УДК 004.272.25, 004.421, 004.032.24 DOI: 10.14529/cmse230304

ОБНАРУЖЕНИЕ АНОМАЛИЙ ВРЕМЕННОГО РЯДА НА ОСНОВЕ ТЕХНОЛОГИЙ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ И НЕЙРОННЫХ СЕТЕЙ

© 2023 Я.А. Краева

Южно-Уральский государственный университет
(454080 Челябинск, пр. им. В.И. Ленина, д. 76)
E-mail: kraevaya@susu.ru

Поступила в редакцию: 20.05.2023

В статье рассмотрена задача поиска аномальных подпоследовательностей временного ряда, решение которой в настоящее время востребовано в широком спектре предметных областей. Предложен новый метод обнаружения аномальных подпоследовательностей временного ряда с частичным привлечением учителя. Метод базируется на концепциях диссонанса и сниппета, которые формализуют соответственно понятия аномальных и типичных подпоследовательностей временного ряда. Предложенный метод включает в себя нейросетевую модель, которая определяет степень аномальности входной подпоследовательности ряда, и алгоритм автоматизированного построения обучающей выборки для этой модели. Нейросетевая модель представляет собой сиамскую нейронную сеть, где в качестве подсети предложено использовать модификацию модели ResNet. Для обучения модели предложена модифицированная функция контрастных потерь. Формирование обучающей выборки выполняется на основе репрезентативного фрагмента ряда, из которого удаляются диссонансы, маломощные сниппеты со своими ближайшими соседями и выбросы в рамках каждого сниппета, трактуемые соответственно как аномальная, нетипичная деятельность субъекта и шумы. Вычислительные эксперименты на временных рядах из различных предметных областей показывают, что предложенная модель по сравнению с аналогами показывает в среднем наиболее высокую точность обнаружения аномалий по стандартной метрике VUS-PR. Обратной стороной высокой точности метода является большее по сравнению с аналогами время, которое затрачивается на обучение модели и распознавание аномалии. Тем не менее, в приложениях интеллектуального управления отоплением зданий метод обеспечивает быстродействие, достаточное для обнаружения аномальных подпоследовательностей в режиме реального времени.

Ключевые слова: временной ряд, поиск аномалий, диссонанс, сниппет, сиамская нейронная сеть.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Краева Я.А. Обнаружение аномалий временного ряда на основе технологий интеллектуального анализа данных и нейронных сетей // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 3. С. 50–71. DOI: 10.14529/cmse230304.

Введение

В настоящее время разработка эффективных моделей, методов и алгоритмов поиска аномалий временного ряда остается одной из наиболее актуальных исследовательских проблем [1]. Поиск аномалий требуется в широком спектре предметных областей, связанных с обработкой временных рядов: Интернет вещей [2], умное управление зданиями [3] и городом [4], персональная медицина [5] и др. При этом целью поиска может быть точечная аномалия (одиночный выброс) или аномальная подпоследовательность (непрерывный интервал элементов ряда, не все из которых являются выбросами). Случай аномальных подпоследовательностей является наиболее востребованным на практике и сложным для поиска, поскольку среди прочих параметров требует учета всевозможных длин аномалии [1].

Методы поиска аномальных подпоследовательностей временного ряда разделяют на три группы [6, 7]: поиск с учителем (supervised), поиск без учителя (unsupervised) и поиск с частичным привлечением учителя (semi-supervised).

Методы поиска аномалий *с учителем* моделируют нормальное и девиантное поведение во временном ряде и включают в себя этап обучения, после которого возможно их применение во временных рядах, не известных модели. Методы данной группы требуют для обучения предварительно размеченный экспертом или специализированной программой временной ряд (ряды), где подпоследовательности имеют одну из двух меток: «норма» или «аномалия». По своей природе методы поиска аномалий с учителем ограничены в своей способности обнаруживать аномалии, не задействованные на этапе обучения, и поэтому в настоящее время они редко применяются на практике [6].

Методы поиска аномалий *без учителя* не требуют предварительных знаний о временном ряде и не включают в себя этап обучения. Указанные методы основываются на предположениях о свойствах, которыми обладают аномальные подпоследовательности: они встречаются реже, имеют иную форму, происходят из другого распределения вероятностей и др.

Методы *с частичным привлечением учителя* включают в себя этап обучения, на котором пытаются изучить только нормальное поведение временного ряда, который фигурирует в качестве обучающей выборки модели. Модель, будучи примененной к тестовому временному ряду, помечает как аномальные подпоследовательности, которые не соответствуют нормальному поведению.

В данной статье предлагается новый метод обнаружения аномальных подпоследовательностей временного ряда с частичным привлечением учителя. Метод включает в себя нейросетевую модель, которая определяет степень аномальности входной подпоследовательности, и алгоритм автоматизированного построения обучающей выборки для этой модели. Нейросетевая модель представляет собой сиамскую нейронную сеть (Siamese Neural Network) [8], где в качестве подсети фигурирует модификация нейросетевой модели ResNet [9], и для обучения которой предлагается модифицированная функция потерь. Алгоритм построения обучающей выборки для нейросетевой модели предполагает очистку репрезентативного ряда от аномальных подпоследовательностей, отражающих аномальные и нетипичные активности субъекта, для поиска которых применяются понятия диссонанса [10] и сниппета [11] соответственно.

Остаток текста статьи организован следующим образом. Раздел 1 содержит краткий обзор работ по тематике исследования. В разделе 2 приводятся формальные определения базовых понятий. В разделе 3 представлен метод обнаружения аномалий временного ряда в реальном времени, основанный на совместном применении технологий нейронных сетей и интеллектуального анализа данных. Раздел 4 описывает результаты вычислительных экспериментов по исследованию эффективности предложенного метода. Заключение подводит итоги исследования.

1. Обзор работ

В недавно опубликованных обзорных статьях о методах поиска аномалий временного ряда [6, 12] суммарно рассматривается около ста различных методов, в том числе более 25 методов с частичным привлечением учителя. Поэтому в данном разделе кратко рассмотрено лишь малое число основных подходов к поиску аномалий, некоторые из которых далее были задействованы в вычислительных экспериментах данного исследования.

В группу методов поиска аномалий без учителя входят поиск диссонансов [13] на основе матричного профиля временного ряда [14], алгоритмы DRAG [10], MERLIN [15], DAMP [16], метод NormA [17, 18] и др.

Как указывалось выше, методы поиска аномалий с учителем редко применяются на практике [6], и поэтому можно привести лишь три относительно недавние разработки, входящие в данную группу методов: MultiHMM [19], HIF [20] и NF [21].

Типичными представителями группы методов поиска аномалий с частичным привлечением учителя являются следующие разработки. Метод LSTM-AD [22] выполняет обнаружение аномалий в многомерных временных рядах на основе применения нейронных сетей долгой краткосрочной памяти (LSTM, Long Short-Term Memory). Метод предполагает, что размеченные данные распределяются на следующие группы. Нормальные подпоследовательности делятся на четыре группы: обучающая выборка (s_N), две валидационные выборки (v_{N1} и v_{N2}) и тестовая выборка (t_N). Аномальные подпоследовательности делятся на две группы: валидационная (v_A) и тестовая выборки (t_A). Метод использует двухступенчатую схему «предсказание-детекция»: сперва модель на основе многослойной сети LSTM предсказывает значения временного ряда, а затем вычисляется распределение ошибок предсказания, с помощью которого обнаруживаются аномалии. Многослойная сеть LSTM организуется следующим образом. Во входном слое для каждой из m размерностей ряда имеется один нейрон, $d \times \ell$ нейронов в выходном слое таких, что на каждое из ℓ предсказанных значений для каждой из d размерностей имеется один нейрон (где d, ℓ — параметры и $1 \leqslant d \leqslant m$). Нейроны скрытого слоя LSTM являются полносвязными, что реализовано с помощью рекуррентных связей. Несколько LSTM слоев (обычно два) объединяются в стек таким образом, чтобы каждый нейрон в скрытом слое LSTM снизу был полностью соединен с каждым нейроном в скрытом слое LSTM над ним посредством прямых соединений. Обучение описанной модели выполняется на выборке s_N, выборка v_{N1} используется для раннего останова обучения при подборе весов нейросети. Фаза детекции выполняется следующим образом. Для каждой из выбранных d размерностей ℓ раз выполняется предсказание ℓ значений. Далее применяется вектор ошибок, элемент которого представляет собой разность между реальным и предсказанным значениями. Модель, обученная на выборке s_N, используется для вычисления векторов ошибок для последовательностей валидационной и тестовой выборок. Векторы ошибок моделируются таким образом. Векторы ошибок для элементов из выборки v_{N1} используются для оценки параметров распределения с использованием оценки максимального правдоподобия. Подпоследовательность классифицируется как аномалия, если функция оценка максимального правдоподобия меньше наперед заданного параметра τ, иначе она помечается как «норма». При этом выборки v_{N2} и v_A применяются для определения τ посредством максимизации значения F-меры, когда аномальные подпоследовательности считаются принадлежащими положительному классу, а нормальные — напротив, отрицательному.

