et al.* Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare. Future Generation Computer Systems. 2018. Vol. 78. P. 659–676. DOI: 10.1016/j.future.2017.04.036.
2. United Nations. World population ageing 2013. United Nations, 2014. 114 p. DOI: 10.18356/30d0966c-en.
3. Yang W., Dall T.M., Halder P. *et al.* Economic costs of diabetes in the U.S. in 2012. Diabetes Care. 2013. Vol. 36. P. 1033–1046. DOI: 10.2337/dc12-2625.
4. Carozzi F., Provenzano S., Roth S. Urban Density and Covid-19. SSRN. 2020. DOI: 10.2139/ssrn.3643204.
5. Qudah B., Luetsch K. The influence of mobile health applications on patient - healthcare provider relationships: A systematic, narrative review. Patient Education and Counseling.

2019. Vol. 102, no. 6. P. 1080–1089. DOI: 10.1016/j.pesc.2019.01.021.
6. Free C., Phillips G., Watson L. *et al.* The Effectiveness of Mobile-Health Technologies to Improve Health Care Service Delivery Processes: A Systematic Review and Meta-Analysis. *PLoS Medicine*. 2013. Vol. 10. P. e1001363. DOI: 10.1371/journal.pmed.1001363.
  7. Healthcare IT Market: Healthcare Information Technology Market Trends & Global Forecast (2010–2015) | MarketsandMarkets. URL: <https://www.marketsandmarkets.com/Market-Reports/healthcare-information-technology-market-136.html> (accessed: 12.01.2021).
  8. Healthcare IT Market Size, Share and Trends forecast to 2026 by Products & Services, Components, End User | COVID-19 Impact Analysis | MarketsandMarkets. URL: <https://www.marketsandmarkets.com/Market-Reports/healthcare-it-252.html> (accessed: 12.01.2021).
  9. Volkov I.A. Mobile Application for Monitoring Diabetes Mellitus - DiaMeter. Proceedings of the XV Scientific and Practical Conference of Young Scientists and Specialists, Chelyabinsk, Russia, April 26, 2017. 2017. P. 25–28. (in Russian).
  10. Volkov I., Radchenko G. DiaMeter: a Mobile Application and Web Service for Monitoring Diabetes Mellitus. 2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBREIT), Yekaterinburg, Russia, May 14–15, 2020. IEEE, 2020. P. 0384–0387. DOI: 10.1109/USBREIT48449.2020.9117654.
  11. Volkov I., Radchenko G., Tchernykh A. Digital Twins, Internet of Things and Mobile Medicine: a Review of Current Platforms to Support Smart Healthcare. *CoRR*. 2021. Vol. abs/2106.11728. URL: <https://arxiv.org/abs/2106.11728>.
  12. Steinhubl S.R., Muse E.D., Topol E.J. The emerging field of mobile health. *Science Translational Medicine*. 2015. Vol. 7. P. 283rv3–283rv3. DOI: 10.1126/scitranslmed.aaa3487.
  13. Mastoi Q.U.A., Wah T.Y., Raj R.G., Lakhan A. A novel cost-efficient framework for critical heartbeat task scheduling using the internet of medical things in a fog cloud system. *Sensors*. 2020. Vol. 20, no. 2. P. 441. DOI: 10.3390/s20020441.
  14. Zheng Y.L., Ding X.R., Poon C.C.Y. *et al.* Unobtrusive sensing and wearable devices for health informatics. *IEEE Transactions on Biomedical Engineering*. 2014. Vol. 61, no. 5. P. 1538–1554. DOI: 10.1109/TBME.2014.2309951.
  15. Meng L., Miao C., Leung C. Towards online and personalized daily activity recognition, habit modeling, and anomaly detection for the solitary elderly through unobtrusive sensing. *Multimedia Tools and Applications*. 2017. Vol. 76, no. 8. P. 10779–10799. DOI: 10.1007/s11042-016-3267-8.
  16. Rudner J., McDougall C., Sailam V. *et al.* Interrogation of Patient Smartphone Activity Tracker to Assist Arrhythmia Management. *Annals of Emergency Medicine*. 2016. Vol. 68. P. 292–294. DOI: 10.1016/j.annemergmed.2016.02.039.
  17. HealthKit | Apple Developer Documentation. URL: <https://developer.apple.com/documentation/healthkit> (accessed: 12.01.2021).

18. Jardine J., Fisher J., Carrick B. Apple's ResearchKit: smart data collection for the smart-phone era? *Journal of the Royal Society of Medicine*. 2015. Vol. 108, no. 8. P. 294–296. DOI: 10.1177/0141076815600673.
19. Esposito M., Minutolo A., Megna R. *et al.* A smart mobile, self-configuring, context-aware architecture for personal health monitoring. *Engineering Applications of Artificial Intelligence*. 2018. Vol. 67. P. 136–156. DOI: 10.1016/j.engappai.2017.09.019.
20. Create Alexa Skills Kit | Amazon Alexa Voice Development. URL: <https://developer.amazon.com/en-US/alexa/alexa-skills-kit> (accessed: 12.01.2021).

