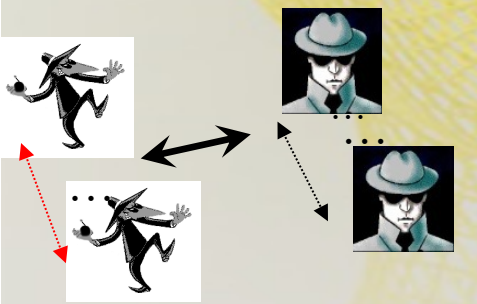
Обнаружение аномалий и атак в сетях IoT на основе кластеризации событий безопасности и графовых моделей

И.В. Котенко

ГНС, д.т.н., проф., Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), заслуженный деятель науки Российской Федерации



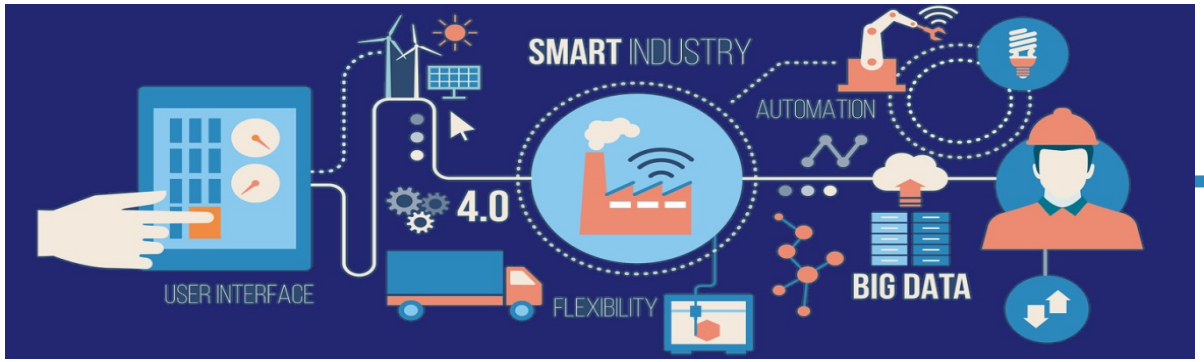
СПб ФИЦ РАН

РусКрипто 2025, 18-21 марта 2025 г.

Содержание

- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Роль корреляции событий в SIEM-системах



Мониторинг событий

Хранение событий

SIEM-система

Корреляция событий безопасности

Основанная на сходстве событий

Основанная на причинно-следственных связях событий

Основанное на знаниях

Оценка уязвимостей

Обнаружение активов

...

Генерация контрмер

- Аналитика безопасности в современных системах IoT означает **анализ множества системных событий** для обеспечения безопасности системы.
- В общем, **событие безопасности** — это идентифицированное состояние системы, которое указывает на потенциальную проблему безопасности или неизвестную ситуацию, связанную с безопасностью.
- **Корреляция событий безопасности** — это процесс поиска взаимосвязей между отдельными событиями для определения текущих и прогнозируемых состояний безопасности.

Основные применения корреляции событий

H.Dong, I.Kotenko. Cybersecurity in the AI Era: Analyzing the Impact of Machine Learning on Intrusion Detection // Knowledge and Information Systems, Springer Science and Business Media Deutschland GmbH. 2025. P.1-52. DOI: 10.1007/s10115-025-02366-w.