Метод DeepAnT [23] позволяет обнаруживать одиночные выбросы и аномальные подпоследовательности временного ряда как в онлайн, так и в офлайн режиме. DeepAnT использует не содержащие разметку временные ряды для изучения распределения данных, которое затем используется для прогнозирования нормального поведения временного ряда. DeepAnT состоит из двух модулей: предсказателя и детектора аномалий временного ряда. Модуль предсказания использует глубокую сверточную нейронную сеть для прогнозирования будущего значения ряда на определенном горизонте, используя в качестве контекста

окно предыдущих значений ряда. Нейронная сеть включает в себя два сверточных слоя с размером ядра 32 и Линейным выпрямителем (ReLU, Rectified linear unit) в качестве функции активации. К выходу каждого из слоев применяется операция подвыборки по максимальному значению (MaxPooling). Последним слоем нейросети является полносвязный. В качестве функции потерь пр обучении нейросети применяется средняя абсолютная ошибка (MAE, Mean Absolute Error). Полученное прогнозное значение затем передается детектору аномалий, который отвечает за разметку значения как нормального или аномального. DeepAnT допускает обучение на временных рядах, из которых не удаляются выбросы и аномальные подпоследовательности.

2. Теоретический базис

2.1. Временной ряд и подпоследовательность

Временной ряд T представляет собой последовательность вещественных значений, взятых в хронологическом порядке:

$$T = \{t_i\}_{i=1}^{n}, \quad t_i \in \mathbb{R}. \tag{1}$$

Число n называется длиной ряда и обозначается $|T|$.

Подпоследовательность $T_{i,\,m}$ временного ряда T представляет собой непрерывный промежуток из m элементов ряда, начиная с позиции i:

$$T_{i,m} = \{t_k\}_{k=i}^{i+m-1}, \quad 3 \leqslant m \ll n, \quad 1 \leqslant i \leqslant n - m + 1. \tag{2}$$

Множество всех подпоследовательностей ряда T, имеющих длину m, обозначим как S_T^m.

2.2. Диссонансы

Подпоследовательности $T_{i,\,m}$ и $T_{j,\,m}$ ряда T считаются *не пересекающимися*, если $|i - j| \geqslant m$. Некая подпоследовательность ряда, не пересекающаяся с данной подпоследовательностью C, обозначается как M_C.

Подпоследовательность D ряда T является *диссонансом* [10], если

$$\min_{M_D \in T} \big(\mathrm{Dist}(D, M_D)\big) \geqslant r, \tag{3}$$

где $\mathrm{Dist}(\cdot\,,\cdot\,)$ — неотрицательная симметричная функция расстояния, r — порог расстояния (параметр). Иными словами, некая подпоследовательность ряда является диссонансом, если ее ближайший сосед (ближайшая и не пересекающаяся с ней подпоследовательность) находится на расстоянии не менее чем r. Для поиска диссонансов в качестве функции расстояния могут быть выбраны евклидова метрика [10], квадрат z-нормализованного евклидова расстояния [24] и др.

2.3. Сниппеты

Сниппеты [11] временного ряда представляют собой подпоследовательности, выражающие типичные активности некоего субъекта, деятельность которого описывает данный ряд. Формальное определение сниппетов выглядит следующим образом.

Пусть имеется временной ряд T и задана длина подпоследовательности m ($m \ll n$). Разобьем ряд на не пересекающиеся *сегменты* длины m, без ограничения общности считая,

что n кратно m. Рассмотрим множество сегментов Seg_T^m:

$$Seg_T^m = \{Seg_i\}_{i=1}^{n/m}, \quad Seg_i = T_{m\cdot(i-1)+1,\,m}. \tag{4}$$

Сниппеты представляют собой непустое подмножество Seg_T^m из K сегментов, где K $(1 \leqslant K \leqslant n/m)$ — параметр, отражающий количество активностей субъекта, интересующее исследователя. Обозначим множество сниппетов ряда T, имеющих длину m, как C_T^m:

$$C_T^m = \{C_i\}_{i=1}^{K}, \quad C_i \in Seg_T^m. \tag{5}$$

С каждым сниппетом ассоциированы следующие атрибуты: индекс, профиль, ближайшие соседи и мощность (значимость) данного сниппета. *Индекс сниппета* $C_i \in C_T^m$ обозначается как $C_i.index$ и представляет собой номер j сегмента Seg_j, которому соответствует подпоследовательность ряда $T_{m\cdot(j-1)+1,\,m}$.

Профиль сниппета, обозначаемый как $C_i.profile$, представляет собой вектор MPdist-расстояний [25] между данным сниппетом и подпоследовательностями ряда:

$$C_i.profile = \{d_k\}_{k=1}^{n-m+1}, \quad d_k = \mathrm{MPdist}(C_i,\, T_{i,\,m}). \tag{6}$$

Множество ближайших соседей сниппета $C_i \in C_T^m$ обозначается как $C_i.NN$ и содержит подпоследовательности ряда, которые более близки данному сниппету, чем другим сегментам ряда, в смысле расстояния MPdist:

$$C_i.NN = \{T_{j,\,m} \mid Seg_{C_i.index} = \arg\min_{1\leqslant q\leqslant n/m} \mathrm{MPdist}(T_{j,\,m},\, Seg_q),\ 1 \leqslant j \leqslant n-m+1\}. \tag{7}$$

Мощность сниппета $C_i \in C_T^m$ обозначается как $C_i.frac$ и вычисляется как доля мощности множества ближайших соседей сниппета от общего количества подпоследовательностей ряда, имеющих длину m, при этом сниппеты упорядочиваются по убыванию их мощности:

$$C_i.frac = \frac{|\,C_i.NN|}{n-m+1}. \tag{8}$$

$$\forall C_i, C_j \in C_T^m :\ i < j \iff C_i.frac \geqslant C_j.frac. \tag{9}$$

Расстояние $\mathrm{MPdist}(\cdot\,,\cdot\,)$ между подпоследовательностями A и B $(|A| = |B| = m)$ определяется следующим образом [25]. Фиксируем параметр ℓ $(\lceil 0.3m\rceil \leqslant \ell \leqslant \lceil 0.8m\rceil)$, который отражает длину семантически значимого непрерывного промежутка точек подпоследовательности. Вычисление MPdist предполагает последовательное выполнение следующих операций: 1) вычисление матричных профилей A и B, взятых в указанном и обратном порядке; 2) конкатенация вычисленных профилей; 3) упорядочение элементов полученного временного ряда по возрастанию; 4) взятие в качестве ответа k-го элемента результирующего ряда. Формальная запись выглядит следующим образом:

$$\mathrm{MPdist}_\ell(A,B) = \mathrm{AscSort}\big(P_{ABBA}\big)(k), \quad P_{ABBA} = P_{AB} \bullet P_{BA}, \tag{10}$$

где $\mathrm{AscSort}(\cdot\,)$ — операция упорядочивания элементов последовательности по возрастанию, символ $\bullet$ обозначает операцию конкатенации, k $(0 < k < m)$ — задаваемый аналитиком параметр (типичное значение $k = \lceil 0.1m\rceil$).

Матричным профилем [26] рядов A и B для длины называется ряд P_{AB}, i-м элементом которого является расстояние между i-й подпоследовательностью ряда A, имеющей длину ℓ, и ее ближайшим соседом в ряде B:

$$P_{AB} = \{\mathrm{ED}^2_{\mathrm{norm}}(A_{i,\ell}, B_{j,\ell})\}_{i=1}^{m-\ell+1}, \quad B_{j,\ell} = \arg \min_{1 \leqslant q \leqslant m-\ell+1} \mathrm{ED}^2_{\mathrm{norm}}(A_{i,\ell},\, B_{q,\ell}), \tag{11}$$

где функция $\mathrm{ED}^2_{\mathrm{norm}}(\cdot,\cdot)$ означает квадрат евклидова расстояния между z-нормализованными подпоследовательностями. Аналогичным образом определяется матричный профиль рассматриваемых рядов, взятых в порядке B и A, и обозначается как P_{BA}.

3. Метод обнаружения аномалий во временных рядах

В данном разделе представлен новый метод обнаружения аномалий во временном ряде в режиме реального времени, получивший название DiSSiD (Discord, Snippet, and Siamese Neural Networket-based Detector of anomalies). Метод включает в себя следующие компоненты: нейросетевая модель, построенная на основе сиамской нейронной сети, и алгоритм подготовки обучающей выборки для указанной модели, — описанные ниже в разделах 3.1 и 3.2 соответственно.