## СИСТЕМА УЧЕТА ПОСЕЩАЕМОСТИ СТУДЕНТОВ НА ОСНОВЕ РАСПОЗНАВАНИЯ ЛИЦ

© 2021 Е.В. Иванова, А.Ю. Струева

*Южно-Уральский государственный университет  
(454080 Челябинск, пр. им. В.И. Ленина, д. 76)*

*E-mail: elena.ivanova@susu.ru, nastasy822@gmail.com*

Поступила в редакцию: 26.09.2021

В настоящее время одним из значимых факторов для повышения качества подготовки специалистов является учет посещаемости студентов. Данный процесс может быть автоматизирован. В статье предлагается подход к построению системы учета посещаемости студентов на основе технологии распознавания лиц, которая позволяет идентифицировать множество людей одновременно без прямого контакта с ними и без использования дорогостоящего оборудования. Данный подход основан на сверточных нейронных сетях RetinaFace и ResNet, выбранных на основе обзора современных методов распознавания лиц, представленного в статье. Архитектура нашей системы учета посещаемости дополнена процедурами предобработки изображений, которые по предложенной нами методике, основанной на мере BREN, проверяют качество изображения и при необходимости применяют к изображению алгоритмы для уменьшения шума, повышения резкости, увеличения яркости и выравнивания цветов. Представлены результаты вычислительных экспериментов, показавшие более высокую эффективность предложенного подхода по сравнению с аналогами.

*Ключевые слова: распознавание лиц, система учета посещаемости, сверточная нейронная сеть, RetinaFace, FaceNet, предобработка изображений.*

### ОБРАЗЕЦ ЦИТИРОВАНИЯ

Иванова Е.В, Струева А.Ю. Система учета посещаемости студентов на основе распознавания лиц // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 4. С. 60–73. DOI: 10.14529/cmse210404.

### Введение

Одной из актуальных задач современного высшего образования является повышение качества подготовки специалистов. Для решения данной задачи недостаточно совершенствования образовательного процесса. Одним из значимых факторов, влияющих на качество подготовки специалистов, является посещаемость обучающимися учебных занятий [1]. В настоящее время существуют различные способы автоматизации учета посещаемости, например, с помощью считывателей отпечатков пальцев, RFID-меток, QR-кодов, смартфонов и др. Однако эти методы требуют дорогостоящих устройств [18]. Система посещаемости, основанная на технологии распознавания лиц, может идентифицировать множество людей одновременно без прямого контакта с ними и без дорогостоящего оборудования [20].

Распознавание лиц состоит из трех этапов: обнаружение лиц (Face Detection), извлечение признаков (Features Extraction) и идентификация лиц (Face Recognition) [14].

На этапе обнаружения лиц происходит определение местоположения человеческих лиц на входном изображении. Помимо фотографий, на вход может подаваться также видеопоток, такие системы применяются для идентификации людей в режиме реального времени. Цель этого этапа — определить наличие лиц на входном изображении и вычислить координаты прямоугольников, описывающих эти лица.

Основная задача *этапа извлечения признаков* — выявлять особенности лиц, обнаруженных на предыдущем этапе. Лицо представляется как набор векторов признаков (*дескриптор лица*), который описывает характерные особенности изображения лица, такие как рот, нос и глаза, с их геометрическим распределением.

На *этапе идентификации лиц* полученный набор векторов признаков каждого лица сравнивается с набором дескрипторов лиц, хранящихся в базе данных (Faces Database). Результатом данного этапа является сравнение лица с известным набором лиц с целью найти наиболее вероятное совпадение, либо с целью принять или отклонить какое-либо действие, например, запрос на доступ.

В данной работе мы описываем методы распознавания лиц, используемые для построения систем учета посещаемости, исследуем причины, влияющие на снижение точности распознавания, и предлагаем подход к построению системы учета посещаемости студентов, позволяющий улучшить точность распознавания.

Статья имеет следующую структуру. В разделе 1 представлен обзор методов распознавания лиц и систем учета посещаемости на основе распознавания лиц. Далее в разделе 2 представлена архитектура нашей системы учета посещаемости, изучаются проблемы снижения точности распознавания лиц и способы их преодоления. В разделе 3 представлены результаты вычислительных экспериментов. Заключение резюмирует результаты исследования.

## 1. Обзор связанных работ

### 1.1. Методы распознавания лиц

Системы учета посещаемости в вузе должны работать с большим количеством распознаваемых студентов, количество новых людей возрастает с каждым новым учебным годом. Для такой системы оптимальным вариантом является наличие базы данных, содержащей информацию для идентификации лиц. При этом алгоритмы распознавания лиц в самой системе и используемые нейронные сети не должны меняться и дообучаться, т.к. на это потребуется очень большое количество времени и ресурсов. Далее будем рассматривать методы распознавания лиц, пригодные для реализации систем такого вида.

Задача обнаружения лиц на фотографии или видеопотоке является первым этапом в процессе решения задачи распознавания лиц. Идеальным для обнаружения является лицо анфас, равномерно освещенное без жестких теней, не закрытое волосами и аксессуарами (очками, шарфом, головным убором) в хорошем разрешении.

В основе многих современных алгоритмов обнаружения объектов лежат идеи, предложенные П. Виолой и М. Джонсом в начале 2000х годов [26, 28]. Метод Виолы—Джонса основывается на извлечении из изображения черт лица функциями Хаара и использовании каскада слабых классификаторов для быстрого определения лица в скользящем по изображению окне. Метод Виолы—Джонса является одним из лучших по соотношению показателей эффективности распознавания/скорость работы, однако точность распознавания снижается при повороте головы и плохой освещенности.

Другой класс методов, основывается на сравнении каждого участка изображения с заданным шаблоном [15, 31]. Такие алгоритмы достаточно точны, но не подходят для задач реального времени, так как имеют очень низкую скорость работы.

Популярными системами на сегодняшний день являются системы, построенные на базе сверточных нейронных сетей (Convolutional Neural Network, CNN). Сверточная нейронная

сеть способна обнаруживать лица при большом наклоне головы и плохом освещении, однако требуют много вычислительных ресурсов. Сверточные нейронные сети предоставляют исследователю широкий спектр настройки, позволяющей достигать высоких показателей точности при обнаружении объектов [11].

Извлечение признаков — следующий за обнаружением лиц этап распознавания лиц. Входными данными для него является фотография выделенного лица. Выходными данными является дескриптор лица, описывающий характерные черты лица человека. В работе [14] выделены три группы методов извлечения признаков: холистические (holistic), локальные (local) и гибридные (hybrid).

