

# НЕЙРОСЕТЕВОЙ МЕТОД ОБНАРУЖЕНИЯ АНОМАЛИЙ В МНОГОМЕРНЫХ ПОТОКОВЫХ ВРЕМЕННЫХ РЯДАХ

© 2024 Я.А. Краева

*Южно-Уральский государственный университет*

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

*E-mail: kraevaya@susu.ru*

Поступила в редакцию: 20.10.2024

В статье рассмотрена задача детекции аномальных подпоследовательностей многомерного потокового временного ряда, элементы которого поступают в режиме реального времени, возникающая в настоящее время в широком спектре предметных областей: промышленный Интернет вещей, персональное здравоохранение и др. Предложен новый метод решения указанной задачи, получивший название mDiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of multivariate anomalies). Предложенный метод использует концепцию диссонанса временного ряда (подпоследовательность, имеющая наиболее непохожего на нее ближайшего соседа), обобщенную на многомерный случай. Под многомерным диссонансом понимается  $N$ -мерная подпоследовательность  $d$ -мерного временного ряда (где  $1 \leq N \leq d$ ), которая наиболее непохожа на все остальные подпоследовательности  $N$ -мерных временных рядов, полученных путем составления всевозможных сочетаний из  $d$  рядов по  $N$ . Детекция аномалий реализуется с помощью нейросетевой модели на основе сиамских нейросетей. Вычислительные эксперименты на реальных временных рядах из различных предметных областей показали, что метод mDiSSiD в среднем опережает по точности обнаружения аномалий передовые аналоги, использующие иные нейросетевые подходы (сверточные и рекуррентные нейронные сети, автоэнкодеры, генеративно-состязательные сети).

*Ключевые слова: многомерный временной ряд, поиск аномалий, диссонанс, сноплет, сиамская нейронная сеть.*

## ОБРАЗЕЦ ЦИТИРОВАНИЯ

Краева Я.А. Нейросетевой метод обнаружения аномалий в многомерных потоковых временных рядах // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2024. Т. 13, № 4. С. 35–52. DOI: 10.14529/cmse240403.

## Введение

В настоящее время в широком спектре приложений востребованы эффективные модели, методы и алгоритмы для детекции аномальных подпоследовательностей временных рядов, элементы которых поступают для обработки в режиме реального времени [1]: Интернет вещей [2], умное управление зданиями [3, 4], персональная медицина [5] и др. На текущий момент научным сообществом разработано большое количество методов и подходов к решению указанной задачи (см., например, обзорные статьи [6, 7], в которых суммарно рассматривается около ста различных методов). Детекция аномалий с частичным привлечением учителя (semi-supervised anomaly detection) является одним из наиболее часто используемых и эффективных подходов [6, 8]. Указанный подход предполагает этап обучения, в рамках которого нейросетевая модель «понимает», каким является нормальное поведение временного ряда, фигурирующего в качестве обучающей выборки. Далее модель применяется к тестовому временному ряду и помечает его подпоследовательности как аномальные, если они не соответствуют обученному нормальному поведению. К основным представителям подхода с частичным привлечением учителя относят методы LSTM-AD [9], DeepAnT [10] и др.

Ранее автором данной статьи был предложен метод DiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of anomalies) [11] для детекции аномальных подпоследовательностей потокового временного ряда с частичным привлечением учителя. Метод включает в себя нейросетевую модель, которая выдает оценку аномальности входной подпоследовательности, и алгоритм автоматизированного построения обучающей выборки для этой модели. Нейросетевая модель основана на архитектуре сиамской нейронной сети (Siamese Neural Network) [12], в которой в качестве подсети фигурирует модификация нейросетевой модели ResNet [13]. Алгоритм построения выборки отбрасывает из обучающей выборки подпоследовательности, соответствующие аномальным и нетипичным активностям субъекта, которые извлекаются на основе концепций диссонанса [14] и снippets [15] соответственно. Однако метод DiSSiD решает задачу детекции аномалий только для одномерного потокового временного ряда. В настоящей статье предлагается новый нейросетевой метод mDiSSiD (multivariate DiSSiD), который представляет собой обобщение вышеупомянутой ранней разработки на случай многомерного ряда. Метод mDiSSiD использует понятие многомерного диссонанса [16], которое представляет собой обобщение концепции диссонанса на случай многомерного временного ряда.

Статья организована следующим образом. В разделе 1 приводятся формальные определения базовых понятий. В разделе 2 представлено описание метода mDiSSiD. Раздел 3 содержит результаты вычислительных экспериментов по исследованию эффективности предложенного метода. Заключение подводит итоги исследования.

## 1. Теоретический базис

### 1.1. Временной ряд и подпоследовательность

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

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

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  $i$ :

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

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

*Многомерный временной ряд* представляет собой занумерованную последовательность нескольких семантически связанных и синхронизированных по времени одномерных временных рядов равной длины. Обозначим размерность многомерного ряда целым положительным числом  $d$  ( $d > 1$ ). Подобно одномерному случаю, введем обозначения  $\mathbf{T}$ ,  $\mathbf{T}_{i,m}$  и  $\mathbf{t}_i$  для многомерных временного ряда, его подпоследовательности и точки соответственно, и определим их следующим образом:

$$\mathbf{T} = [\{T^{(k)}\}_{k=1}^d]^\top, \quad (3)$$

$$\mathbf{T}_{i,m} = [\{T_{i,m}^{(k)}\}_{k=1}^d]^\top, \quad (4)$$

$$\mathbf{t}_i = [\{t_i^{(k)}\}_{k=1}^d]^\top. \quad (5)$$

## 1.2. Сниметы

*Сниметы* [15] (поведенческие шаблоны) временного ряда представляют собой множество подпоследовательностей, выражающих типичные активности субъекта, деятельность которого описывает данный ряд. Для формального определения сниметов зафиксируем временной ряд  $T$ , длину подпоследовательности  $m$  ( $m \ll n$ ) и количество активностей субъекта  $K$  ( $1 \leq K \leq n/m$ ), интересующее исследователя. Множество сниметов обозначается как  $C_T^m$  и представляет собой непустое подмножество сегментов ряда, имеющих длину  $m^1$ :

$$C_T^m = \{C_i\}_{i=1}^K, \quad C_i \in Seg_T^m, \quad Seg_T^m = \{T_{m \cdot (i-1)+1, m}\}_{i=1}^{n/m}. \quad (6)$$