3.1. Нейросетевая модель

3.1.1. Архитектура модели

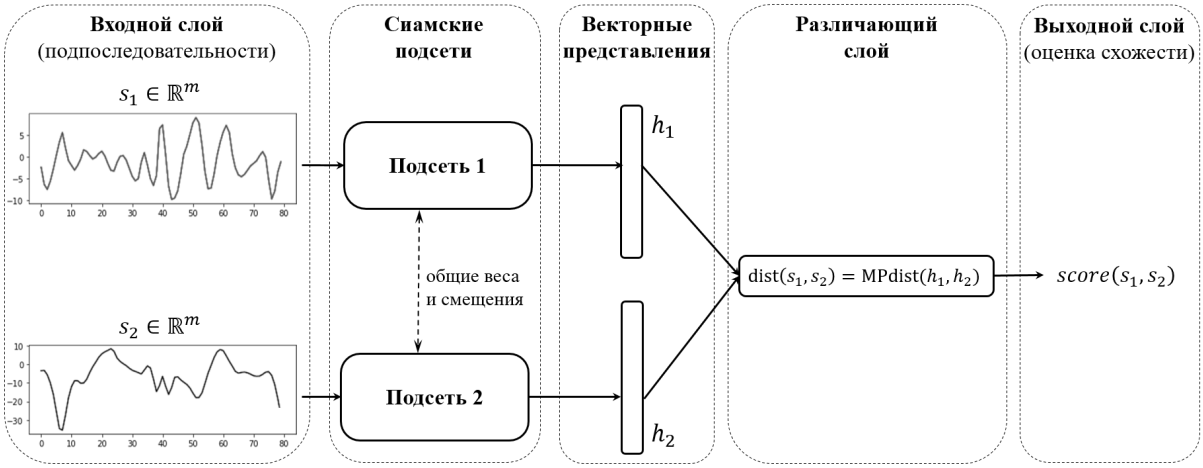


Рис. 1. Нейросетевая модель DiSSiD

Разработанная нейросетевая модель представлена на рис. 1. DiSSiD представляет собой сиамскую нейронную сеть (Siamese Neural Network, SNN) [8]. SNN объединяет в себе две подсети, которые имеют одинаковые архитектуру, конфигурацию (количество слоев, число нейронов в каждом слое, размерность входного и выходного слоев, функции активации и др.), а также наборы весов и смещений, полученных в результате обучения. Каждая из указанных подсетей формирует векторное представление (embedding) поданной на вход подпоследовательности, а на выходе модель выдает MPdist-расстояние между сформированными векторными представлениями. В качестве подсети фигурирует модификация нейросетевой модели ResNet [9]. Архитектура SNN по сравнению с традиционными нейросетевыми моделями лучше приспособлена к обучению в случае дисбаланса классов и позволяет добавить новый класс в уже развернутую модель без ее повторного обучения [8]. Архитек-

тура ResNet в настоящее время является одним из наиболее эффективных средств решения проблемы затухания градиента (vanishing gradient) при обучении нейросетевой модели [9].

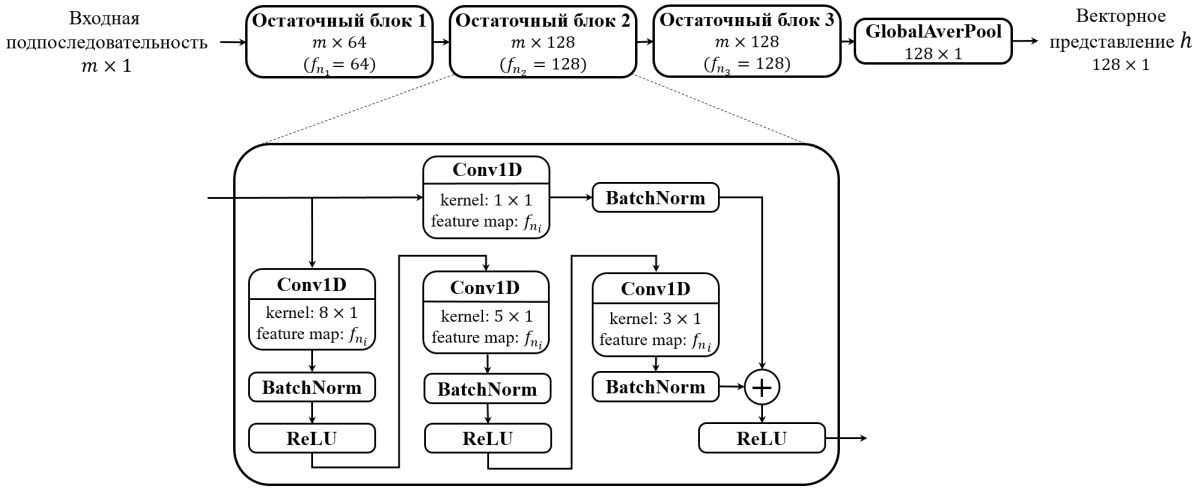


Рис. 2. Архитектура подсети ResNet

Подсеть на основе ResNet имеет следующую архитектуру (см. рис. 2). На входной слой поступает подпоследовательность временного ряда, имеющая длину m. Далее подсеть включает в себя три одинаковых остаточных блока (residual block) и за ними слой глобальной усредняющей агрегации (GlobalAveragePooling), формирующий векторное представление. Размерность итогового векторного представления определяется количеством карт признаков последнего слоя в последнем остаточном блоке.

Каждый остаточный блок включает в себя три сверточных слоя, на которых применяются фильтры (ядра) с размерами 8×1, 5×1 и 3×1 соответственно. Каждый сверточный слой чередуется со слоем пакетной нормализации (batch normalization) [27], к которому применяется функция активации Линейный выпрямитель (ReLU, Rectified linear unit). Пакетная нормализация преобразует набор входных данных таким образом, что его математическое ожидание обращается в ноль, а дисперсия — в единицу, и предназначена для ускорения сходимости обучения.

После прохождения трех сверточных слоев остаточный блок выдает карты признаков (feature maps): первый блок — 64 карты, остальные два блока — по 128 карт. Далее выполняется сложение входа остаточного блока, пропущенного через сверточный слой с ядром размера 1×1, с выходом этого остаточного блока. Однако входы не могут добавляться напрямую к выходам остаточного блока, поскольку они не имеют одинаковых размеров. Данный прием применяется как средство преодоления проблемы затухающего градиента (vanishing gradient) [28] между внутренними слоями нейронной сети.

3.1.2. Обучение модели

Для обучения модели DiSSiD из заданного временного ряда формируется обучающая выборка, определяемая следующим образом:

$$\begin{aligned} \mathcal{L} =& \mathcal{L}_{\text{true}} \cup \mathcal{L}_{\text{false}} \setminus \mathcal{L}_{\text{anomaly}} \\ \mathcal{L}_{\text{true}} =& \{< s_1; s_2; 1 > |\ \ s_1, s_2 \in C_i.NN, \quad 1 \leqslant i \leqslant K\} \\ \mathcal{L}_{\text{false}} =& \{< s_1; s_2; 0 > |\ \ s_1 \in C_i.NN,\, s_2 \in C_j.NN, \quad i \neq j,\, 1 \leqslant i, j \leqslant K\}, \end{aligned} \tag{12}$$

где множество $\mathcal{L}_{\text{true}}$ включает в себя пары подпоследовательностей, входящих в множество ближайших соседей одного и того же сниппета, в множество $\mathcal{L}_{\text{false}}$ — пары подпоследовательностей из множеств ближайших соседей разных сниппетов, а множество $\mathcal{L}_{\text{anomaly}}$ содержит аномальные подпоследовательности, не включаемые в обучающую выборку. Алгоритм очистки исходного временного ряда от аномальных подпоследовательностей и формирования обучающей выборки рассмотрен далее в разделе 3.2.

Для обучения модели DiSSiD предлагается следующая модифицированная функция контрастных потерь (contrastive loss) [29]:

$$L(s_1, s_2, \delta_{s_1 s_2}) = \delta_{s_1 s_2} \cdot \text{MPdist}(h_1, h_2) + (1 - \delta_{s_1 s_2})\big(\max(\tau - \text{MPdist}(h_1, h_2), 0)\big)^2, \qquad (13)$$

где s_1 и s_2, h_1 и h_2 — исходные подпоследовательности и их векторные представления соответственно; кронекериан $\delta_{s_1 s_2}$ принимает значение 1, если исходные подпоследовательности являются ближайшими соседями одного и того же сниппета, и 0 в противном случае; τ — минимальное расстояние MPdist между векторными представлениями исходных подпоследовательностей, являющихся ближайшими соседями разных сниппетов (параметр модели). Указанная функция потерь обеспечивает обучение модели, в результате которого похожие подпоследовательности исходного ряда получат векторные представления, отстоящие друг от друга в смысле расстояния MPdist не более чем на τ, а непохожие — более чем на τ соответственно.

Перед обучением элементы множества $\mathcal{L}$ случайным образом разделяются на два не пересекающихся подмножества: обучающую и валидационную выборки $\mathcal{L}_{\text{train}}$ и $\mathcal{L}_{\text{valid}}$, используемые для обучения модели и настройки ее гиперпараметров соответственно. Мощности указанных выборок находятся в традиционном соотношении 80% и 20% соответственно.

3.1.3. Применение модели

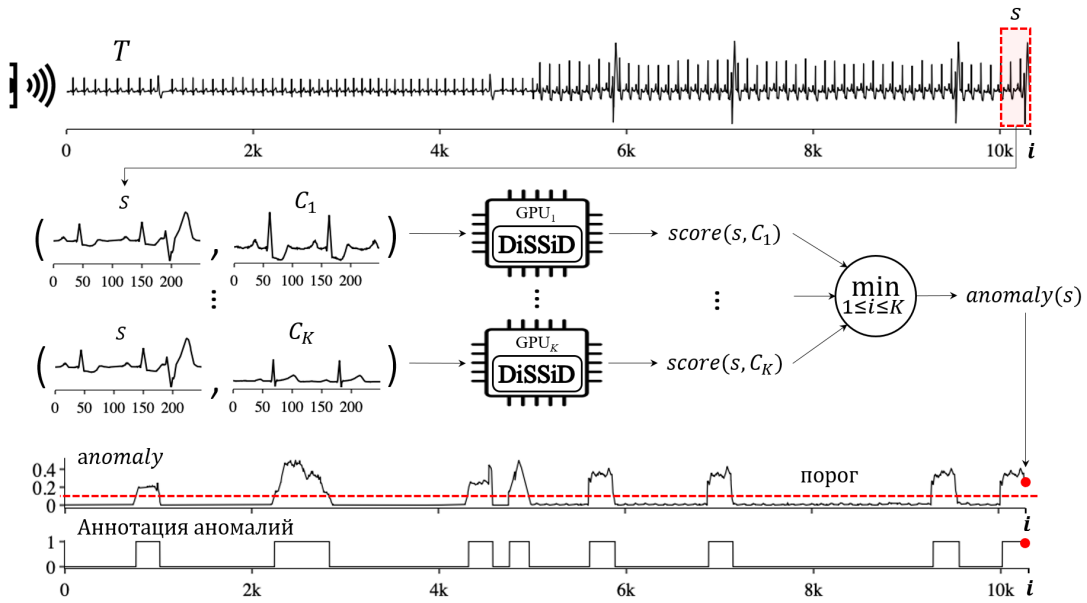


Рис. 3. Применение модели DiSSiD

Пусть имеется обученная модель DiSSiD, обучающая выборка которой содержит набор сниппетов $\{C_i\}_{i=1}^{K}$ и с помощью модели требуется определить, является ли входная подпо-

следовательность s аномальной. Предполагается, что K экземпляров модели запускаются каждый на отдельном графическом процессоре. Применение обученной модели выглядит следующим образом (см. рис. 3).