*Холистические методы* используют все лицо в качестве входных данных, а затем проецируют его в небольшое подпространство или в плоскость корреляции. Основная идея холистических методов — обработка лица целиком и представление его изображения в виде матрицы пикселей. Холистические методы являются относительно быстрыми и простыми. Но при работе с очень большими базами данных могут появиться проблемы с точностью, так как вычислений в ходе распознавания очень много. Одним из ярких представителей таких методов является метод главных компонент (Principal Component Method, PCA) [23].

*Локальные методы* основываются на распознавании отдельных черт лица, без учета лица в целом. Основная цель локальных методов — выявить отличительные черты лица [14]. Данные методы нацелены на описание отдельных участков лица (например, для носа, губ, лба, глаз, нижней части подбородка) либо координат черт лица (например, внутренний угол глаза, внешний угол глаза), совокупность которых является дескриптором.

Наиболее популярным локальным методом является метод построения гистограмм направленных градиентов (Histogram of Oriented Gradients, HOG) [6]. Основная идея метода заключается в том, что изображение может быть описано распределением градиентов интенсивности или направления краев. Как правило, построение этих гистограмм происходит путем разбиения изображения на ячейки, и присвоения каждой ячейке гистограммы направлений градиентов для пикселей внутри ячейки, их комбинация и является дескриптором.

*Гибридный подход* основан на совместном использовании локального и холистического подходов, чтобы использовать их преимущества, которые могут дать лучшую производительность для систем распознавания лиц.

Этап идентификации включает в себя сравнение полученного вектора с уже имеющимися в базе данных распознаваемых лиц. Сравнение может вестись разными способами, наиболее популярным и простым является сравнение дескрипторов по евклидовой метрике [9]. Чем меньше расстояние между дескрипторами, тем больше соответствующие дескрипторам лица похожи друг на друга.

## 1.2. Системы учета посещаемости

В настоящее время существуют различные подходы к решению задачи распознавая лиц. Такие системы имеют похожую архитектуру, но алгоритмы и методы для решения каждого из трех этапов распознавания лиц выбираются разные.

В работе [13] обнаружение лиц и извлечение признаков выполняются с помощью нейронной сети YOLOv3 [22] и сервиса для распознавания лиц Microsoft Azure face [16] соответственно, что позволило авторам приблизиться к точности распознавания 100%. Однако рассматриваемая система была написана для ограниченного количества людей в классе, а

нейронные сети были обучены на 20 изображениях каждого распознаваемого. Системы, использующий такой подход не являются гибкими. Они требуют сбора данных для каждого студента и последующее обучение нейронной сети.

В работе [18] для обнаружения лиц на фотографии используется гистограмма направленных градиентов (HOG, Histogram of Oriented Gradients) [6], затем для извлечения дескриптора лиц используется переобученная сверточная нейронная сеть. Для последнего этапа идентификации используются метод опорных векторов (SVM, support vector machine) [5]. Точность распознавания данной системы составила 81%.

В работе [24] для обнаружения лиц и извлечения признаков используются модель МТСNN [32] и нейронная сеть ArcFace [7] соответственно. Для идентификации выбран метод опорных векторов SVM [5]. Достигнутая точность равна 89%. В данной системе применялась предобработка изображения: определение и выравнивание направления головы. В системах [18] и [24] для идентификации используется метод опорных векторов, являющийся более простым способом сравнения дескрипторов, чем нейронная сеть, однако так же требующий предварительного обучения. Использование подобных систем для больших и часто обновляемых баз данных распознаваемых лиц является затруднительным.

В работе [29] для этапа обнаружения лиц используются каскадные классификаторы на основе признаков Хаара, извлечение дескриптора осуществляется с помощью функции `face_encoding` из библиотеки `dlib` [8], полученные дескрипторы сравниваются с помощью евклидовой метрики. Для улучшения качества фотографии изображение переводилось в оттенки серого. Данная система имеет высокую скорость обработки и точность распознавания 83% для фотографий с фронтальными лицами при дневном свете и хорошем разрешении.

Другой подход используется в работе [17] и позволяет достигнуть 90% точности распознавания. Для этапа обнаружения лиц используется сверточная нейронная сеть, для извлечения дескриптора — метод главных компонент (Principal Component Method, PCA) [23]. Сравниваются дескрипторы с помощью вычисления косинусного расстояния Махаланобиса. Система имеет более высокую точность распознавания по сравнению с [29] благодаря использованию нейронной сети для обнаружения лиц и методам предобработки изображения, позволяющим нормализовать контраст, и уменьшить шум.

Системы [17, 29] схожи в отсутствии необходимости дополнительного обучения. Идентификация при таких подходах происходит путем извлечения вектора признака и сравнения его с уже имеющимися в базе данных.

На основе обзора аналогов можно выделить два вида систем для распознавания лиц. Первый — это системы с нейронными сетями или линейными классификаторами, заранее обученными на наборе лиц, которые в дальнейшем и будут распознаваться на других фотографиях или видеопотоке [13, 18, 24]. Второй — системы, нейронные сети которых обучены на стандартных наборах данных, а информация для идентификации лица берется из базы данных [17, 29]. Системы первого вида имеют более высокую точность распознавания. Системы второго вида менее точные, но позволяют быстро добавлять новых распознаваемых в базу данных без дообучения используемых нейронных сетей.

## 2. Реализация системы учета посещаемости

### 2.1. Архитектура системы

Общая архитектура системы представлена на рис. 1. Система включает в себя следующие основные модули: захват изображения, распознавание лиц и веб-интерфейс. Разра-



батываемая нами система учета посещаемости состоит из трех модулей: модуль захвата изображения, модуль распознавания лиц и веб-интерфейс.