Со сниметом  $C_i \in C_T^m$  ассоциированы следующие атрибуты: индекс, ближайшие соседи и значимость данного снимета, которые обозначаются соответственно  $C_i.index$ ,  $C_i.NN$  и  $C_i.frac$ , и имеют следующие определения.

*Индекс снимета* представляет собой номер  $j$  сегмента  $Seg_j$ , которому соответствует подпоследовательность ряда  $T_{m \cdot (j-1)+1, m}$ .

*Множество ближайших соседей* содержит подпоследовательности, которые среди всех сегментов наиболее похожи на данный снимет в смысле расстояния MPdist:

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

*Значимость снимета*  $C_i$  вычисляется как отношение мощностей множеств  $C_i.NN$  и  $S_T^m$ , при этом сниметы упорядочиваются по убыванию их значимости:

$$C_i.frac = \frac{|C_i.NN|}{n - m + 1}, \quad \forall C_i, C_j \in C_T^m : i < j \iff C_i.frac \geq C_j.frac. \quad (8)$$

Для формального определения расстояния  $MPdist(A, B)$ ,  $|A| = |B| = m$ , фиксируем параметр  $\ell$  ( $1 < \ell < m$ ), который отражает длину семантически значимого непрерывного промежутка точек подпоследовательности. Вычисление  $MPdist(A, B)$  предполагает следующие шаги [17]: 1) вычисление матричных профилей  $A$  и  $B$ , взятых в указанном и обратном порядке; 2) конкатенация профилей; 3) сортировка полученного ряда по возрастанию; 4) выдача  $k$ -го элемента результирующего ряда. То же в формальной записи:

$$MPdist_\ell(A, B) = AscSort(P_{ABBA})(k), \quad P_{ABBA} = P_{AB} \bullet P_{BA}, \quad (9)$$

где  $AscSort(\cdot)$  — сортировка по возрастанию, символ  $\bullet$  обозначает конкатенацию,  $k$  ( $0 < k < m$ ) — параметр, имеющий типичное значение  $k = \lceil 0.1m \rceil$ .

*Матричным профилем* [18] рядов  $A$  и  $B$  называется ряд  $P_{AB}$ , где  $i$ -й элемент есть расстояние между  $i$ -й подпоследовательностью ряда  $A$ , имеющей длину  $\ell$ , и ее ближайшим соседом в ряде  $B$ :

$$P_{AB} = \{ED_{\text{norm}}^2(A_{i, \ell}, B_{j, \ell})\}_{i=1}^{m-\ell+1}, \quad B_{j, \ell} = \arg \min_{1 \leq q \leq m-\ell+1} ED_{\text{norm}}^2(A_{i, \ell}, B_{q, \ell}), \quad (10)$$

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

<sup>1</sup>Поскольку  $m \ll n$ , без ограничения общности можно полагать, что  $n$  кратно  $m$  и представить ряд как набор не пересекающихся сегментов длины  $m$ .

### 1.3. Диссонансы

Подпоследовательности ряда  $T_{i,m}$  и  $T_{j,m}$  назовем *не пересекающимися*, если  $|i - j| \geq m$ . Подпоследовательность  $D$  ряда  $T$  является *диссонансом* [14], если ее ближайший сосед (ближайшая и не пересекающаяся с ней подпоследовательность) находится на расстоянии не менее чем  $r$  (заданный аналитиком порог):

$$\min_{M_D \in T} (\text{Dist}(D, M_D)) \geq r, \quad (11)$$

где  $\text{Dist}(\cdot, \cdot)$  — неотрицательная симметричная функция расстояния, а  $M_D$  обозначает подпоследовательность ряда, не пересекающуюся с данной подпоследовательностью  $D$ .

Обобщение концепции диссонанса на случай многомерного ряда выглядит следующим образом [16]. Под многомерным диссонансом понимается  $N$ -мерная подпоследовательность  $d$ -мерного временного ряда (где  $1 \leq N \leq d$ ), которая наиболее не похожа на все остальные подпоследовательности  $N$ -мерных временных рядов, полученных путем составления всевозможных сочетаний из  $d$  рядов по  $N$ .

## 2. Метод обнаружения аномалий во временных рядах

В данном разделе представлен новый метод обнаружения аномальных подпоследовательностей многомерного временного ряда в режиме реального времени, получивший название mDiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of multivariate anomalies). Ниже в разделе 2.1 дана общая схема метода mDiSSiD. Далее подробно описываются компоненты разработанного метода: Препроцессор, формирующий обучающую выборку для нейросетевой модели, и Детектор аномалий, построенный на основе сиамской нейронной сети, — в разделах 2.2 и 2.3 соответственно.