Сначала формируется набор пар $\{< s; C_i >\}_{i=1}^{K}$ «входная подпоследовательность, сниппет». Затем элементы данного набора параллельно подаются на вход экземплярам модели, каждая из которых выдает $score(s, C_i)$, соответствующую оценку схожести векторных представлений элементов входной пары в смысле расстояния MPdist. Далее оценка аномальности входной подпоследовательности получается как выполнение редукции по операции минимума $anomaly(s) = \min\limits_{1 \leqslant i \leqslant K} score(s, C_i)$. Подпоследовательность s считается аномальной, если $anomaly(s)$ превышает значение наперед заданного аналитиком порога.

В качестве порога используется значение k-го процентиля по набору оценок схожести, которые выдает модель DiSSiD на кортежах валидационной выборки $\mathcal{L}_{\text{valid}}$, имеющих вид $< s_1; s_2; 1 >$ (иными словами, порог — это k-й процентиль подпоследовательностей валидационной выборки, входящих в множество ближайших соседей одного и того же сниппета). В данном исследовании в качестве порога применяется $k = 95$.

3.2. Алгоритм формирования обучающей выборки

Для формирования обучающей выборки $\mathcal{L}$ для представленной выше модели DiSSiD аналитик выбирает репрезентативный временной ряд, адекватно отражающий типичную деятельность субъекта (противоположную аномалиям, которые предполагается обнаруживать с помощью модели). Автоматизированное формирование обучающей выборки включает в себя два шага: очистка и генерация. Очистка подразумевает формирование множества подпоследовательностей ряда, имеющих заданную аналитиком длину, и удаление из указанного множества аномальных подпоследовательностей, которые не должны попасть в обучающую выборку. На шаге генерации из очищенного множества подпоследовательностей тривиальным образом формируются описанные выше множества $\mathcal{L}_{\text{true}}$ и $\mathcal{L}_{\text{false}}$.

Алг. 1 CleanData (in: T, m, α, φ, K; out: $\mathcal{L}$)

1: $C_T^m \leftarrow \text{PSF}(T, m, K)$ ▷ Поиск типичной активности
2: $C_{\text{weak}} \leftarrow \{C_i \in C_T^m \mid C_i.frac \leqslant \varphi,\ 1 \leqslant i \leqslant K\}$ ▷ Поиск нетипичной активности
3: $C_T^m \leftarrow C_T^m \setminus C_{\text{weak}}$
4: **for** $i \leftarrow 1$ **to** $|C_T^m|$ **do**
5: $\quad O \leftarrow \text{IsolationForest}(C_i.profile) \cup O$ ▷ Поиск шумов
6: $n_{\text{discord}} \leftarrow \lceil \alpha \cdot (n - m + 1) \rceil$
7: $D \leftarrow \text{PALMAD}(T, m, m, n_{\text{discord}})$ ▷ Поиск аномальной активности
8: $\mathcal{L}_{\text{anomaly}} \leftarrow C_{\text{weak}} \cup C_{\text{weak}}.NN \cup D \cup O$
9: $\mathcal{L} \leftarrow S_T^m \setminus \mathcal{L}_{\text{anomaly}}$ ▷ Очистка
10: **return** $\mathcal{L}$

Псевдокод шага очистки представлен в алг. 1. Данный алгоритм имеет следующие задаваемые аналитиком параметры: репрезентативный ряд T ($|T| = n$), длина подпоследовательности m ($m \ll n$), предполагаемая доля аномальных подпоследовательностей в ряде α ($0 < \alpha \ll 1$), предполагаемое количество сниппетов K ($K > 1$), пороговая мощность сниппета φ ($0 < \varphi < 1/K$).

Для очистки обучающей выборки из исходного ряда удаляются подпоследовательности, которые соответствуют аномальной и нетипичной активности субъекта, а также подпоследовательности-шумы. Подпоследовательности, отражающие аномальную активность, трактуются как диссонансы. Подпоследовательностям нетипичной активности субъекта сопоставляются сниппеты, имеющие мощность меньше заданного порога φ, с множеством своих ближайших соседей. Подпоследовательности-шумы трактуются как выбросы в рамках каждого сниппета.

Поиск сниппетов выполняется с помощью параллельного алгоритма PSF (Parallel Snippet-Finder) [30] (см. стр. 1 в алг. 1). Затем из найденного множества сниппетов исключаются маломощные сниппеты (см. стр. 2, 3 в алг. 1). Далее в множестве ближайших соседей каждого сниппета выполняется нахождение подпоследовательностей-шумов, реализуемое как поиск выбросов в профиле данного сниппета с помощью метода изолирующего леса (Isolation Forest) [31] (см. стр. 4, 5 в алг. 1). Наконец, с помощью разработанного автором параллельного алгоритма PALMAD [24] выполняется поиск диссонансов, реализующий нахождение подпоследовательностей аномальной активности (см. стр. 6, 7 в алг. 1). Мощность множества диссонансов вычисляется на основе параметра α, долей аномальных подпоследовательностей в исходном временном ряде (типичным значением данного параметра является $\alpha = 0.05$). В завершении очистки из множества подпоследовательностей ряда исключаются найденные ранее маломощные сниппеты и их ближайшие соседи, а также выбросы и диссонансы (см. стр. 8, 9 в алг. 1), формируя тем самым множество, используемое для генерации обучающей выборки модели.

4. Вычислительные эксперименты

В данном разделе представлены результаты вычислительных экспериментов, проведенных на реальных временных рядах, которые имеют истинную разметку аномалий. В экспериментах выполняется сравнение точности предлагаемого метода DiSSiD с рассмотренными в обзоре аналогами (см. раздел 1), относящихся к методам с частичным привлечением учителя. Помимо этого, исследуется влияние функции расстояния между векторными представлениями входных подпоследовательностей (метрики L1 и предложенной в данной работе использование меры MPdist [25]) на эффективность обнаружения аномалий с помощью метода DiSSiD. В заключении данного раздела выполняется оценка времени обучения и тестирования DiSSiD и аналогов.

4.1. Наборы данных, аналоги и метрики сравнения

Временные ряды, использованные в экспериментах, взяты из реальных предметных областей и резюмированы в табл. 1. Данные взяты из общедоступного фреймворка TSB-UAD [12], предназначенного для проведения вычислительных экспериментов с алгоритмами обнаружения аномалий во временных рядах.

В экспериментах разработанная модель сравнивалась со следующими аналогами, принадлежащими, как и DiSSiD, к группе методов обнаружения аномалий с частичными привлечением учителя: LSTM-AD [22], OCSVM [39], AE [40], DeepAnT [23], IE-CAE [41], OceanWNN [42], Bagel [43], TAnoGAN [44]. Реализация указанных методов взята из работ [12, 45]. Кроме того, в сравнении участвовала версия DiSSiD, в которой модель выдает оценку схожести векторных представлений входных подпоследовательностей в виде евклидова расстояния вместо MPdist-расстояния.

Таблица 1. Временные ряды для вычислительных экспериментов

№ п/п	Временной ряд	Предметная область	Длина ряда n	Длина сниппета m	Длина значимого участка ℓ	Кол-во сниппетов K	Доля аномалий $\alpha, \times 10^{-4}$
1	SMD [32]	Показания потребления оперативной памяти сервером интернет-провайдера	23 706	100	20	2	5
2	OPP [33]	Показания носимого датчика движения при повседневной активности человека	26 204	200	50	2	5
3	Daphnet [34]	Показания носимого виброакселерометра, закрепленного на человеке с болезнью Паркинсона	19 200	216	72	2	5
4	ECG-2 (803, 805) [35]	Показания ЭКГ пациентов, страдающих синдромом преждевременного сокращения желудочков (конкатенация данных нескольких пациентов)	200 000	250	75	2	8
5	ECG-2 (803, 806) [35]		200 000	250	75	2	5
6	ECG-3 (803, 805, 806) [35]		300 000	250	75	3	10
7	MITDB [36]	Показания ЭКГ пациента с нарушенным сердечным ритмом	200 000	520	75	2	2
8	IOPS [37]	Показания производительности одного из серверов (операции ввода-вывода) компании Alibaba	129 010	1000	500	2	1
9	YAHOO [38]	Показания производительности одного из серверов (операции обращения к памяти) компании Yahoo	1422	60	30	2	10

Для оценки качества обнаружения аномалий используется метрика VUS-PR [46], интегрирует в себе как стандартные метрики — точность (precision) и полноту (recall), так и величину смещения найденной аномальной подпоследовательности относительно истинной аномалии. Метрика VUS-PR принимает значения из отрезка $[0, 1]$, большему значению соответствует лучшее качество.