Обнаружение аномалий в IoT

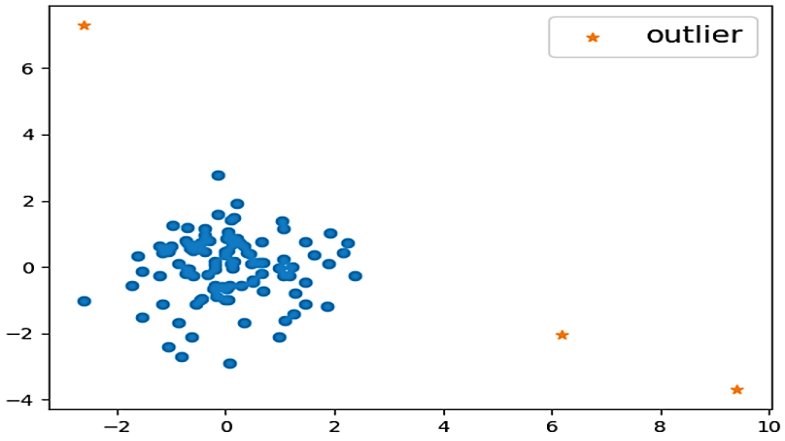
Модели кластеризации

K-means

Density-based Spatial Clustering of Applications with Noise (DBSCAN)

Balanced Iterative Reduction and Clustering using Hierarchies (BIRCH)

Fuzzy C-Means



Модели глубокого обучения

Recurrent Neural Network (RNN)

Autoencoder (AE)

Long Short-Term Memory (LSTM)

Gated Recurrent Unit (GRU)

Temporal Convolutional Networks (TCN)

Generative Adversarial Network (GAN)

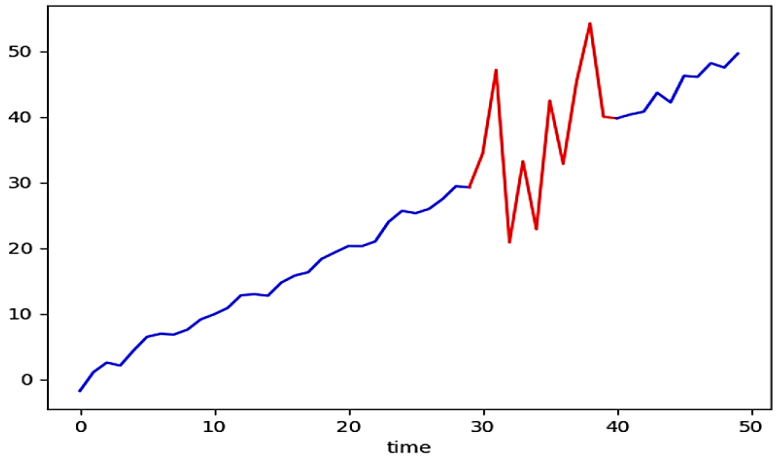
CNN-LSTM

Convolutional Neural Networks (CNN)

Graph Convolutional Network (GCN)

Graph Neural Networks (GNN)

Graph Attention Network (GAT)



Модели кластеризации могут использоваться в качестве неконтролируемого механизма обучения для обнаружения закономерностей или групп событий в данных. Оценка аномалии обычно выполняется на основе расстояния между подпоследовательностями и центрами кластеров. Таким образом, можно идентифицировать выбросы в данных. **Модели глубокого обучения** хорошо зарекомендовали себя при обработке многомерных временных рядов. RNN и автоэнкодеры хороши для обнаружения аномалий во временных рядах, но в основном они способны идентифицировать точечные аномалии на основе указанных временных шагов. В свою очередь, сверточные и графовые нейронные сети позволяют анализировать внутреннюю структуру данных, такую как матрицы и графы.

Содержание

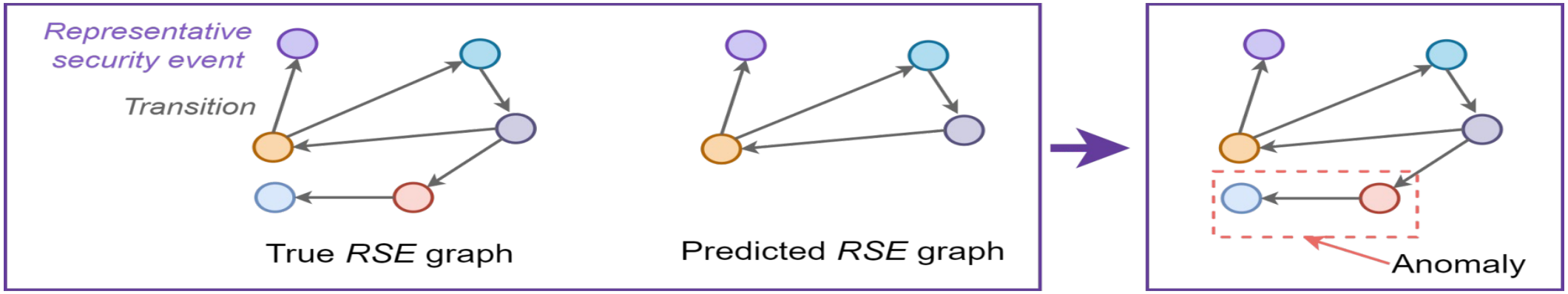
- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Сущность подхода