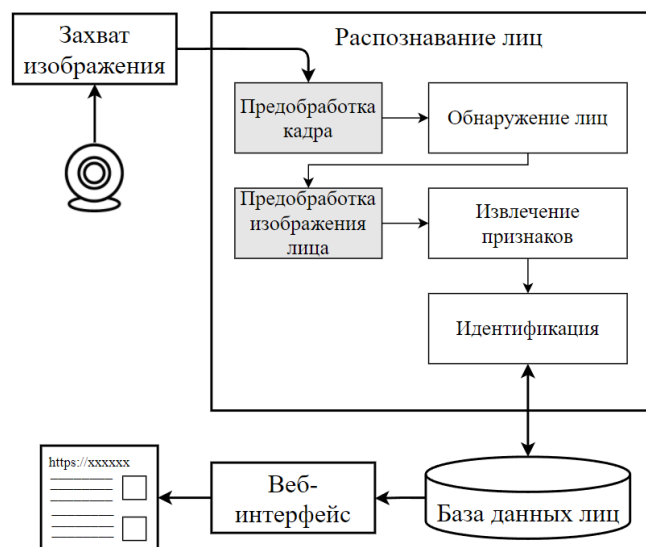


Рис. 1. Архитектура системы учета посещаемости

Модуль захвата изображения осуществляет подключение к веб-камере, расположенной в учебной аудитории, и получает от нее видеопоток. Захваченный с камеры видеопоток разбивается на кадры. Модуль распознавания лиц отвечает за осуществления всех этапов распознавания: обнаружение лиц, извлечение признаков и идентификация. По сравнению с общепринятой структурой распознавания лиц наша система дополнена этапами предобработки кадра и предобработки изображения лица, которые позволяют повысить точность распознавания. Модуль веб-интерфейса позволяет отобразить результат распознавания лиц на захваченном изображении в виде веб-страницы.

Работа системы кратко может быть описана следующим образом. Модуль захвата изображения каждый полученный кадр передает в модуль распознавания лиц. Модуль распознавания лиц имеет следующие основные процедуры: предобработки кадра, обнаружения лиц, предобработки изображения лица, извлечения признаков, идентификации. Вначале происходит предобработка кадра, что позволяет отфильтровать непригодные для распознавания фотографии и улучшить при возможности качество фотографии в целом (уменьшить шум, повысить резкость и др.). Далее к изображению применяется алгоритм обнаружения лиц. Каждое изображение выделенного лица снова проходит этап предобработки, после чего производится извлечение признаков лица. Такая двухэтапная проверка и улучшение качества изображения необходима, т.к. даже в случае хорошего качества кадра видеопотока качество изображения лиц на нем может быть недостаточным для распознавания. Например, кадр может быть высокого разрешения, но часть лиц на заднем плане могут быть очень маленькими и в результате извлеченная фотография такого лица будет иметь низкое разрешение, непригодное для распознавания. После извлечения признаков лица построенный дескриптор сравнивается с имеющимися в базе данных лиц для идентификации лица. Результат сравнения записывается в базу данных и предоставляется пользователю через веб-интерфейс.

В нашей работе для реализации системы учета посещаемости студентов мы использовали язык программирования Python совместно с библиотеками, обеспечивающими взаи-

модействие с искусственными нейронными сетями, Keras [12], Tensorflow [25] и библиотеками, позволяющими проводить предобработку изображений, dlib [8], OpenCV [19]. Для этапа обнаружения лиц использовалась предварительно обученная сверточная нейронная сеть Retina Face [7]. Сеть Retina Face обучена на открытом наборе данных WIDER Face (hard) [30]. Этот набор данных содержит данные с плохим освещением лица на фотографии, небольшим размером лиц, лицами с разными поворотами головы и др. Все эти проблемы характерны для системы учета посещаемости. На этапе извлечения признаков мы используем сверточную нейронную сеть ResNet [10], которая является одним из лидеров в области создания дескрипторов. Дескриптор, который мы получаем, описывает такие точки лица как нос, глаза, уголки губ и др. Дескриптор может содержать около 50 таких ключевых точек. Чем больше точек, тем более точной будет идентификация человека. Для сравнения дескрипторов используется евклидова метрика.

## 2.2. Повышение точности распознавания лиц

Имеется набор факторов, влияющих на точность распознавания. В работах [13, 17, 18, 24, 29] выделяются следующие факторы, которые снижают точность распознавания:

- низкое разрешение, шумы и плохая освещенность фотографии лица (рис. 2а–в);
- окклюзия — заслонение лица чем-либо: светом (рис. 2г), аксессуарами (головные уборы, очки, шарфы, борода, яркий макияж и др., рис. 2д) и поворотом головы (само-окклюзия, рис. 2е);
- наклон головы (рис. 2ж);
- эмоциональные выражения лица (рис. 2з);
- различная внешность людей (пол, телосложение, этническая принадлежность и др.).

Наилучшую точность распознавания лиц можно получить при отсутствии указанных проблемных факторов. Такое лицо далее будем называть эталонным (рис. 2и). Задача повышения точности распознавания сводится к задаче приближения фотографии лица к эталонному виду.

В настоящее время в области обработки изображений разработаны различные методы для улучшения качества изображения [4]. В нашей работе использовались следующие методы:

- размытие изображения для уменьшения шума (метод Medianblue из библиотеки OpenCV);
- повышение резкости изображения (применение нерезкой маскировки с использованием библиотеки OpenCV);
- преобразования гистограммы для увеличения яркости и выравнивания цветов (метод equalizeHist из библиотеки OpenCV).

Кроме этого каждое изображение лица приводится к единому размеру и подвергается нормализации, что необходимо для корректного сравнения дескрипторов лица. Нормализация представляет собой изменение параметров входных данных, чтобы они находились в пределах от 0 до 1.