Вычислительные эксперименты выполнялись на вычислительном узле комплекса «Нейрокомпьютер» Суперкомпьютерного центра ЮУрГУ [47], который оснащен графическим процессором NVIDIA Tesla V100 SXM2 (5120 ядер @1.3 ГГц).

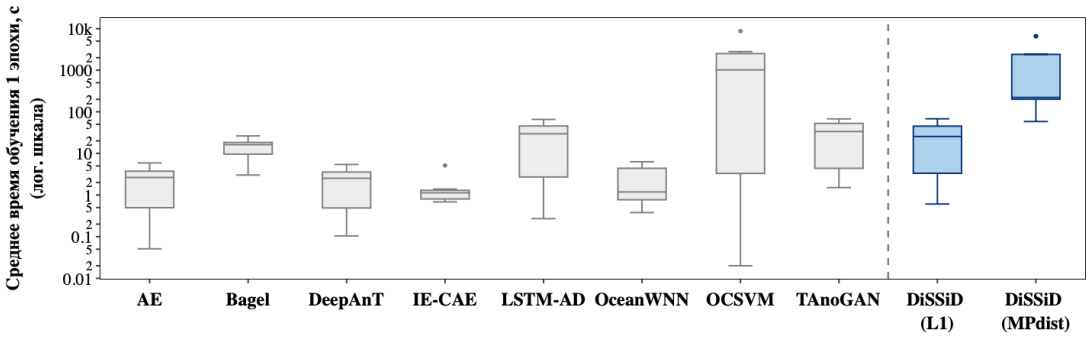
4.2. Результаты

Таблица 2. Сравнение точности метода DiSSiD с аналогами (метрика VUS-PR)

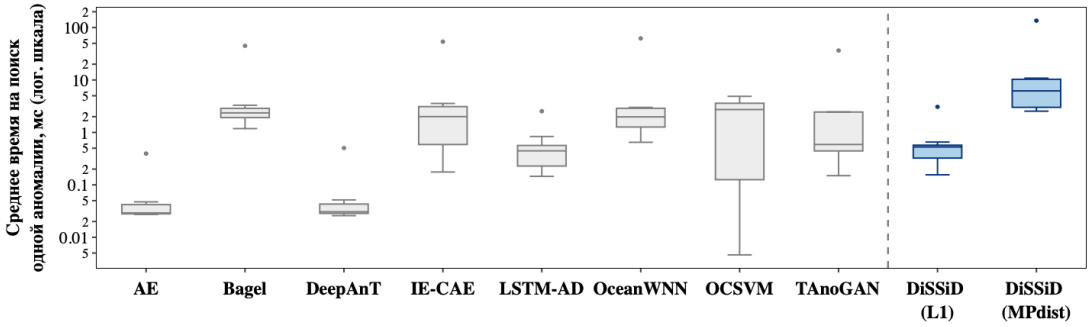
Методы / Ряды	AE	Bagel	DeepAnT	IE-CAE	LSTM-AD	OceanWNN	OCSVM	TAnoGAN	DiSSiD (L1)	DiSSiD (MPdist)
SMD	0.0767 (6)	0.0559 (8)	0.0522 (9)	0.1297 (3)	0.0653 (7)	0.1075 (4)	0.0119 (10)	0.0965 (5)	0.1543 (2)	**0.4889 (1)**
OPP	0.1979 (5)	nan (10)	0.0605 (9)	**0.9002 (1)**	0.0650 (8)	0.4678 (4)	0.1795 (6)	0.8090 (2)	0.1222 (7)	0.5340 (3)
Daphnet	0.2160 (6)	0.2269 (5)	0.2573 (4)	0.3079 (3)	0.1711 (8)	0.1812 (7)	0.1388 (10)	0.1609 (9)	**0.4124 (1)**	0.3332 (2)
ECG-2 (803, 805)	0.7758 (2)	0.3302 (8)	0.3350 (7)	0.5234 (5)	0.2897 (10)	0.5544 (4)	0.3548 (6)	0.3002 (9)	0.7477 (3)	**0.7801 (1)**
ECG-2 (803, 806)	0.5589 (3)	0.1878 (10)	0.2346 (7)	0.5397 (4)	0.1934 (9)	0.2003 (8)	0.3069 (6)	0.4635 (5)	**0.8008 (1)**	0.7927 (2)
ECG-3 (803, 805, 806)	0.7651 (2)	0.2988 (7)	0.2906 (8)	0.4739 (4)	0.2330 (9)	0.3596 (5)	0.3315 (6)	0.1430 (10)	0.7505 (3)	**0.8124 (1)**
MITDB	0.0759 (8)	0.0833 (5)	0.0795 (7)	0.1713 (3)	0.0799 (6)	0.1058 (4)	0.0474 (10)	0.0714 (9)	**0.3718 (1)**	0.3544 (2)
IOPS	0.3720 (7)	0.2678 (8)	0.1834 (10)	**0.9163 (1)**	0.1595 (10)	0.9085 (4)	0.7533 (6)	0.9130 (2)	0.2464 (9)	0.7922 (5)
YAHOO	0.7238 (2)	0.4871 (8)	0.5659 (7)	0.7050 (3)	0.4478 (10)	0.6126 (5)	0.6639 (4)	0.4591 (9)	0.5961 (6)	**0.7306 (1)**
Средний VUS-PR	0.4181 (4)	0.2422 (8)	0.2288 (9)	0.5186 (2)	0.1894 (10)	0.3886 (5)	0.3098 (7)	0.3851 (6)	0.4669 (3)	**0.6242 (1)**
Средний ранг	4.56 (4)	7.67 (9)	7.56 (8)	2.56 (2)	8.56 (10)	5 (5)	7.11 (7)	6.67 (6)	3.67 (3)	**2 (1)**

Результаты сравнения точности метода DiSSiD с аналогами представлены в табл. 2. В ячейке таблицы дано значение меры VUS-PR и в скобках — ранг метода, указанного в соответствующем столбце, среди всех аналогов на временном ряде, указанном в соответствующей строке. Полужирным шрифтом даны результат и место лучшего метода на заданном временном ряде. Две последние строки таблицы являются резюмирующими, в них

указаны соответственно средние значения метрики и ранга по всем рядам, а также среднее значение метрики и ранга метода в скобках. Можно видеть, что при применении евклидова расстояния в функции контрастных потерь метод DiSSiD в среднем входит в тройку лучших методов по точности обнаружения аномалий. Однако использование модифицированной функции контрастных потерь с расстоянием MPdist в формуле (13) позволяет добиться лучшей в среднем точности обнаружения аномалий.



а) обучение



б) тестирование

Рис. 4. Сравнение быстродействия метода DiSSiD с аналогами

Рисунок 4 показывает время работы метода DiSSiD по сравнению с аналогами. Можно видеть, что применение расстояния MPdist в функции контрастных потерь делает предложенную модель наиболее медленной как по времени ее обучения, так и по времени поиска аномалий. Причиной этого является применение в DiSSiD вычислительно затратной функции MPdist для нахождения расстояния между векторными представлениями входных подпоследовательностей (временная сложность указанной функции является кубической по отношению к длине подпоследовательности [11]). Низкое быстродействие является в данном случае обратной стороной высокой точности обнаружения аномалий (см. табл. 2).

Рассмотрим отдельно возможность применения модели DiSSiD в режиме реального времени. Из рис. 4б видно, что в экспериментах предложенная модель показывает среднее время поиска аномалий 0.1 с. Такое быстродействие модели допускает ее применение в режиме реального времени, что подтверждают следующие примеры. В системах автоматизации управления сеть передачи данных, обслуживающая датчики измерения температуры, влажности и давления, имеет цикл передачи данных до 100 мс [48]. Компания Emerson, один из ведущих мировых производителей измерительных систем, поставляет беспроводные температурные датчики, имеющие период обновления данных не менее 1 с [49].

Заключение

В статье рассмотрена задача поиска аномальных подпоследовательностей временного ряда, решение которой в настоящее время востребовано в широком спектре предметных областей: Интернет вещей, умное управление зданиями и городом, персональная медицина и др. Предложен новый метод обнаружения аномальных подпоследовательностей временного ряда с частичным привлечением учителя. Теоретическую основу метода составляют концепции диссонанса [10] и сниппета [11], которые формализуют понятия аномальных и типичных подпоследовательностей временного ряда соответственно. Предложенный метод включает в себя нейросетевую модель, которая определяет степень аномальности входной подпоследовательности ряда, и алгоритм автоматизированного построения обучающей выборки для этой модели.

Нейросетевая модель представляет собой сиамскую нейронную сеть (Siamese Neural Network) [8], которая объединяет в себе две идентичные подсети: количество слоев, число нейронов в каждом слое, размерности входного и выходного слоев, функции активации и др., а также наборы весов и смещений, полученных в результате обучения, одинаковы. Подсеть формирует векторное представление (embedding) входной подпоследовательности. На выходе модель выдает близость между сформированными векторными представлениями в смысле расстояния MPdist [11], используемого для поиска сниппетов. В качестве подсети фигурирует модификация нейросетевой модели ResNet [9]. Для обучения модели предложена модифицированная функция контрастных потерь.