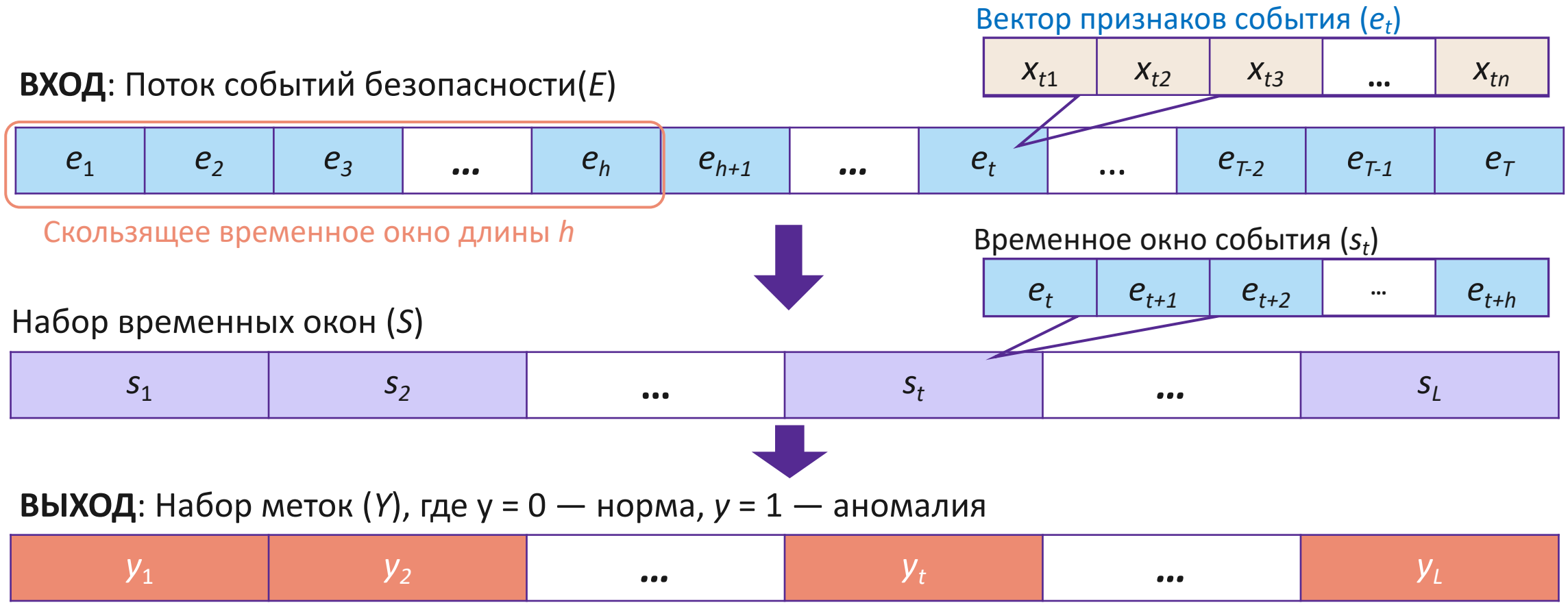
Предлагается подход к обнаружению аномалий, использующий причинно-следственную интеллектуальную корреляцию событий безопасности, который объединяет кластерный анализ и глубокое обучение**.**

- Кластерный анализ помогает определить отношения сходства событий друг с другом и построить графовые модели корреляции событий безопасности. Такой граф является репрезентативным графом событий безопасности (**Representative Security Event graph**, RSE graph).
- Сверточная рекуррентная нейронная сеть - сверточная LSTM (ConvLSTM) - анализирует **пространственно-временные связи событий**.