Использование предварительной обработки для повышения качества требуется не всегда. Фотография может находиться в трех состояниях: непригодна для распознавания, требует предварительной обработки и подходит для распознавания. Таким образом, необходимо провести предварительную проверку качества полученной фотографии. Оценка качества изображения является количественной оценкой восприятия качества человеком и



Другие проблемы — это фотографии с окклюзией, с эмоциональным выражением лица и различным типом внешности. Эти проблемы значительно снижают общую точность распознавания [23]. Для решения каждого вида описанных проблем необходимы специфические подходы. Например, проблема распознавания лиц под разными углами может быть решена с помощью алгоритмов выравнивания лиц. Проблема распознавания человека с разными выражениями лица может быть решена с помощью алгоритма определения типа эмоций и получения «нейтрального» выражения лица. Большинство методов извлечения признаков нацелено на преодоление только одного вида проблемных факторов, например, только поворота/наклона головы или только окклюзии. Методы глубокого обучения, такие как сверточные нейронные сети состоят из нескольких уровней, каждый из которых может преодолевать один проблемный фактор, но все вместе могут решать комплекс проблем, снижающих точность распознавания [27]. В данной работе эти алгоритмы не рассматривались.

### 3. Вычислительные эксперименты

Для проверки эффективности разработанной системы учета посещаемости [3] нами были проведены вычислительные эксперименты. Под эффективностью понимаем точность распознавания лиц при наличии проблемных факторов (см. раздел 2.2). Для проведения вычислительных экспериментов была подготовлена база данных лиц, включающая фотографии и ФИО студентов одной группы (всего 25 человек). Были сделаны 80 фотографий группы при различных проблемных факторах. На каждой фотографии представлены от 3-х до 20-ти студентов из базы данных лиц. Фотографии были сгруппированы по наличию в них проблемных факторов, всего 9 групп. Для каждой из 80 фотографий вручную подсчитывалось количество присутствующих лиц ( $N_{faces}$ ) и на основе выдачи системы учета посещаемости определялось количество верно распознанных лиц ( $N_{truefaces}$ ). Точность распознавания лиц на одной фотографии была рассчитана по формуле:

$$accuracy = \frac{N_{truefaces}}{N_{faces}} \times 100\%. \quad (1)$$

Таблица 1. Вычислительные эксперименты

№	Низкое разрешение, шумы	Плохая освещенность	Окклюзии	Эмоциональное выражение	Самоокклюзия	Наклон головы	Точность (%)
1	—	—	—	—	—	—	100
2	—	—	+	+	—	—	86
3	+	—	—	—	—	—	81
4	+	+	+	+	—	—	75
5	+	+	—	—	—	—	70
6	+	—	+	+	+	+	66
7	+	+	—	—	+	+	61
8	—	—	—	—	+	+	60
9	+	+	+	+	+	+	58

Результаты экспериментов представлены в табл. 1. В колонке «Точность» приведено среднее арифметическое точности по всем фотографиям в группе с указанными проблемными факторами.

Для эталонных лиц система выдает точность 100%. При наклоне и повороте головы или эмоциональном выражении лица, точность снижается на 15–25%. Стоит отметить, что система имеет низкую точность при плохой освещенности и низком разрешении фотографии лица. При таких условиях точность снижается еще на 10%. Точность, которую мы получили для эталонных лиц, в среднем выше, чем у аналогичных систем [29] и [17] более чем на 10%.

## Заключение

В настоящее время учет посещаемости студентов является значимым фактором для повышения качества подготовки специалистов. Для автоматизации этого процесс хорошо подходит технология распознавания лиц. Мы рассмотрели современные методы для каждого этапа распознавания лиц: обнаружение лиц на фотографии, извлечение признаков и идентификация. Рассмотрели существующие системы учета посещаемости. Выделили два вида систем: системы, требующие дообучения при изменении базы данных распознаваемых лиц, и системы, предварительно обученные на стандартных базах данных лиц, не требующие дообучения. В нашей работе предложена архитектура системы учета посещаемости, не требующая дообучения при изменении базы данных лиц. Система состоит из трех основных модулей захвата изображения, распознавания лиц и веб-интерфейса. Свою системы мы дополнили процедурами предобработки изображения, которые, используя нашу методику, основанную на мере BREN, позволяют классифицировать качество изображения по трем классам: не пригодные для распознавания, пригодные для распознавания после предобработки и пригодные для распознавания. После классификации качества изображение может быть отброшено, если не пригодно для распознавания, подвергнуться предобработке (уменьшение шума, повышение резкости, увеличение яркости и выравнивание цветов) либо останется без изменений, если классифицировано как пригодное для распознавания. Система была реализована. Для проверки точности распознавания были проведены вычислительные эксперименты, подтверждающие эффективность нашего метода распознавания лиц по сравнению с аналогами.

*Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (грант № 20-07-00140) и Министерства науки и высшего образования РФ (государственное задание FENU-2020-0022).*

## Литература

1. Зарубин В.И., Бибалова С.А. Посещаемость занятий в вузе как фактор эффективности подготовки современных специалистов // Вестник Майкопского государственного технологического университета. 2017. № 2. С. 3–6.
2. Старовойтов В.В., Старовойтов В.Ф. Сравнительный анализ безэталонных мер оценки качества цифровых изображений // Системный анализ и прикладная информатика. 2017. № 1. С. 24–32. DOI: 10.21122/2309-4923-2017-1-24-32.
3. Струева А.Ю., Иванова Е.В. Свидетельство о регистрации программы для ЭВМ «Библиотека подпрограмм для распознавания лиц в системах контроля деятельности персонала» № RU 2021669594 от 01 декабря 2021 г., правообладатель: ФГАОУ ВО «ЮУрГУ»

(НИУ)».