Входным данным для формирования обучающей выборки является репрезентативный временной ряд, адекватно отражающего типичную деятельность субъекта, противоположную аномалиям, которые предполагается обнаруживать с помощью модели. Формирование обучающей выборки включает в себя два шага: очистка и генерация. Очистка подразумевает формирование множества подпоследовательностей ряда, имеющих заданную аналитиком длину, и удаление из указанного множества подпоследовательностей, которые отражают аномальную, нетипичную активность субъекта и шум. Подпоследовательности, отражающие аномальную активность, трактуются как диссонансы. Подпоследовательностям нетипичной активности субъекта сопоставляются сниппеты с мощностью менее заданного аналитиком порога и их ближайшие соседи. Подпоследовательности-шумы трактуются как выбросы в рамках каждого сниппета. На шаге генерации из очищенных подпоследовательностей формируются два множества, объединение которых дает искомую выборку. Элементами первого из них являются пары подпоследовательностей-ближайших соседей одного и того же сниппета, второго — ближайших соседей разных сниппетов.

Применение модели происходит следующим образом. Сначала формируется набор пар «входная подпоследовательность, сниппет». Затем элементы данного набора последовательно подаются на вход модели, которая выдает набор соответствующих оценок схожести векторных представлений элементов входных пар. Далее оценка аномальности входной подпоследовательности получается как минимальное значение по указанному набору. Входная подпоследовательность считается аномальной, если ее оценка превышает значение наперед заданного аналитиком порога. В качестве порога используется значение k-го процентиля подпоследовательностей валидационной выборки, входящих в множество ближайших соседей одного и того же сниппета (в данном исследовании применяется значение порога при $k = 95$).

Вычислительные эксперименты на временных рядах из различных предметных областей показывают, что предложенная модель по сравнению с аналогами показывает в среднем наиболее высокую точность обнаружения аномалий по стандартной метрике VUS-PR. Обратной стороной высокой точности метода является большее по сравнению с аналогами время, которое затрачивается на обучение модели и распознавание аномалии. Тем не менее, в приложениях интеллектуального управления отоплением зданий метод обеспечивает быстродействие, достаточное для обнаружения аномальных подпоследовательностей в режиме реального времени.

В качестве направления будущих исследований можно рассматривать расширение разработанного метода для обнаружения аномальных подпоследовательностей в многомерных временных рядах.

Работа выполнена при финансовой поддержке Российского научного фонда (грант № 23-21-00465).

Литература

1. Blázquez-García A., Conde A., Mori U., Lozano J.A. A Review on Outlier/Anomaly Detection in Time Series Data // ACM Comput. Surv. 2021. Vol. 54, no. 3. P. 56:1–56:33. DOI: 10.1145/3444690.
2. Kumar S., Tiwari P., Zymbler M.L. 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.
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. Volkov I., Radchenko G.I., Tchernykh A. Digital Twins, Internet of Things and Mobile Medicine: A Review of Current Platforms to Support Smart Healthcare // Program. Comput. Softw. 2021. Vol. 47, no. 8. P. 578–590. DOI: 10.1134/S0361768821080284.
6. Schmidl S., Wenig P., Papenbrock T. Anomaly Detection in Time Series: A Comprehensive Evaluation // Proc. VLDB Endow. 2022. Vol. 15, no. 9. P. 1779–1797. URL: https://www.vldb.org/pvldb/vol15/p1779-wenig.pdf.
7. Hodge V.J., Austin J. A Survey of Outlier Detection Methodologies // Artif. Intell. Rev. 2004. Vol. 22, no. 2. P. 85–126. DOI: 10.1023/B:AIRE.0000045502.10941.a9.
8. Chicco D. Siamese Neural Networks: An Overview // Artificial Neural Networks / ed. by H. Cartwright. New York, NY: Springer US, 2021. P. 73–94. DOI: 10.1007/978-1-0716-0826-5_3.
9. 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. IEEE Computer Society, 2016. P. 770–778. DOI: 10.1109/CVPR.2016.90.

10. Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: Finding unusual time series in terabyte sized datasets // Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007), October 28-31, 2007, Omaha, Nebraska, USA. 2007. P. 381–390. DOI: 10.1109/ICDM.2007.61.

11. Imani S., Madrid F., Ding W., *et al.* Matrix Profile XIII: Time Series Snippets: A New Primitive for Time Series Data Mining // 2018 IEEE International Conference on Big Knowledge, ICBK 2018, Singapore, November 17-18, 2018 / ed. by X. Wu, Y. Ong, C.C. Aggarwal, H. Chen. IEEE Computer Society, 2018. P. 382–389. DOI: 10.1109/ICBK.2018.00058.

12. Paparrizos J., Kang Y., Boniol P., *et al.* TSB-UAD: An End-to-End Benchmark Suite for Univariate Time-Series Anomaly Detection // Proc. VLDB Endow. 2022. Vol. 15, no. 8. P. 1697–1711. URL: https://www.vldb.org/pvldb/vol15/p1697-paparrizos.pdf.

13. Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: finding unusual time series in terabyte sized datasets // Knowl. Inf. Syst. 2008. Vol. 17, no. 2. P. 241–262. DOI: 10.1007/s10115-008-0131-9.

14. Yeh C.M., Zhu Y., Ulanova L., *et al.* Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile // Data Min. Knowl. Discov. 2018. Vol. 32, no. 1. P. 83–123. DOI: 10.1007/s10618-017-0519-9.

15. Nakamura T., Imamura M., Mercer R., Keogh E.J. MERLIN: Parameter-free discovery of arbitrary length anomalies in massive time series archives // 20th IEEE International Conference on Data Mining, ICDM 2020, Sorrento, Italy, November 17-20, 2020 / ed. by C. Plant, H. Wang, A. Cuzzocrea, *et al.* 2020. P. 1190–1195. DOI: 10.1109/ICDM50108.2020.00147.

16. Lu Y., Wu R., Mueen A., *et al.* DAMP: accurate time series anomaly detection on trillions of datapoints and ultra-fast arriving data streams // Data Min. Knowl. Discov. 2023. Vol. 37, no. 2. P. 627–669. DOI: 10.1007/s10618-022-00911-7.

17. Boniol P., Linardi M., Roncallo F., *et al.* Unsupervised and scalable subsequence anomaly detection in large data series // VLDB J. 2021. Vol. 30, no. 6. P. 909–931. DOI: 10.1007/s00778-021-00655-8.

18. Boniol P., Linardi M., Roncallo F., *et al.* Correction to: Unsupervised and scalable subsequence anomaly detection in large data series // VLDB J. 2023. Vol. 32, no. 2. P. 469. DOI: 10.1007/s00778-021-00678-1.

19. Li J., Pedrycz W., Jamal I. Multivariate time series anomaly detection: A framework of Hidden Markov Models // Appl. Soft Comput. 2017. Vol. 60. P. 229–240. DOI: 10.1016/j.asoc.2017.06.035.

20. Marteau P., Soheily-Khah S., Béchet N. Hybrid Isolation Forest - Application to Intrusion Detection // CoRR. 2017. Vol. abs/1705.03800. arXiv: 1705.03800. URL: http://arxiv.org/abs/1705.03800.

21. Ryzhikov A., Borisyak M., Ustyuzhanin A., Derkach D. Normalizing flows for deep anomaly detection // CoRR. 2019. Vol. abs/1912.09323. arXiv: 1912.09323. URL: http://arxiv.org/abs/1912.09323.

22. Malhotra P., Vig L., Shroff G., Agarwal P. Long Short Term Memory Networks for Anomaly Detection in Time Series // 23rd European Symposium on Artificial Neural Networks, ESANN 2015, Bruges, Belgium, April 22-24, 2015. 2015. URL: https://www.esann.org/sites/default/files/proceedings/legacy/es2015-56.pdf.

23. Munir M., Siddiqui S.A., Dengel A., Ahmed S. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series // IEEE Access. 2019. Vol. 7. P. 1991–2005. DOI: 10.1109/ACCESS.2018.2886457.

24. Zymbler M., Kraeva Y. High-Performance Time Series Anomaly Discovery on Graphics Processors // Mathematics. 2023. Vol. 11, no. 14. P. 3193. DOI: 10.3390/math11143193.

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. Yeh C.M., Zhu Y., Ulanova L., *et al.* Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets // IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain / ed. by F. Bonchi, J. Domingo-Ferrer, R. Baeza-Yates, *et al.* IEEE Computer Society, 2016. P. 1317–1322. DOI: 10.1109/ICDM.2016.0179.

27. Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. Vol. 37 / ed. by F.R. Bach, D.M. Blei. JMLR.org, 2015. P. 448–456. JMLR Workshop and Conference Proceedings. URL: http://proceedings.mlr.press/v37/ioffe15.html.

28. Hochreiter S. The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions // Int. J. Uncertain. Fuzziness Knowl. Based Syst. 1998. Vol. 6, no. 2. P. 107–116. DOI: 10.1142/S0218488598000094.

29. Hadsell R., Chopra S., LeCun Y. Dimensionality Reduction by Learning an Invariant Mapping // 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA. IEEE Computer Society, 2006. P. 1735–1742. DOI: 10.1109/CVPR.2006.100.

30. Zymbler M., Goglachev A. Fast Summarization of Long Time Series with Graphics Processor // Mathematics. 2022. Vol. 10, no. 10. P. 1781. DOI: 10.3390/math10101781.

31. Liu F.T., Ting K.M., Zhou Z. Isolation Forest // Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy. IEEE Computer Society, 2008. P. 413–422. DOI: 10.1109/ICDM.2008.17.