### 2.1. Общая схема

Разработанный метод mDiSSiD включает в себя два следующих компонента: Препроцессор и Детектор аномалий. На вход Препроцессору поступает репрезентативный фрагмент потокового временного ряда для исследуемой предметной области. При этом предполагается, что данный фрагмент ряда содержит все типичные активности субъекта. Далее каждый одномерный временной ряд, который входит в состав исходного многомерного ряда, подвергается очистке и разметке с помощью разработанного ранее алгоритма формирования обучающей выборки, представленного автором данной статьи в работе [11]. В результате алгоритм для каждого измерения формирует очищенное множество подпоследовательностей с указанием для них метки активности, к которой они принадлежат. Далее в рамках каждого измерения Препроцессор выполняет минимаксную нормализацию подпоследовательностей, приводя их значения к диапазону  $[0, 1]$ , и формирует обучающую выборку для Детектора, который представляет собой ансамбль модифицированных нейросетевых моделей DiSSiD.

Каждый участник ансамбля обрабатывает соответствующее измерение многомерного временного ряда и определяет, с каким из сниппетов ряда поданная на вход подпоследовательность этого измерения имеет наибольшее семантическое сходство. Затем выполняется агрегация результатов, полученных участниками ансамбля, с целью вычисления оценок аномальности для  $N$ -мерных подпоследовательностей, составленных из всевозможных сочетаний из  $d$  измерений входной многомерной подпоследовательности по  $N$ , где  $1 \leq N \leq d$ .

Многомерная подпоследовательность считается аномалией, если хотя бы одна из полученных оценок аномальности превышает порог, вычисленный для каждой размерности  $N$  на валидационной выборке.

## 2.2. Препроцессор

Препроцессор выполняет подготовку обучающей выборки для нейросетевой модели Детектора аномалий. Для этого зафиксируем репрезентативный фрагмент временного ряда, который содержит все активности субъекта, и обозначим его за  $\mathbf{T}$ . Далее каждое измерение  $T^{(i)}$  фрагмента  $\mathbf{T}$  подвергается предварительной обработке с помощью разработанного ранее алгоритма формирования обучающей выборки [11]. В качестве входных параметров алгоритма задаются длина подпоследовательности  $m$  ( $m \ll n$ ), доля аномальных подпоследовательностей в ряде  $\alpha$  ( $0 < \alpha \ll 1$ ), количество снипшетов  $K$  ( $K > 1$ ) и пороговая значимость снипшета  $\phi$  ( $0 < \phi < 1/K$ ). При этом для обработки всех измерений ряда устанавливаются одинаковые значения входных параметров. Сначала с помощью параллельного алгоритма PSF [19] выполняется разметка подпоследовательностей измерения многомерного ряда в соответствии с активностями субъекта. Тем самым формируется множество подпоследовательностей  $S^{(i)}$ , определяемое следующим образом:

$$S^{(i)} = \{\langle T_{j,m}^{(i)}, k \rangle \mid T_{j,m}^{(i)} \in C_k^{(i)}.NN, 1 \leq i \leq d, 1 \leq j \leq n - m + 1, 1 \leq k \leq K\}. \quad (12)$$

Далее из дальнейшей обработки исключаются подпоследовательности-шумы  $O^{(i)}$ , трактуемые как выбросы в рамках каждого снипшета, и подпоследовательности, соответствующие редко встречающимся активностям  $C_{\text{weak}}^{(i)}$ :

$$C_{\text{weak}}^{(i)} = \left\{ \{C_k^{(i)} \cup C_k^{(i)}.NN\} \in C_{T^{(i)}}^m \mid C_k^{(i)}.frac \leq \phi, 1 \leq i \leq d, 1 \leq k \leq K \right\}. \quad (13)$$

Во-вторых, с помощью параллельного алгоритма PALMAD [20] выполняется поиск диссонансов (аномальных подпоследовательностей)  $D^{(i)}$ ,  $|D^{(i)}| = \alpha \cdot (n - m + 1)$ , которые также исключаются из дальнейшей обработки. В завершении выполняется очистка размеченного множества подпоследовательностей  $S^{(i)}$  от найденных на предыдущих шагах Препроцессора различного рода аномалий:

$$S^{(i)} = S^{(i)} \setminus \{\langle T_{j,m}^{(i)}, k \rangle\}, \text{ где } T_{j,m}^{(i)} \in \{C_{\text{weak}}^{(i)} \cup O^{(i)} \cup D^{(i)}\}. \quad (14)$$

Затем Препроцессор выполняет min-max нормализацию подпоследовательностей, входящих в множество  $S^{(i)}$ , для каждого измерения  $T^{(i)}$ , преобразуя их значения в диапазон  $[0, 1]$ . Нормализованная версия  $\hat{T}_{j,m}^{(i)}$  подпоследовательности  $T_{j,m}^{(i)}$  вычисляется следующим образом:

$$\hat{T}_{j,m}^{(i)} = (\hat{t}_j^{(i)}, \dots, \hat{t}_{j+m-1}^{(i)}), 1 \leq i \leq d, \quad (15)$$

$$\text{где } \hat{t}_j^{(i)} = \frac{t_j^{(i)} - \min_{1 \leq p \leq n} t_p^{(i)}}{\max_{1 \leq p \leq n} t_p^{(i)} - \min_{1 \leq p \leq n} t_p^{(i)}}.$$

Далее выполняется процедура генерации обучающей выборки для нейросетевой модели Детектора аномалий. Поскольку предусматривается осуществлять поиск аномалий в каждом измерении независимо друг от друга, то для каждого измерения формируется своя обучающая выборка. Обозначим обучающую выборку для  $i$ -го измерения многомерного ряда  $\mathbf{T}$