Цель: обнаруживать аномалии путем выявления закономерностей событий безопасности для нормального поведения и поиска отклонений от них**.** Обнаружение аномалий основано на вычислении ошибки реконструкции (**reconstruction error**) при прогнозировании RSE-графов с течением времени.



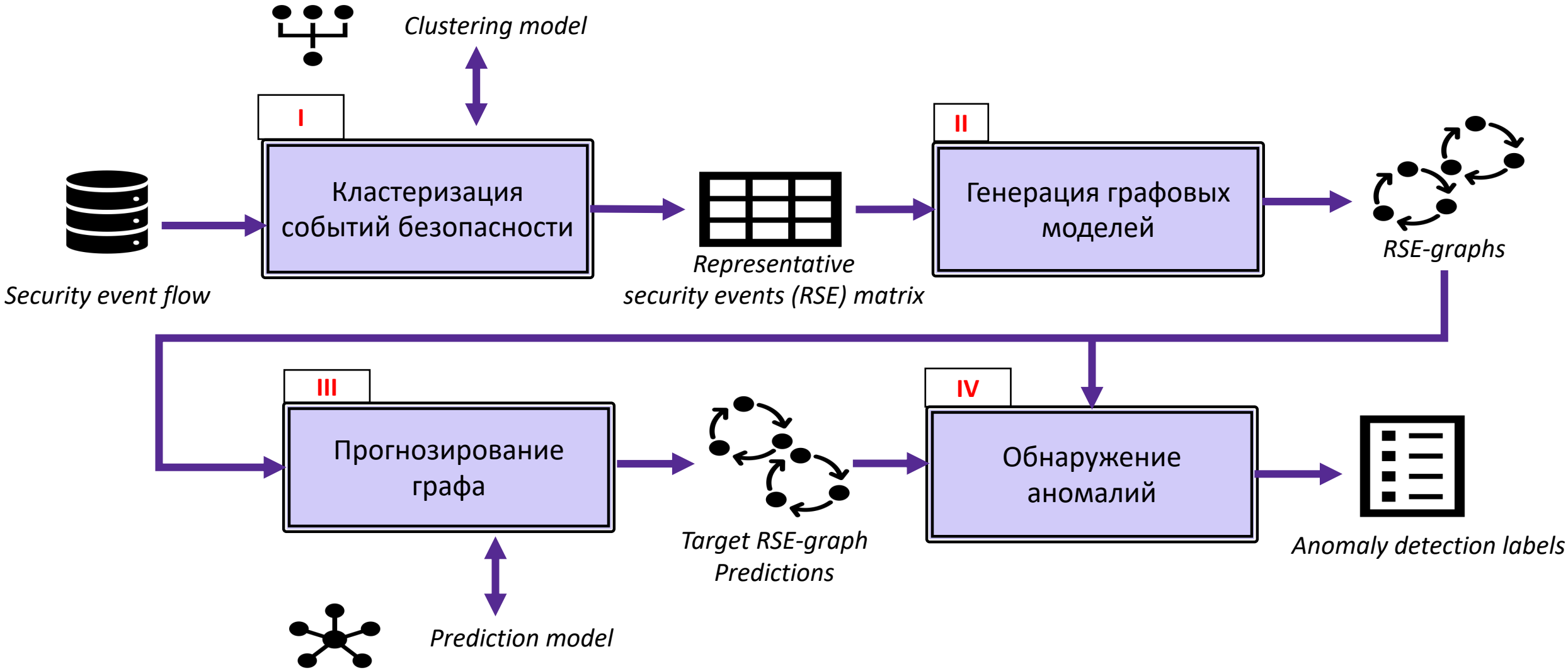
Постановка задачи



В сети IoT события безопасности обычно собираются с нескольких датчиков и представляют собой многомерный временной ряд. Результат обнаружения аномалий вычисляется не поточечно для событий, а для скользящего временного окна (длиной h). Выходом предлагаемого метода является набор меток, используемых для отображения результата обнаружения аномалий для каждого временного окна заданной длины.