4. Фурашев Д.А., Фаворская М.Н. Методы улучшения качества изображений // Актуальные проблемы авиации и космонавтики. 2010. Т. 1, № 6. С. 383–384.
5. Cortes C., Vapnik V. Support-vector networks // Machine Learning. 1995. No. 20. P. 273–297. DOI: 10.1007/BF00994018.
6. Dalal N., Triggs B. Histograms of oriented gradients for human detection // 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'05, San Diego, CA, USA, June 20–25, 2005. P. 1–8. DOI: 10.1109/CVPR.2005.177.
7. Deng J., Zafeririou S. Arcface for disguised face recognition // 17th IEEE/CVF International Conference on Computer Vision Workshop, ICCVW 2019, Seoul, October 27–28, 2019. P. 485–493. DOI: 10.1109/ICCVW.2019.00061.
8. Dlib C++ Library. Documentation. URL: <http://dlib.net/> (дата обращения: 21.08.2021).
9. Indyk P., Wagner T. Near-optimal (euclidean) metric compression // Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'17, Barcelona, Spain, January 16–19, 2017. P. 710–723. DOI: 10.1137/1.9781611974782.45.
10. He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016. P. 770–778. DOI: 10.1109/CVPR.2016.90.
11. Kalinovsky I., Spitsyn V. Review and Testing of Frontal Face Detectors // Computer Optics. 2016. Vol. 40, no. 1. P. 99–111. DOI: 10.18287/2412-6179-2016-40-1-99-111.
12. Keras. Documentation. URL: <https://keras.io/guides/> (дата обращения: 21.08.2021).
13. Khan S., Akram A., Usman N. Real Time Automatic Attendance System for Face Recognition Using Face API and OpenCV // Wireless Personal Communications. 2020. Vol. 113. P. 469–480. DOI: 10.1007/s11277-020-07224-2.
14. Kortli Y., Jridi M., Atri M. Face Recognition Systems: A Survey // Sensors. 2020. Vol. 20, no. 2. P. 1–36. DOI: 10.3390/s20020342.
15. Li H., Lin Z., Brandt J., Shen X., Hua G. Efficient boosted exemplar-based face detection // IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, June 23–28, 2014. P. 1843–1850. DOI: 10.1109/CVPR.2014.238.
16. Microsoft Azure face. Documentation. URL: <https://docs.microsoft.com/en-us/azure/cognitive-services/face/> (дата обращения: 01.09.2021)
17. Mothwa L., Tapamo J.-R., Mapayi T. Conceptual Model of the Smart Attendance Monitoring System Using Computer Vision // The 14th International Conference on Signal Image Technology & Internet Based Systems, SITIS 2019, Las Palmas de Gran Canaria, Spain, November 26–29, 2018. P. 229–234. DOI: 10.1109/SITIS.2018.00042.
18. Nurkhamid N., Setialana P., Jati H., Wardani R. Intelligent Attendance System with Face Recognition using the Deep Convolutional Neural Network Method // Journal of Physics: Conference Series. 2021. Vol. 1737, no. 1. P. 1–9. DOI: 10.1088/1742-6596/1737/1/012031.
19. OpenCV. Documentation. URL: <https://docs.opencv.org/> (дата обращения: 01.09.2021).
20. Patel S., Kumar R. Face Recognition based smart attendance system using IOT // International Journal of Computer Sciences and Engineering. 2018. Vol. 6, no. 5. P. 871–877. DOI: 10.26438/ijcse/v6i5.871877.

21. Pertuz S., Puig D., Garcia M.A. Analysis of focus measure operators for shape-from-focus // Pattern Recognition. 2013. Vol. 46, no. 5. P. 1415–1432. DOI: 10.1016/j.patcog.2012.11.0117.
22. Redmon J., Farhadi A. YOLOv3: An Incremental Improvement // CoRR. 2018. Vol. abs/1804.02767. URL: <http://arxiv.org/abs/1804.02767>
23. Samal D.B., Starovoitov V.V. A geometric approach to face recognition // Proceedings of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing, NSIP'99, Antalya, Turkey, June 20–23, 1999. P. 1–5.
24. Son N.T., Anh B.N., Ban T.Q. *et al.* Implementing CCTV-Based Attendance Taking Support System Using Deep Face Recognition: A Case Study at FPT Polytechnic College // Symmetry. 2020. Vol. 12, no. 307. DOI: 10.3390/sym12020307.
25. TensorFlow. URL: <https://www.tensorflow.org/> (дата обращения: 01.09.2021).
26. Viola P., Jones M.J. Robust Real-Time Face Detection // International Journal of Computer Vision. 2004. Vol. 57, no. 2. P. 137–154. DOI: 10.1023/B:VISI.0000013087.49260.fb.
27. Wang M., Deng W. Deep Face Recognition: A Survey // Neurocomputing. 2021. Vol. 429. P. 215–244. DOI: 10.1016/j.neucom.2020.10.081.
28. Winarno E., Hadikurniawati W., Nirwanto A.A. Multi-View Faces Detection Using Viola–Jones Method // Journal of Physics Conf. Series. 2018. Vol. 1114, no. 1. DOI: 10.1088/1742-6596/1114/1/012068.
29. Wu H., Cao Y., Wei H., Tian Z. Face Recognition Based on Haar Like and Euclidean Distance // Journal of Physics: Conference Series. 2021. Vol. 1813. P. 1–7. DOI: 10.1088/1742-6596/1813/1/012036.
30. Yang S., Luo P., Loy C.C., Tang X. WIDER FACE: A Face Detection Benchmark // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016. P. 5525–5533. DOI: 10.1109/CVPR.2016.596.
31. Zhu X., Ramanan D. Face detection, pose estimation, and landmark localization in the wild // IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16–21, 2012. P. 2879–2886. DOI: 10.1109/CVPR.2012.6248014.
32. Zhang K., Zhang Z., Li Z., Qiao Y. Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks // IEEE Signal Processing Letters. 2016. Vol. 3, no. 10. P. 1499–1503. DOI: 10.1109/LSP.2016.2603342.