за  $\mathcal{L}^{(i)}$ . Обучающая выборка  $\mathcal{L}^{(i)}$  представляет собой множество прецедентов вида  $\langle X, Y \rangle$ . Под  $X$  понимается набор пар подпоследовательностей, количество которых совпадает с количеством активностей (сниплетов) во временном ряде. Первый элемент в каждой паре одинаков и представляет собой подпоследовательность  $\widehat{T}_{j,m}^{(i)}$  из множества  $S^{(i)}$ , оставшуюся в результате предобработки. В качестве второго (различающего) элемента пары фигурирует соответствующий сниплет  $C_k^{(i)}$  или ближайший сосед этого сниплета, взятый из множества  $C_k^{(i)}.NN$ . Под  $Y$  подразумевается  $K$ -мерный бинарный вектор, один элемент которого равен 1, если подпоследовательности в паре отражают одинаковую активность, остальные элементы равны 0 — в противном случае. Таким образом, для каждой предобработанной подпоследовательности измерения можно получить  $nnQty$  прецедентов, где  $nnQty$  — параметр алгоритма. При этом чем больше устанавливается значение  $nnQty$ , тем более вариативной становится обучающая выборка. Формальное определение обучающей выборки  $\mathcal{L}^{(i)}$  выглядит следующим образом:

$$\begin{aligned} \mathcal{L}^{(i)} = \left\{ \langle X^{(i)}, Y^{(i)} \rangle \mid X^{(i)} = \{ \langle x_1^{(i)}, x_{2_k}^{(i)} \rangle \}_{k=1}^K, \right. \\ x_1^{(i)} = \widehat{T}_{j,m}^{(i)}, \quad x_{2_k}^{(i)} = C_k^{(i)} \vee x_{2_k}^{(i)} = \widehat{T}_{p,m}^{(i)}, \quad \widehat{T}_{j,m}^{(i)}, \widehat{T}_{p,m}^{(i)} \in S^{(i)}, \\ Y^{(i)} = \{ [y_1^{(i)} = y_{2_k}^{(i)}] \}_{k=1}^K, \\ y_1^{(i)} = k_1, \quad x_1^{(i)} \in C_{k_1}^{(i)}.NN, \quad y_{2_k}^{(i)} = k_2, \quad x_{2_k}^{(i)} \in C_{k_2}^{(i)}.NN, \\ \left. 1 \leq i \leq d, 1 \leq j, p \leq n - m + 1 \right\}. \end{aligned} \quad (16)$$

### 2.3. Детектор аномалий

Нейросетевая модель mDiSSiD представляет собой ансамбль из  $d$  модификаций модели DiSSiD (для одномерного ряда), где  $d$  — размерность исходного многомерного ряда  $\mathbf{T}$ . Задача участника ансамбля заключается в том, что необходимо определить, с каким из сниплетов ряда поданная на вход подпоследовательность назначенного ему измерения имеет наибольшее семантическое сходство. На вход модифицированной модели DiSSiD поступает набор из  $K$  пар подпоследовательностей  $X^{(i)}$ , каждая из которых включает входную подпоследовательность  $T_{n-m+1,m}^{(i)}$ , оканчивающуюся текущим поступившим значением  $t_n^{(i)}$ , и один из сниплетов  $C_k^{(i)}$ . На выходе модель выдает  $K$ -мерный вектор  $\widehat{Y}^{(i)}$ , содержащий оценки схожести между подпоследовательностями в парах. Таким образом, участника ансамбля формально можно определить как  $f: \mathbb{R}^{K \times 2 \times m} \rightarrow \mathbb{R}^K$ . Изучение соответствующего измерения и выдача результатов каждым участником ансамбля ведется независимо друг от друга. Такая организация обработки обусловлена определением  $N$ -мерного диссонанса, требующего выполнять сравнение поданной на вход одномерной подпоследовательности с подпоследовательностями из того же измерения.

На рис. 1 представлена структура участника ансамбля, являющейся общей для всех участников. Участник ансамбля включает  $K$  подсетей, каждая из которых представляет собой разработанную ранее модель DiSSiD [11] (см. рис. 2). Модель DiSSiD является сиамской нейронной сетью (Siamese Neural Network, SNN) [12], сконструированной из двух сверточных нейронных сетей ResNet, идентичных по конфигурации и наборам обучаемых параметров (весов и смещений). Сначала на вход DiSSiD поступает соответствующая пара подпоследовательностей  $\langle x_1^{(i)}, x_{2_k}^{(i)} \rangle$ . Далее сиамская сеть выполняет отображение входных данных в скрытые представления (embedding)  $h_1^{(i)}$  и  $h_2^{(i)}$  изученного латентного пространства  $\mathbb{R}^s$ , имеющие меньшую размерность, чем исходные,  $s < m$ .