32. Su Y., Zhao Y., Niu C., *et al.* Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network // Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019. ACM, 2019. P. 2828–2837. DOI: 10.1145/3292500.3330672.

33. Roggen D., Calatroni A., Rossi M., *et al.* Collecting complex activity datasets in highly rich networked sensor environments // Seventh International Conference on Networked Sensing Systems, INSS 2010, Kassel, Germany, June 15-18, 2010. IEEE, 2010. P. 233–240. DOI: 10.1109/INSS.2010.5573462.

34. Bächlin M., Plotnik M., Roggen D., *et al.* Wearable assistant for Parkinson's disease patients with the freezing of gait symptom // IEEE Trans. Inf. Technol. Biomed. 2010. Vol. 14, no. 2. P. 436–446. DOI: 10.1109/TITB.2009.2036165.

35. Goldberger A.L., Amaral L.A.N., Glass L., *et al.* PhysioBank, PhysioToolkit, and PhysioNet components of a new research resource for complex physiologic signals // Circulation. 2000. Vol. 101, no. 23. P. 215–220. DOI: 10.1161/01.CIR.101.23.e215.

36. Moody G., Mark R. The impact of the MIT-BIH Arrhythmia Database // IEEE Engineering in Medicine and Biology Magazine. 2001. Vol. 20, no. 3. P. 45–50. DOI: 10.1109/51.932724.

37. KPI Anomaly Detection Dataset. 2018. URL: http://iops.ai/dataset_detail/?id=10 (дата обращения: 15.08.2023).

38. Laptev N., Amizadeh S., Billawala Y. S5 - A Labeled Anomaly Detection Dataset, version 1.0(16M). 2015. URL: https://webscope.sandbox.yahoo.com/catalog.php?%20datatype=s%5C&did=70 (дата обращения: 15.08.2023).

39. Schölkopf B., Williamson R.C., Smola A.J., *et al.* Support Vector Method for Novelty Detection // Advances in Neural Information Processing Systems 12, [NIPS Conference, Denver, Colorado, USA, November 29 - December 4, 1999] / ed. by S.A. Solla, T.K. Leen, K. Müller. The MIT Press, 1999. P. 582–588. URL: http://papers.nips.cc/paper/1723-support-vector-method-for-novelty-detection.

40. Sakurada M., Yairi T. Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction // Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, Gold Coast, Australia, QLD, Australia, December 2, 2014 / ed. by A. Rahman, J.D. Deng, J. Li. ACM, 2014. P. 4. DOI: 10.1145/2689746.2689747.

41. Garcia G.R., Michau G., Ducoffe M., *et al.* Time Series to Images: Monitoring the Condition of Industrial Assets with Deep Learning Image Processing Algorithms // CoRR. 2020. Vol. abs/2005.07031. arXiv: 2005.07031. URL: https://arxiv.org/abs/2005.07031.

42. Wang Y., Han L., Liu W., *et al.* Study on wavelet neural network based anomaly detection in ocean observing data series // Ocean Engineering. 2019. Vol. 186. P. 106129. DOI: 10.1016/j.oceaneng.2019.106129.

43. Li Z., Chen W., Pei D. Robust and Unsupervised KPI Anomaly Detection Based on Conditional Variational Autoencoder // 37th IEEE International Performance Computing and Communications Conference, IPCCC 2018, Orlando, FL, USA, November 17-19, 2018. IEEE, 2018. P. 1–9. DOI: 10.1109/PCCC.2018.8710885.

44. Bashar M.A., Nayak R. TAnoGAN: Time Series Anomaly Detection with Generative Adversarial Networks // 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Canberra, Australia, December 1-4, 2020. IEEE, 2020. P. 1778–1785. DOI: 10.1109/SSCI47803.2020.9308512.

45. Wenig P., Schmidl S., Papenbrock T. TimeEval: A Benchmarking Toolkit for Time Series Anomaly Detection Algorithms // Proc. VLDB Endow. 2022. Vol. 15, no. 12. P. 3678–3681. URL: https://www.vldb.org/pvldb/vol15/p3678-schmidl.pdf.

46. Paparrizos J., Boniol P., Palpanas T., *et al.* Volume Under the Surface: A New Accuracy Evaluation Measure for Time-Series Anomaly Detection // Proc. VLDB Endow. 2022. Vol. 15, no. 11. P. 2774–2787. URL: https://www.vldb.org/pvldb/vol15/p2774-paparrizos.pdf.

47. Биленко Р.В., Долганина Н.Ю., Иванова Е.В., Рекачинский А.И. Высокопроизводительные вычислительные ресурсы Южно-Уральского государственного университета // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2022. Т. 11, № 1. С. 15–30. DOI: 10.14529/cmse220102.

48. Лопухов И. Сети Real-Time Ethernet: от теории к практической реализации // СТА: Современные технологии автоматизациии. 2010. Т. 10, № 3. С. 8–15.

49. Каталог 2021. Датчики температуры Emerson. URL: https://www.c-o-k.ru/library/catalogs/emerson/110477.pdf (дата обращения: 03.09.2021).

Краева Яна Александровна, старший преподаватель, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

DOI: 10.14529/cmse230304

DETECTION OF TIME SERIES ANOMALIES BASED ON DATA MINING AND NEURAL NETWORK TECHNOLOGIES

© 2023 Ya.A. Kraeva

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

E-mail: kraevaya@susu.ru

Received: 20.05.2023

The article touches upon the problem of discovering subsequence anomalies in time series, which is currently in demand in a wide range of subject domains. We propose a new semi-supervised method to detect subsequence anomalies in time series. The method is based on the concepts of discord and snippet, which formalize, respectively, the concepts of anomalous and typical time series subsequences. The proposed method includes a neural network model that calculates the anomaly score of the input subsequence and an algorithm to automatically construct the model's training set. The model is implemented as a Siamese neural network, where we employ a modification of ResNet as a subnet. To train the model, we proposed a modified contrast loss function. The training set is formed as a representative fragment of the time series from which discords, low-fraction snippets with their nearest neighbors, and outliers within each snippet are removed since they are interpreted as abnormal, atypical activity of the subject, and noise, respectively. Computational experiments over time series from various subject domains showed that the proposed model, compared with analogues, has on average the highest accuracy of anomaly detection with respect to the standard VUS-PR metric. The downside of the high accuracy of the method is the longer time spent on model training and anomaly detection compared to analogues. Nevertheless, in applications of intelligent building heating control, the method provides a speed sufficient to detect subsequence anomalies in real time.

Keywords: time series, anomaly detection, discord, snippet, Siamese neural network.

FOR CITATION

Kraeva Ya.A. Detection of Time Series Anomalies Based on Data Mining and Neural Network Technologies. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. 12, no. 3. P. 50–71. (in Russian) DOI: 10.14529/cmse230304.

References

1. Blázquez-García A., Conde A., Mori U., Lozano J.A. A Review on Outlier/Anomaly Detection in Time Series Data. ACM Comput. Surv. 2021. Vol. 54, no. 3. P. 56:1–56:33. DOI: 10.1145/3444690.
2. Kumar S., Tiwari P., Zymbler M.L. 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.
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 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 Software Engineering. 2020. Vol. 9, no. 4. P. 5–23. (in Russian) DOI: 10.14529/cmse200401.
5. Volkov I., Radchenko G.I., Tchernykh A. Digital Twins, Internet of Things and Mobile Medicine: A Review of Current Platforms to Support Smart Healthcare. Program. Comput. Softw. 2021. Vol. 47, no. 8. P. 578–590. DOI: 10.1134/S0361768821080284.
6. Schmidl S., Wenig P., Papenbrock T. Anomaly Detection in Time Series: A Comprehensive Evaluation. Proc. VLDB Endow. 2022. Vol. 15, no. 9. P. 1779–1797. URL: https://www.vldb.org/pvldb/vol15/p1779-wenig.pdf.
7. Hodge V.J., Austin J. A Survey of Outlier Detection Methodologies. Artif. Intell. Rev. 2004. Vol. 22, no. 2. P. 85–126. DOI: 10.1023/B:AIRE.0000045502.10941.a9.
8. Chicco D. Siamese Neural Networks: An Overview. Artificial Neural Networks / ed. by H. Cartwright. New York, NY: Springer US, 2021. P. 73–94. DOI: 10.1007/978-1-0716-0826-5_3.
9. 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. IEEE Computer Society, 2016. P. 770–778. DOI: 10.1109/CVPR.2016.90.
10. Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: Finding unusual time series in terabyte sized datasets. Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007), October 28-31, 2007, Omaha, Nebraska, USA. 2007. P. 381–390. DOI: 10.1109/ICDM.2007.61.
11. Imani S., Madrid F., Ding W., *et al.* Matrix Profile XIII: Time Series Snippets: A New Primitive for Time Series Data Mining. 2018 IEEE International Conference on Big Knowledge, ICBK 2018, Singapore, November 17-18, 2018 / ed. by X. Wu, Y. Ong, C.C. Aggarwal, H. Chen. IEEE Computer Society, 2018. P. 382–389. DOI: 10.1109/ICBK.2018.00058.
12. Paparrizos J., Kang Y., Boniol P., *et al.* TSB-UAD: An End-to-End Benchmark Suite for Univariate Time-Series Anomaly Detection. Proc. VLDB Endow. 2022. Vol. 15, no. 8. P. 1697–1711. URL: https://www.vldb.org/pvldb/vol15/p1697-paparrizos.pdf.

13. Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: finding unusual time series in terabyte sized datasets. Knowl. Inf. Syst. 2008. Vol. 17, no. 2. P. 241–262. DOI: 10.1007/s10115-008-0131-9.