T – длина потока событий безопасности
x_{ti} – значение i-го признака в момент времени t
n – количество признаков
L – количество временных окон

Схема реализации предлагаемого подхода



Основные этапы предлагаемого подхода

I. Кластеризация событий безопасности (выявление репрезентативных событий безопасности RSE):

1. Предварительная обработка данных
2. Реализация алгоритма BIRCH
3. Отображение RSE

II. Генерация графовых моделей (определение переходов между репрезентативными событиями для каждого временного окна):

4. Создание скользящих окон
5. Частотный анализ переходов
6. Генерация графа

III. Подготовка модели прогнозирования графа (обучение сверточной LSTM на кадрах RSE-графа):

7. Создание кадров графа
8. Обучение ConvLSTM
9. Прогнозирование кадров целевого графа
10. Извлечение целевых графов

IV. Обнаружение аномалий (использование ошибок реконструкции модели и векторных расстояний между событиями и центрами кластеров для поиска отклонений):

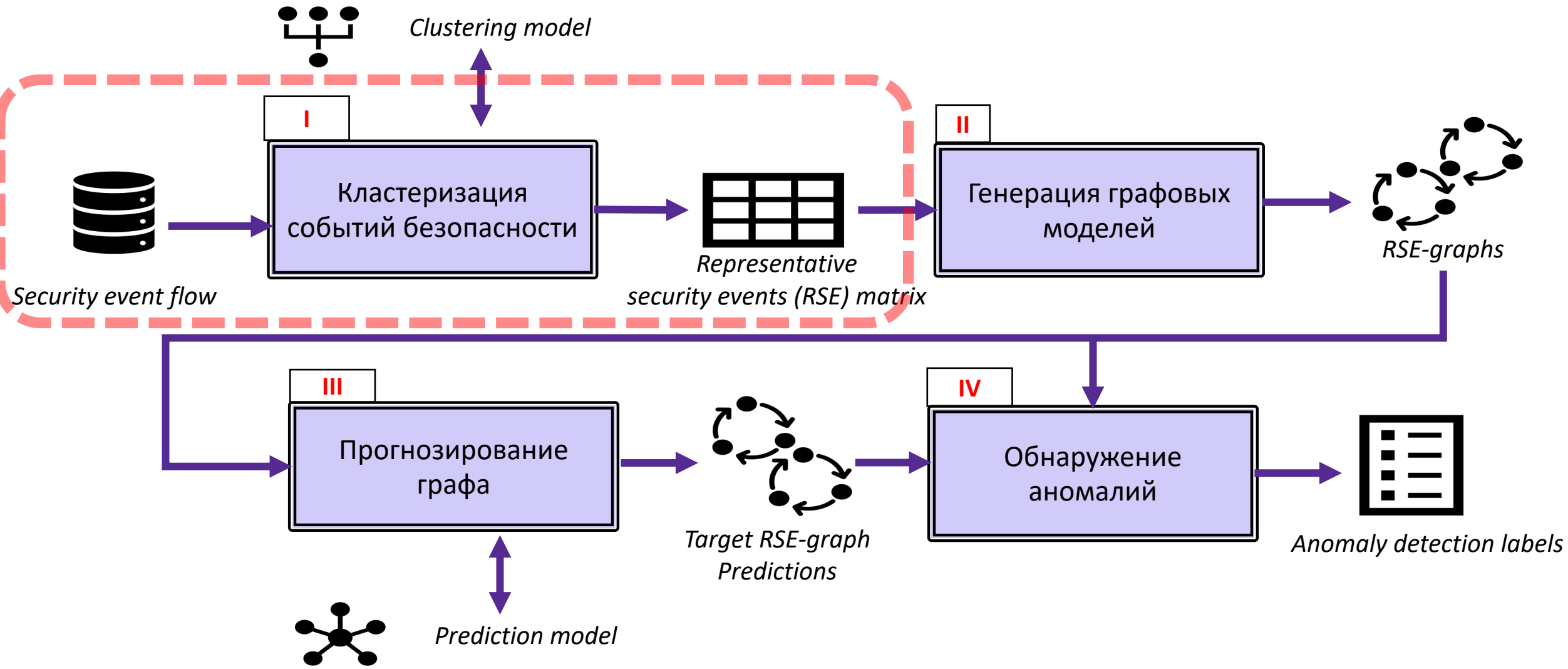
11. Вычисление ошибки реконструкции графа
12. Расчет расстояний между RSE и исходным событием
13. Расчет среднего расстояния для временного окна
14. Сравнение с пороговыми значениями