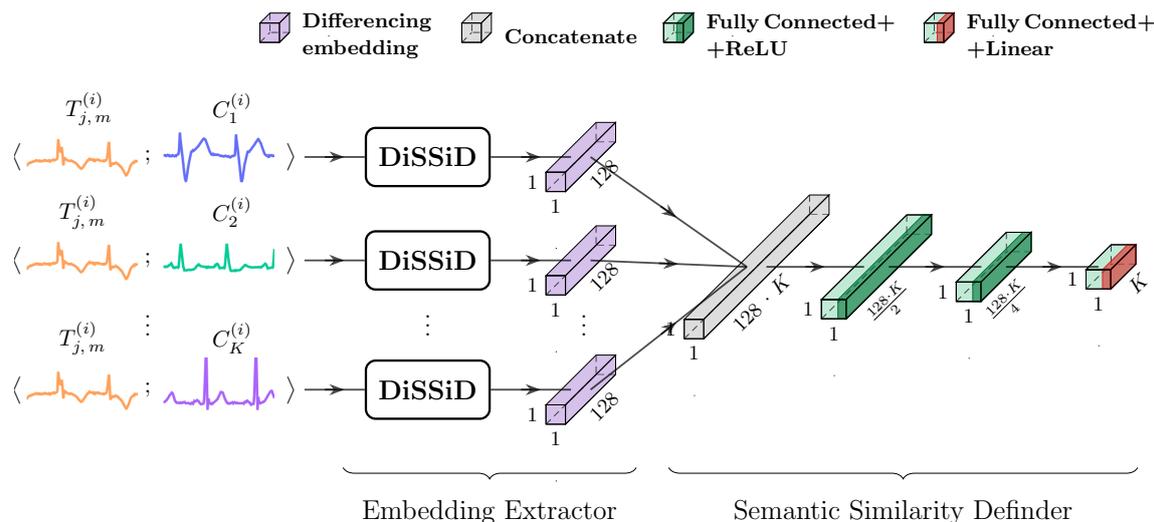


Рис. 1. Структура участника нейросетевой модели mDiSSiD

ResNet включает в себя три одинаковых остаточных блока (residual block), состоящих из трех сверточных слоев с одномерными ядрами свертки размеров  $8 \times 1$ ,  $5 \times 1$  и  $3 \times 1$  соответственно. Каждый сверточный слой чередуется с пакетной нормализацией (batch normalization) [21], после которой применяется функция активации Линейный выпрямитель (ReLU, Rectified linear unit). После прохождения всех слоев очередного остаточного блока формируются карты признаков, которые далее складываются с входом этого остаточного блока, пропущенного через сверточный слой с ядром свертки размера  $1 \times 1$ . Первый остаточный блок формирует 64 карты признаков, остальные два блока — по 128 карт. За остаточными блоками следует слой глобальной усредняющей агрегации (GlobalAveragePooling), формирующий итоговое скрытое представление,  $i$ -й элемент которого вычисляется как среднее арифметическое по всем элементам  $i$ -й карты признаков.

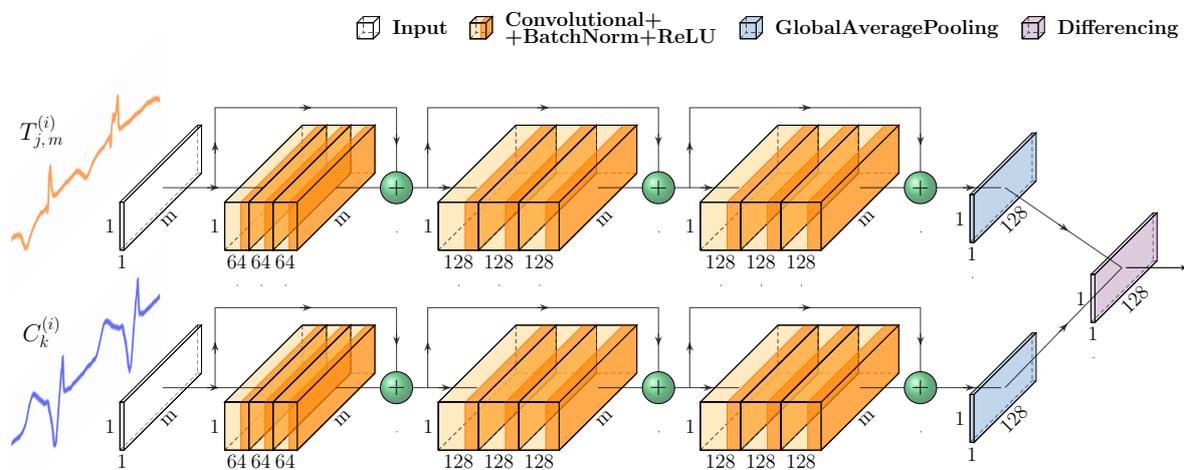


Рис. 2. Нейросетевая модель DiSSiD

Далее на различающем слое (differencing layer) определяется схожесть между сформированными скрытыми представлениями  $h_1^{(i)}$  и  $h_{2_k}^{(i)}$  путем вычисления вектора  $h_{diff_k}^{(i)}$  из расстояний между соответствующими элементами этих скрытых представлений:

$$h_{diff_k}^{(i)}(j) = \text{dist}(h_1^{(i)}(j), h_{2_k}^{(i)}(j)).$$

В данном исследовании в качестве функции расстояния  $\text{dist}(\cdot, \cdot)$  выбираются метрики L1 и L2. Затем следует слой конкатенации, выполняющий сцепление результатов, выданных различающими слоями по всем  $K$  подсетям участника ансамбля. В результате формируется вектор

$$h_{concat}^{(i)} = h_{diff_1}^{(i)} \bullet \dots \bullet h_{diff_K}^{(i)}, \quad h_{concat}^{(i)} \in \mathbb{R}^{s \times K},$$

где  $\bullet$  — операция конкатенации. Замыкают модифицированную модель DiSSiD три полносвязных слоя, которые направлены на вычисление оценок схожести входной подпоследовательности со всеми снippetsами:

$$\hat{Y}^{(i)} = f(X^{(i)}, W_i, B_i),$$

где  $W_i$  и  $B_i$  — матрицы весов и смещений  $i$ -го участника ансамбля соответственно.