Иванова Елена Владимировна, к.ф.-м.н., доцент, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Струева Анастасия Юрьевна, студент, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

## THE STUDENT ATTENDANCE CONTROL SYSTEM BASED ON FACE RECOGNITION

© 2021 E.V. Ivanova, A.Yu. Strueva

*South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)*

*E-mail: elena.ivanova@susu.ru, nastasy822@gmail.com*

Received: 26.09.2021

Currently, one of the significant factors for improving the quality of training of specialists is the control of student attendance. This process can be automated. The paper suggests an approach to building a student attendance control system based on face recognition technology, which allows you to identify many people at the same time without direct contact with them and without using expensive equipment. This approach is based on the convolutional neural networks RetinaFace and ResNet, selected based on the review of modern methods of facial recognition presented in the paper. The architecture of our attendance control system is complemented by image preprocessing procedures, which, according to our proposed method based on the BREN measure, check the image quality and, if necessary, apply algorithms to the image to reduce noise, sharpen, increase brightness and align colors. The results of computational experiments are presented, which have shown a higher efficiency of the proposed approach compared with analogues.

*Keywords: face recognition, attendance control system, convolutional neural network, RetinaFace, FaceNet, image preprocessing.*

### FOR CITATION

Ivanova E.V., Strueva A.Yu. The Student Attendance Control System Based on Face Recognition. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2021. Vol. 10, no. 4. P. 60–73. (in Russian) DOI: 10.14529/cmse210404.

*This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.*

### References

1. Zarubin V.I., Bibalova S.A. Attendance of classes at the university as a factor of the effectiveness of training modern specialists. Bulletin of the Maikop State Technological University. 2017. No. 2. P. 3–6. (in Russian)
2. Starovoitov V.V., Starovoitov V.F. Comparative analysis of standardfree measures for assessing the quality of digital images. System Analysis and Applied Informatic. 2017. No. 1, P. 24–32. (in Russian) DOI: 10.21122/2309-4923-2017-1-24-32.
3. Strueva A.Yu., Ivanova E.V. Certificate of registration of a computer program “Library of routines for face recognition in personnel activity control systems” № RU 2021669594, 01.12.2021, copyright holder: FSAEIHE SUSU (NRU). (in Russian)
4. Furashev D.A., Favorskaya M.Y. Methods for improving the quality of images. Actual problems of aviation and astronautics. 2010. Vol. 1, no. 6. P. 383–384. (in Russian)
5. Cortes C., Vapnik V. Support-vector networks. Machine Learning. 1995. No. 20. P. 273–297. DOI: 10.1007/BF00994018.
6. Dalal N., Triggs B. Histograms of oriented gradients for human detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR’05, San



- Diego, CA, USA, June 20–25, 2005. P. 1–8. DOI: 10.1109/CVPR.2005.177.
7. Deng J., Zafeririou S. Arcface for disguised face recognition. 17th IEEE/CVF International Conference on Computer Vision Workshop, ICCVW 2019, Seoul, October 27–28, 2019. P. 485–493. DOI: 10.1109/ICCVW.2019.00061.
  8. Dlib C++ Library. Documentation. URL: <http://dlib.net/> (accessed: 21.08.2021).
  9. Indyk P., Wagner T. Near-optimal (euclidean) metric compression. Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'17, Barcelona, Spain, January 16–19, 2017. P. 710–723. DOI: 10.1137/1.9781611974782.45.
  10. He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016. P. 770–778. DOI: 10.1109/CVPR.2016.90
  11. Kalinovsky I., Spitsyn V. Review and Testing of Frontal Face Detectors. Computer Optics. 2016. Vol. 40, no. 1. P. 99–111. DOI: 10.18287/2412-6179-2016-40-1-99-111.
  12. Keras. Documentation. URL: <https://keras.io/guides/> (accessed: 21.08.2021).
  13. Khan S., Akram A., Usman N. Real Time Automatic Attendance System for Face Recognition Using Face API and OpenCV. Wireless Personal Communications. 2020. Vol. 113. P. 469–480. DOI: 10.1007/s11277-020-07224-2.
  14. Kortli Y., Jridi M., Atri M. Face Recognition Systems: A Survey. Sensors. 2020. Vol. 20, no. 2. P. 1–36. DOI: 10.3390/s20020342.
  15. Li H., Lin Z., Brandt J., Shen X., Hua G. Efficient boosted exemplar-based face detection. IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, June 23–28, 2014. P. 1843–1850. DOI: 10.1109/CVPR.2014.238.
  16. Microsoft Azure face. Documentation. URL: <https://docs.microsoft.com/en-us/azure/cognitive-services/face/> (accessed: 01.09.2021).
  17. Mothwa L., Tapamo J.-R., Mapayi T. Conceptual Model of the Smart Attendance Monitoring System Using Computer Vision. The 14th International Conference on Signal Image Technology & Internet Based Systems, SITIS 2019, Las Palmas de Gran Canaria, Spain, November 26–29, 2018. P. 229–234. DOI: 10.1109/SITIS.2018.00042.
  18. Nurkhamid N., Setialana P., Jati H., Wardani R. Intelligent Attendance System with Face Recognition using the Deep Convolutional Neural Network Method. Journal of Physics: Conference Series. 2021. Vol. 1737, no. 1. P. 1–9. DOI: 10.1088/1742-6596/1737/1/012031.
  19. OpenCV. Documentation. URL: <https://docs.opencv.org/> (accessed: 01.09.2021).
  20. Patel S., Kumar R. Face Recognition based smart attendance system using IO. International Journal of Computer Sciences and Engineering. 2018. Vol. 6, no. 5. P. 871–877. DOI: 10.26438/ijcse/v6i5.871877.
  21. Pertuz S., Puig D., Garcia M.A. Analysis of focus measure operators for shape-from-focus. Pattern Recognition. 2013. Vol. 46. no. 5. P. 1415–1432. DOI: 10.1016/j.patcog.2012.11.0117.
  22. Redmon J., Farhadi A. YOLOv3: An Incremental Improvement. CoRR. 2018. Vol. abs/1804.02767. URL: <http://arxiv.org/abs/1804.02767>
  23. Samal D.B., Starovoitov V.V. A geometric approach to face recognition. Proceedings of the IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing, NSIP'99, Antalya, Turkey, June 20–23, 1999. P. 1–5.