Содержание

- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Схема реализации предлагаемого подхода

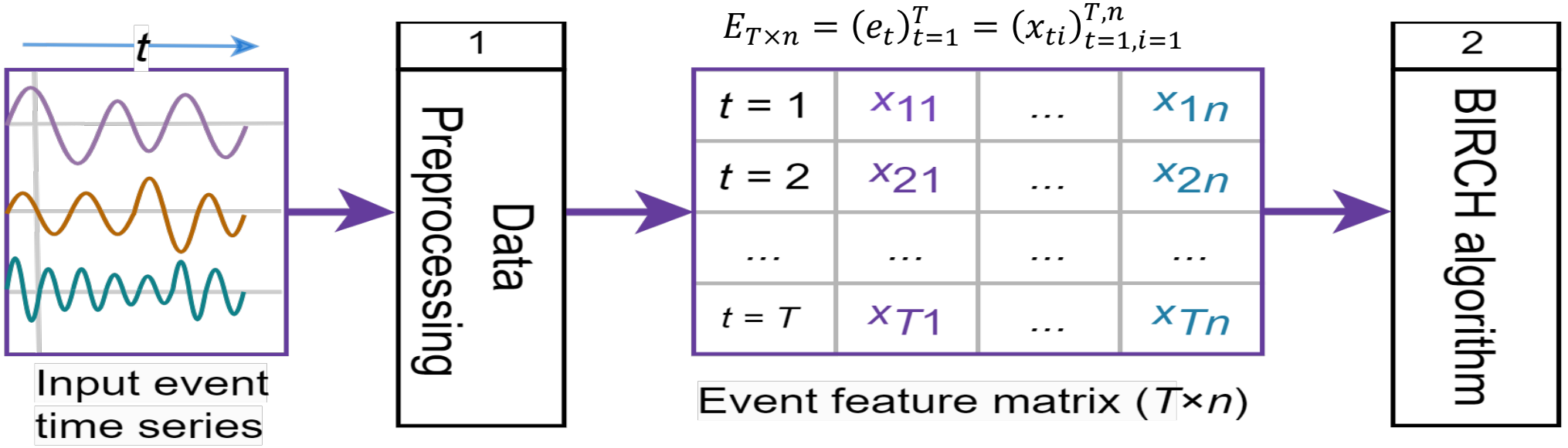
Выявление репрезентативных событий безопасности



Фаза I: Кластеризация событий безопасности [1/4]

Предварительная обработка данных (шаг 1) состоит из извлечения признаков и нормализации. Результатом является матрица признаков событий безопасности.

BIRCH (шаг 2) упрощает иерархическую кластеризацию больших наборов данных и не требует знания точного количества кластеров, в отличие от k-средних и спектральной кластеризации.



В качестве метрики сходства используется евклидово расстояние между векторами событий безопасности.

Schubert, Erich; Lang, Andreas (2022), "5.1 Data Aggregation for Hierarchical Clustering", *Machine Learning under Resource Constraints - Fundamentals*, De Gruyter, pp. 215–226.