На скрытых полносвязных слоях с числом нейронов  $(s \cdot K)/2$  и  $(s \cdot K)/4$  используется функция активации ReLU, на выходном слое из  $K$  нейронов — линейная функция активации. В завершении формируется итоговая оценка для подпоследовательности  $T_{n-m+1,m}^{(i)}$ , относящейся к соответствующему измерению, путем взятия минимума по всем выданным на последнем слое нейронной сети оценкам:

$$\text{score}(T_{n-m+1,m}^{(i)}) = \min_{1 \leq k \leq K} \hat{Y}_k^{(i)}.$$

Итоговая оценка определяет степень схожести подпоследовательности с наиболее близким к ней снippetом.

Для определения того, является ли входная подпоследовательность  $T_{n-m+1,m}$  многомерным диссонансом, сначала формируются  $N$ -мерные подпоследовательности путем составления всевозможных сочетаний из  $d$  измерений этой подпоследовательности по  $N$ , где  $1 \leq N \leq d$ . Обозначим множество всех подпоследовательностей, содержащих различное количество измерений от 1 до  $d$ , за  $\mathcal{P}$ . Данное множество  $\mathcal{P}$  является булеаном (показательным множеством) и определяется следующим образом:

$$\begin{aligned} \mathcal{P} &= \{1\text{-}\mathcal{P}, 2\text{-}\mathcal{P}, \dots, d\text{-}\mathcal{P}\}, \text{ где} \\ 1\text{-}\mathcal{P} &= \{\{T_{n-m+1,m}^{(i)}\} \mid 1 \leq i \leq d\}, \\ 2\text{-}\mathcal{P} &= \{\{T_{n-m+1,m}^{(i)}, T_{n-m+1,m}^{(j)}\} \mid 1 \leq i, j \leq d, i < j\}, \\ &\dots \\ d\text{-}\mathcal{P} &= \{\{T_{n-m+1,m}^{(1)}, \dots, T_{n-m+1,m}^{(d)}\}\}. \end{aligned}$$

Количество сгенерированных многомерных подпоследовательностей в множестве  $\mathcal{P}$  составляет  $|\mathcal{P}| = \sum_{N=1}^d C_d^N = \sum_{N=1}^d \frac{d!}{(d-N)!N!} = 2^d - 1$ .

Затем для каждой  $N$ -мерной подпоследовательности  $N\text{-}\mathcal{P}_i$  определяется ее оценка схожести  $N\text{-score}(N\text{-}\mathcal{P}_i)$ . Для этого формируется набор из итоговых оценок  $\{\text{score}(N\text{-}\mathcal{P}_{i,j})\}_{j=1}^N$ ,

полученных нейросетевой моделью mDiSSiD на предыдущем шаге для входящих в это сочетание элементов (одномерных подпоследовательностей). Далее среди всех имеющихся в данном наборе оценок выбирается минимальная, которая назначается в качестве результирующей оценки схожести для  $N$ -мерной подпоследовательности  $N-\mathcal{P}_i$ :

$$N\text{-score}(N-\mathcal{P}_i) = \min_{1 \leq j \leq N} \text{score}(N-\mathcal{P}_{i,j}).$$

После этого для каждой группы сочетаний, объединенных по равному числу входящих в них элементов, вычисляется оценка аномальности путем взятия максимального значения по оценкам схожести, полученных для всех сочетаний из одной группы:

$$N\text{-anomaly}(\mathbf{T}_{n-m+1,m}) = \max_{1 \leq i \leq |N-\mathcal{P}|} N\text{-Score}(N-\mathcal{P}_i).$$

Максимум означает то, что в качестве определяющего сочетания в группе выбирается наименее похожее. На основании этого сочетания будет делаться вывод о том, является ли многомерная подпоследовательность нормой или аномалией. Входная многомерная подпоследовательность  $\mathbf{T}_{n-m+1,m}$  считается многомерным диссонансом, если хотя бы одна из оценок аномальности  $N\text{-anomaly}$  превышает некоторый установленный порог,  $1 \leq N \leq d$ .

Порог для каждого значения  $N$  формируется путем взятия  $k$ -го перцентиля по набору оценок аномальности, полученных для сочетаний из  $N$  измерений многомерных подпоследовательностей валидационной выборки. В данном исследовании в качестве порога применяется значение  $k = 95$ .

### 3. 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 mDiSSiD с аналогами, относящихся к методам с частичным привлечением учителя. Помимо этого, исследуется влияние функции расстояния между векторными представлениями входных подпоследовательностей (метрики L1 и L2) на эффективность обнаружения аномалий с помощью метода mDiSSiD.

#### 3.1. Параметры экспериментов

**Наборы данных.** Временные ряды, использованные в экспериментах, взяты из реальных предметных областей и резюмированы в табл. 1. Данные взяты из общедоступного фреймворка TimeEval [22], предназначенного для проведения вычислительных экспериментов с алгоритмами обнаружения аномалий в многомерных временных рядах.

Ряд OPPORTUNITY (OPP) [23] представляет собой показания носимых датчиков движения, закрепленных на человеке, выполняющего в течение 2 час. различные виды утренней повседневной активности: подъем, перемещение по комнате, приготовление завтрака, уборка и отдых.

Ряд Daphnet [24] содержит показания виброакселерометров, закрепленных на пациенте с болезнью Паркинсона, для обнаружения у него симптомов замирания походки. Пациент выполнял следующие виды физической активности: ходьба по прямой, ходьба с поворотами, перемещения между разными комнатами с открыванием дверей и др.

Каждый из рядов MITDB-2 (105, 217) и MITDB-3 (105, 116, 217) [25] представляет собой конкатенацию ЭКГ реальных пациентов, страдающих нарушениями сердечного ритма. Каждая из сцепленных ЭКГ отражает одну активность пациента.