14. Yeh C.M., Zhu Y., Ulanova L., *et al.* Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile. Data Min. Knowl. Discov. 2018. Vol. 32, no. 1. P. 83–123. DOI: 10.1007/s10618-017-0519-9.

15. Nakamura T., Imamura M., Mercer R., Keogh E.J. MERLIN: Parameter-free discovery of arbitrary length anomalies in massive time series archives. 20th IEEE International Conference on Data Mining, ICDM 2020, Sorrento, Italy, November 17-20, 2020 / ed. by C. Plant, H. Wang, A. Cuzzocrea, *et al.* IEEE, 2020. P. 1190–1195. DOI: 10.1109/ICDM50108.2020.00147.

16. Lu Y., Wu R., Mueen A., *et al.* DAMP: accurate time series anomaly detection on trillions of datapoints and ultra-fast arriving data streams. Data Min. Knowl. Discov. 2023. Vol. 37, no. 2. P. 627–669. DOI: 10.1007/s10618-022-00911-7.

17. Boniol P., Linardi M., Roncallo F., *et al.* Unsupervised and scalable subsequence anomaly detection in large data series. VLDB J. 2021. Vol. 30, no. 6. P. 909–931. DOI: 10.1007/s00778-021-00655-8.

18. Boniol P., Linardi M., Roncallo F., *et al.* Correction to: Unsupervised and scalable subsequence anomaly detection in large data series. VLDB J. 2023. Vol. 32, no. 2. P. 469. DOI: 10.1007/s00778-021-00678-1.

19. Li J., Pedrycz W., Jamal I. Multivariate time series anomaly detection: A framework of Hidden Markov Models. Appl. Soft Comput. 2017. Vol. 60. P. 229–240. DOI: 10.1016/j.asoc.2017.06.035.

20. Marteau P., Soheily-Khah S., Béchet N. Hybrid Isolation Forest - Application to Intrusion Detection. CoRR. 2017. Vol. abs/1705.03800. arXiv: 1705.03800. URL: http://arxiv.org/abs/1705.03800.

21. Ryzhikov A., Borisyak M., Ustyuzhanin A., Derkach D. Normalizing flows for deep anomaly detection. CoRR. 2019. Vol. abs/1912.09323. arXiv: 1912.09323. URL: http://arxiv.org/abs/1912.09323.

22. Malhotra P., Vig L., Shroff G., Agarwal P. Long Short Term Memory Networks for Anomaly Detection in Time Series. 23rd European Symposium on Artificial Neural Networks, ESANN 2015, Bruges, Belgium, April 22-24, 2015. 2015. URL: https://www.esann.org/sites/default/files/proceedings/legacy/es2015-56.pdf.

23. Munir M., Siddiqui S.A., Dengel A., Ahmed S. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series. IEEE Access. 2019. Vol. 7. P. 1991–2005. DOI: 10.1109/ACCESS.2018.2886457.

24. Zymbler M., Kraeva Y. High-Performance Time Series Anomaly Discovery on Graphics Processors. Mathematics. 2023. Vol. 11, no. 14. P. 3193. DOI: 10.3390/math11143193.

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. Yeh C.M., Zhu Y., Ulanova L., *et al.* Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets. IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain / ed. by F. Bonchi, J. Domingo-Ferrer, R. Baeza-Yates, *et al.* IEEE Computer Society, 2016. P. 1317–1322. DOI: 10.1109/ICDM.2016.0179.

27. Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. Vol. 37 / ed. by F.R. Bach, D.M. Blei. JMLR.org, 2015. P. 448–456. JMLR Workshop and Conference Proceedings. URL: http://proceedings.mlr.press/v37/ioffe15.html.

28. Hochreiter S. The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions. Int. J. Uncertain. Fuzziness Knowl. Based Syst. 1998. Vol. 6, no. 2. P. 107–116. DOI: 10.1142/S0218488598000094.

29. Hadsell R., Chopra S., LeCun Y. Dimensionality Reduction by Learning an Invariant Mapping. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA. IEEE Computer Society, 2006. P. 1735–1742. DOI: 10.1109/CVPR.2006.100.

30. Zymbler M., Goglachev A. Fast Summarization of Long Time Series with Graphics Processor. Mathematics. 2022. Vol. 10, no. 10. P. 1781. DOI: 10.3390/math10101781.

31. Liu F.T., Ting K.M., Zhou Z. Isolation Forest. Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy. IEEE Computer Society, 2008. P. 413–422. DOI: 10.1109/ICDM.2008.17.

32. Su Y., Zhao Y., Niu C., *et al.* Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019. ACM, 2019. P. 2828–2837. DOI: 10.1145/3292500.3330672.

33. Roggen D., Calatroni A., Rossi M., *et al.* Collecting complex activity datasets in highly rich networked sensor environments. Seventh International Conference on Networked Sensing Systems, INSS 2010, Kassel, Germany, June 15-18, 2010. IEEE, 2010. P. 233–240. DOI: 10.1109/INSS.2010.5573462.

34. Bächlin M., Plotnik M., Roggen D., *et al.* Wearable assistant for Parkinson's disease patients with the freezing of gait symptom. IEEE Trans. Inf. Technol. Biomed. 2010. Vol. 14, no. 2. P. 436–446. DOI: 10.1109/TITB.2009.2036165.

35. Goldberger A.L., Amaral L.A.N., Glass L., *et al.* PhysioBank, PhysioToolkit, and PhysioNet components of a new research resource for complex physiologic signals. Circulation. 2000. Vol. 101, no. 23. P. 215–220. DOI: 10.1161/01.CIR.101.23.e215.

36. Moody G., Mark R. The impact of the MIT-BIH Arrhythmia Database. IEEE Engineering in Medicine and Biology Magazine. 2001. Vol. 20, no. 3. P. 45–50. DOI: 10.1109/51.932724.

37. KPI Anomaly Detection Dataset. 2018. URL: http://iops.ai/dataset_detail/?id=10 (accessed: 15.08.2023).

38. Laptev N., Amizadeh S., Billawala Y. S5 - A Labeled Anomaly Detection Dataset, version 1.0(16M). 2015. URL: https://webscope.sandbox.yahoo.com/catalog.php?%20datatype=s&did=70 (accessed: 15.08.2023).

39. Schölkopf B., Williamson R.C., Smola A.J., *et al.* Support Vector Method for Novelty Detection. Advances in Neural Information Processing Systems 12, [NIPS Conference, Denver, Colorado, USA, November 29 - December 4, 1999] / ed. by S.A. Solla, T.K. Leen, K. Müller. The MIT Press, 1999. P. 582–588. URL: http://papers.nips.cc/paper/1723-support-vector-method-for-novelty-detection.

40. Sakurada M., Yairi T. Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction. Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, Gold Coast, Australia, QLD, Australia, December 2, 2014 / ed. by A. Rahman, J.D. Deng, J. Li. ACM, 2014. P. 4. DOI: 10.1145/2689746.2689747.

41. Garcia G.R., Michau G., Ducoffe M., *et al.* Time Series to Images: Monitoring the Condition of Industrial Assets with Deep Learning Image Processing Algorithms. CoRR. 2020. Vol. abs/2005.07031. arXiv: 2005.07031. URL: https://arxiv.org/abs/2005.07031.

42. Wang Y., Han L., Liu W., *et al.* Study on wavelet neural network based anomaly detection in ocean observing data series. Ocean Engineering. 2019. Vol. 186. P. 106129. DOI: 10.1016/j.oceaneng.2019.106129.

43. Li Z., Chen W., Pei D. Robust and Unsupervised KPI Anomaly Detection Based on Conditional Variational Autoencoder. 37th IEEE International Performance Computing and Communications Conference, IPCCC 2018, Orlando, FL, USA, November 17-19, 2018. IEEE, 2018. P. 1–9. DOI: 10.1109/PCCC.2018.8710885.

44. Bashar M.A., Nayak R. TAnoGAN: Time Series Anomaly Detection with Generative Adversarial Networks. 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Canberra, Australia, December 1-4, 2020. IEEE, 2020. P. 1778–1785. DOI: 10.1109/SSCI47803.2020.9308512.

45. Wenig P., Schmidl S., Papenbrock T. TimeEval: A Benchmarking Toolkit for Time Series Anomaly Detection Algorithms. Proc. VLDB Endow. 2022. Vol. 15, no. 12. P. 3678–3681. URL: https://www.vldb.org/pvldb/vol15/p3678-schmidl.pdf.

46. Paparrizos J., Boniol P., Palpanas T., *et al.* Volume Under the Surface: A New Accuracy Evaluation Measure for Time-Series Anomaly Detection. Proc. VLDB Endow. 2022. Vol. 15, no. 11. P. 2774–2787. URL: https://www.vldb.org/pvldb/vol15/p2774-paparrizos.pdf.

47. Bilenko R.V., Dolganina N.Y., Ivanova E.V., Rekachinsky A.I. High-performance Computing Resources of South Ural State University. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2022. Vol. 11, no. 1. P. 15–30. (in Russian) DOI: 10.14529/cmse220102.

48. Lopukhov I. Real-Time Ethernet network: from theory to practical implementation. MAT: Modern automation technologies. 2010. Vol. 10, no. 3. P. 8–15.

49. Catalogue 2021. Emerson temperature sensors. URL: https://www.c-o-k.ru/library/catalogs/emerson/110477.pdf (accessed: 03.09.2021).