24. Son N.T., Anh B.N., Ban T.Q. *et al.* Implementing CCTV-Based Attendance Taking Support System Using Deep Face Recognition: A Case Study at FPT Polytechnic College. *Symmetry*. 2020. Vol. 12, no. 307. DOI: 10.3390/sym12020307.
25. TensorFlow. URL: <https://www.tensorflow.org/> (accessed: 01.09.2021).
26. Viola P., Jones M.J. Robust Real-Time Face Detection. *International Journal of Computer Vision*. 2004. Vol. 57, no. 2. P. 137–154. DOI: 10.1023/B:VISI.0000013087.49260.fb.
27. Wang M., Deng W. Deep Face Recognition: A Survey. *Neurocomputing*. 2021. Vol. 429. P. 215–244. DOI: 10.1016/j.neucom.2020.10.081.
28. Winarno E., Hadikurniawati W., Nirwanto A.A. Multi-View Faces Detection Using Viola–Jones Method. *Journal of Physics Conf. Series*. 2018. Vol. 1114, no. 1. DOI: 10.1088/1742-6596/1114/1/012068.
29. Wu H., Cao Y., Wei H., Tian Z. Face Recognition Based on Haar Like and Euclidean Distance. *Journal of Physics: Conference Series*. 2021. Vol. 1813. P. 1–7. DOI: 10.1088/1742-6596/1813/1/012036.
30. Yang S., Luo P., Loy C.C., Tang X. WIDER FACE: A Face Detection Benchmark. 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016. P. 5525–5533. DOI: 10.1109/CVPR.2016.596.
31. Zhu X., Ramanan D. Face detection, pose estimation, and landmark localization in the wild. IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16–21, 2012. P. 2879–2886. DOI: 10.1109/CVPR.2012.6248014.
32. Zhang K., Zhang Z., Li Z., Qiao Y. Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks. *IEEE Signal Processing Letters*. 2016. Vol. 3, no. 10. P. 1499–1503. DOI: 10.1109/LSP.2016.2603342.

## СВЕДЕНИЯ ОБ ИЗДАНИИ

Научный журнал «Вестник ЮУрГУ. Серия «Вычислительная математика и информатика» основан в 2012 году.

Учредитель — Федеральное государственное автономное образовательное учреждение высшего образования «Южно-Уральский государственный университет» (национальный исследовательский университет).

Главный редактор — Л.Б. Соколинский.

Свидетельство о регистрации ПИ ФС77-57377 выдано 24 марта 2014 г. Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций.

Журнал включен в Реферативный журнал и Базы данных ВИНИТИ; индексируется в библиографической базе данных РИНЦ. Журнал размещен в открытом доступе на Всероссийском математическом портале MathNet. Сведения о журнале ежегодно публикуются в международной справочной системе по периодическим и продолжающимся изданиям «Ulrich's Periodicals Directory».

Решением Президиума Высшей аттестационной комиссии Министерства образования и науки Российской Федерации журнал включен в «Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук» по научным специальностям и соответствующим им отраслям науки: 05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей (физико-математические науки), 05.13.17 – Теоретические основы информатики (физико-математические науки).

Подписной индекс научного журнала «Вестник ЮУрГУ», серия «Вычислительная математика и информатика»: 10244, каталог «Пресса России». Периодичность выхода — 4 выпуска в год.

Адрес редакции, издателя: 454080, г. Челябинск, проспект Ленина, 76, Издательский центр ЮУрГУ, каб. 32.

## ПРАВИЛА ДЛЯ АВТОРОВ

1. Правила подготовки рукописей и пример оформления статей можно загрузить с сайта серии <http://vestnikvmi.susu.ru>. **Статьи, оформленные без соблюдения правил, к рассмотрению не принимаются.**
2. Адрес редакционной коллегии научного журнала «Вестник ЮУрГУ», серия «Вычислительная математика и информатика»:  
Россия 454080, г. Челябинск, пр. им. В.И. Ленина, 76, ЮУрГУ, кафедра СП,  
ответственному секретарю Цымблеру М.Л.
3. Адрес электронной почты редакции: [vestnikvmi@susu.ru](mailto:vestnikvmi@susu.ru)
4. **Плата с авторов за публикацию рукописей не взимается, и гонорары авторам не выплачиваются.**

ВЕСТНИК  
ЮЖНО-УРАЛЬСКОГО  
ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА  
Серия  
«ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА И ИНФОРМАТИКА»  
Том 10, № 4  
2021

16+

Техн. редактор *А.В. Миних*

Издательский центр Южно-Уральского государственного университета

Подписано в печать 30.11.2021. Дата выхода в свет 15.12.2021. Формат 60×84 1/8. Печать цифровая.  
Усл. печ. л. 8,83. Тираж 500 экз. Заказ 343/404. Цена свободная.

Отпечатано в типографии Издательского центра ЮУрГУ.  
454080, г. Челябинск, проспект Ленина, 76.