Таблица 1. Временные ряды для вычислительных экспериментов

№ п/п	Временной ряд	Длина ряда $n$	Размерность ряда $d$	Длина снippetsа $m$	Длина значимого участка $\ell$	Кол-во снippetsов $K$	Доля аномалий $\alpha, \times 10^{-4}$
1	OPP	26 204	6	200	50	2	5
2	Daphnet	28 800	3	216	72	2	5
3	MITDB-2	100 000	2	600	75	2	2
4	MITDB-3	150 000	2	600	75	3	3

**Аналоги.** В экспериментах разработанная модель сравнивалась со следующими аналогами, принадлежащими, как и mDiSSiD, к группе методов обнаружения аномалий с частичным привлечением учителя: LSTM-AD [9], AE [26], DeepAnT [10], TAnoGAN [27]. Реализация указанных методов взята из работы [22].

**Метрики сравнения.** Для оценки качества обнаружения аномалий используется метрика VUS-PR [28], 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.

**Аппаратная платформа.** Вычислительные эксперименты выполнялись на вычислительном узле комплекса «Нейрокомпьютер» Суперкомпьютерного центра ЮУрГУ [29], оснащенного графическим процессором NVIDIA Tesla V100 SXM2 (5120 ядер @1.3 ГГц).

**Обучение модели.** Для обучения нейросетевой модели mDiSSiD (участника ансамбля) предлагается следующая функция контрастных потерь (contrastive loss) [30], адаптированная под многомерный случай:

$$L(\langle X_j^{(i)}, Y_j^{(i)} \rangle, W_i, B_i) = \sum_{k=1}^K \left( y_{j,k}^{(i)} \cdot (f(x_{j,k}^{(i)}, W_i, B_i))^2 + (1 - y_{j,k}^{(i)}) \cdot (\max(\tau - f(x_{j,k}^{(i)}, W_i, B_i), 0))^2 \right), \quad (17)$$

где  $i$  — номер участника ансамбля,  $\langle X_j^{(i)}, Y_j^{(i)} \rangle$  — прецедент выборки,  $K$  — количество снippetsов,  $\tau$  — минимальное расстояние между векторными представлениями исходных подпоследовательностей, являющихся ближайшими соседями разных снippetsов (параметр модели). Указанная функция потерь обеспечивает обучение модели, в результате которого похожие подпоследовательности исходного ряда получают векторные представления, отстоящие друг от друга в смысле расстояния не более чем на  $\tau$ , а непохожие — более чем на  $\tau$  соответственно.

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

Эксперименты выполнялись при следующих установленных значениях гиперпараметров: скорость обучения  $\mu = 1 \cdot 10^{-3}$ , количество эпох  $epochs = 50$ , размер батча  $batch\_size = 128$ , отступ  $\tau = 1$ . В качестве оптимизатора стохастического градиентного спуска использовался Adam [31].

### 3.2. Результаты

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

Методы Ряды	AE	DeepAnT	LSTM-AD	TAnoGAN	mDiSSiD (L1)	mDiSSiD (L2)
OPP	0.4812 (2)	0.2105 (6)	0.2178 (5)	<b>0.7632 (1)</b>	0.4154 (4)	0.4273 (3)
Daphnet	0.3569 (4)	0.3817 (3)	0.2347 (5)	0.2105 (6)	0.4356 (2)	<b>0.4632 (1)</b>
MITDB-2	0.5181 (4)	0.4514 (5)	0.4153 (6)	0.5208 (3)	0.6467 (2)	<b>0.6917 (1)</b>
MITDB-3	0.4496 (3)	0.3189 (6)	0.3284 (5)	0.4265 (4)	0.5898 (2)	<b>0.6250 (1)</b>
Средний VUS-PR	0.4515 (4)	0.3406 (5)	0.2990 (6)	0.4803 (3)	0.5219 (2)	<b>0.5518 (1)</b>
Средний ранг	3.25 (4)	5.00 (5)	5.25 (6)	3.50 (3)	2.50 (2)	<b>1.50 (1)</b>

Результаты сравнения точности метода mDiSSiD 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 метрики L1 в на различающемся слое нейронной сети участника ансамбля метод mDiSSiD в среднем входит в тройку лучших методов по точности обнаружения аномалий. При этом при использовании метрики L2 позволяет добиться лучшей в среднем точности обнаружения аномалий. Это связано с тем, что метрика L2 более чувствительна к выбросам, поэтому она делает более важный и больший акцент на тех парах скрытых представлений на различающей слое, которые соответствуют подпоследовательностям, принадлежащим разным активностям, или когда входная подпоследовательность является аномалией.

### Заключение

В статье рассмотрена задача детекции аномальных подпоследовательностей многомерного потокового временного ряда, элементы которого поступают в режиме реального времени, востребованная в настоящее время в широком спектре предметных областей: промышленный Интернет вещей, персональное здравоохранение и др. Описан новый метод решения указанной задачи, получивший название mDiSSiD (multivariate DiSSiD), который обобщает на многомерный случай предложенный автором ранее метод DiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of anomalies) для детекции аномалий в одномерном потоковом временном ряде.

Предложенный метод использует концепцию диссонанса временного ряда (подпоследовательность, имеющая наиболее не похожего на нее ближайшего соседа) [14], обобщенную на многомерный случай. Под многомерным диссонансом понимается [16]  $N$ -мерная подпоследовательность  $d$ -мерного временного ряда (где  $1 \leq N \leq d$ ), которая наиболее не похожа на все остальные подпоследовательности  $N$ -мерных временных рядов, полученных путем составления всевозможных сочетаний из  $d$  рядов по  $N$ .

Детекция аномалий реализуется с помощью нейросетевой модели на основе сиамских нейросетей. Модель mDiSSiD представляет собой ансамбль из  $d$  модификаций модели DiSSiD. Входной слой модифицированной модели DiSSiD представляет собой  $K$  пар, где  $K$  — количество различаемых активностей исследуемого субъекта, в которых первой компонентой является входная подпоследовательность, а второй — сноплет либо один из его ближайших соседей. Далее следуют слои, предусмотренные моделью DiSSiD: остаточные блоки ResNet, слой GlobalAveragePooling и различающий слой. После этого следует слой конкатенации, выполняющий сцепление результатов, выданных различающими слоями по всем измерениям. Замыкают модифицированную модель три полносвязных слоя, обеспечивающие оценки схожести входной подпоследовательности со всеми сноплетами. Затем формируется итоговая оценка для подпоследовательности, относящейся к соответствующему измерению, путем взятия минимума по всем выданным на последнем слое нейронной сети оценкам. Итоговая оценка определяет степень схожести подпоследовательности с наиболее близким к ней сноплетом.

Вычислительные эксперименты на реальных временных рядах из различных предметных областей (ЭКГ, показания виброакселерометра во время физической активности, мониторинг нагрузки серверов баз данных) показали, что метод mDiSSiD в среднем опережает по точности обнаружения аномалий передовые аналоги, использующие иные нейросетевые архитектуры: сверточные и рекуррентные нейронные сети, автоэнкодеры, генеративно-состязательные сети.

*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. 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. 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>.
8. 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.
9. 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>.
10. 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.
11. 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.
12. 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.
13. 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.
14. 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.
15. 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.
16. Tafazoli S., Keogh E.J. Matrix Profile XXVIII: Discovering Multi-Dimensional Time Series Anomalies with  $K$  of  $N$  Anomaly Detection<sup>†</sup> // Proceedings of the 2023 SIAM International Conference on Data Mining, SDM 2023, Minneapolis-St. Paul Twin Cities, MN, USA, April 27-29, 2023 / ed. by S. Shekhar, Z. Zhou, Y. Chiang, G. Stiglic. SIAM, 2023. P. 685–693. DOI: 10.1137/1.9781611977653.CH77.
17. 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.

18. 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.
19. Zymbler M., Goglavchev A. Fast Summarization of Long Time Series with Graphics Processor // Mathematics. 2022. Vol. 10, no. 10. P. 1781. DOI: 10.3390/math10101781.
20. 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.
21. 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>.
22. 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>.
23. 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.
24. 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.
25. 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.
26. 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.
27. 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.
28. 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>.
29. 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.

30. 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.
31. Kingma D., Ba J. Adam: A Method for Stochastic Optimization // International Conference on Learning Representations (ICLR). San Diego, CA, USA, 2015.

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

---

DOI: 10.14529/cmse240403

## DEEP LEARNING METHOD FOR ANOMALY DETECTION IN STREAMING MULTIVARIATE TIME SERIES

© 2024 Ya.A. Kraeva

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

*E-mail: kraevaya@susu.ru*

Received: 20.10.2024

The article touches upon the problem of detecting anomalous subsequences of multivariate streaming time series, where the elements arrive in real time, which currently arises in a wide range of subject domains: industrial Internet of Things, personal healthcare, etc. In the article, we introduce a novel method to solve such a problem, called mDiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of multivariate anomalies). The mDiSSiD method employs the time series discord concept (a subsequence with the most dissimilar nearest neighbor), which is generalized to the multivariate case. Multivariate discord refers to the  $N$ -dimensional subsequence of a  $d$ -dimensional time series (where  $1 \leq N \leq d$ ), which is the most dissimilar to all other subsequences of  $N$ -dimensional time series obtained by composing all the possible combinations of  $d$  series of  $N$ . Anomaly detection is implemented through a deep learning model based on the Siamese neural network architecture. Experimental evaluation of mDiSSiD over real time series from various subject domains showed that the proposed method is on average ahead of state-of-the-art analogs based on other deep learning approaches (convolutional and recurrent neural networks, autoencoders, and generative-adversarial networks) in terms of anomaly detection accuracy.

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

### FOR CITATION

Kraeva Ya.A. Deep Learning Method for Anomaly Detection in Streaming Multivariate Time Series. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2024. Vol. 13, no. 4. P. 35–52. (in Russian) DOI: 10.14529/cmse240403.

*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. 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. 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. 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>.
8. 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.
9. 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>.
10. 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.
11. Kraeva Y.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.
12. 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.
13. 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.

14. 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.
15. 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.
16. Tafazoli S., Keogh E.J. Matrix Profile XXVIII: Discovering Multi-Dimensional Time Series Anomalies with  $K$  of  $N$  Anomaly Detection<sup>†</sup>. Proceedings of the 2023 SIAM International Conference on Data Mining, SDM 2023, Minneapolis-St. Paul Twin Cities, MN, USA, April 27-29, 2023 / ed. by S. Shekhar, Z. Zhou, Y. Chiang, G. Stiglic. SIAM, 2023. P. 685–693. DOI: 10.1137/1.9781611977653.CH77.
17. 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.
18. 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.
19. Zymbler M., Goglavchev A. Fast Summarization of Long Time Series with Graphics Processor. Mathematics. 2022. Vol. 10, no. 10. P. 1781. DOI: 10.3390/math10101781.
20. 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.
21. 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>.
22. 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>.
23. 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.
24. 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.
25. 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.

26. 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.
27. 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.
28. 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>.
29. 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.
30. 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.
31. Kingma D., Ba J. Adam: A Method for Stochastic Optimization. International Conference on Learning Representations (ICLR). San Diego, CA, USA, 2015.