T – длина потока событий безопасности
x_{ti} – значение i-го признака в момент времени t
n – количество признаков
BIRCH – Balanced Iterative Reduction and Clustering using Hierarchies (Сбалансированное итеративное сокращение и кластеризация с использованием иерархий)

Фаза I: Кластеризация событий безопасности [2/4]

Признак кластеризации (CF) с использованием BIRCH:

$$CF = (N, LS, SS),$$

N – размер кластера (количество элементов)
LS – векторная сумма точек данных
SS – сумма квадратов точек данных

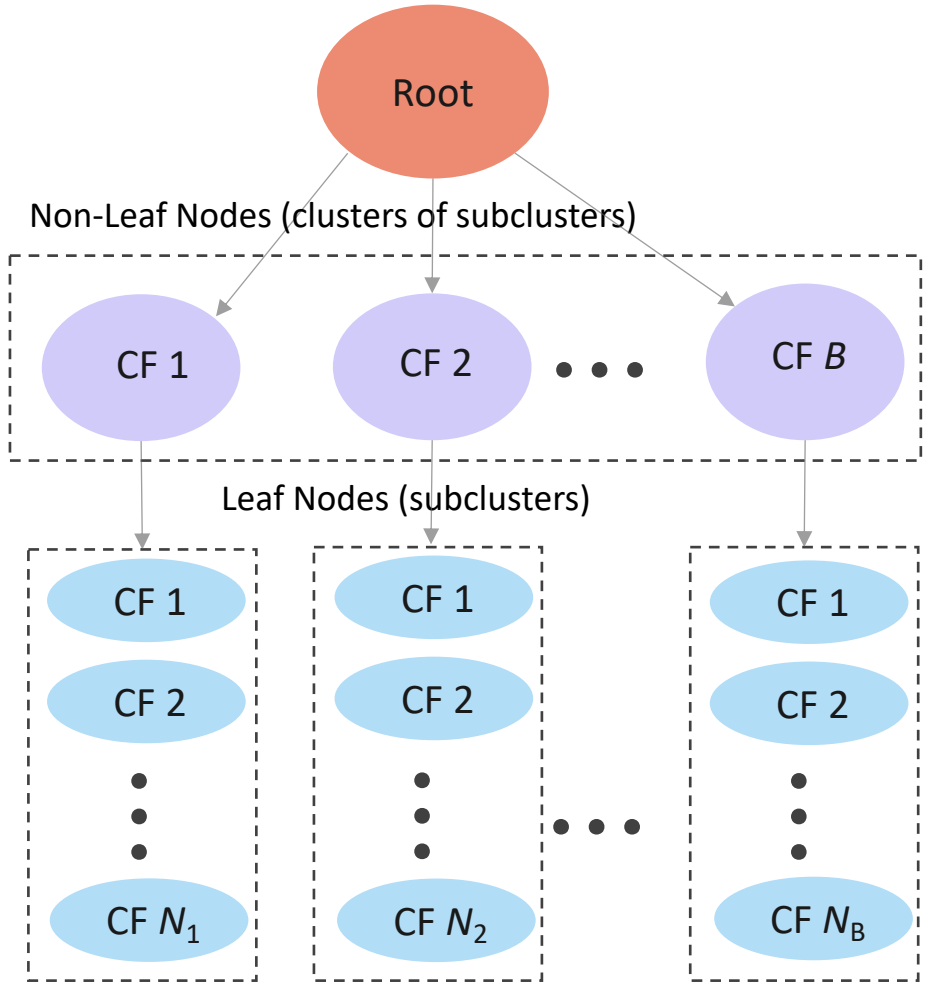
CF-дерево характеризуется двумя параметрами:

- ❑ *B* – **коэффициент ветвления**, максимальное количество нелистовых узлов
- ❑ *λ* – **порог**, максимальный диаметр подкластеров в листовых узлах

Для каждого вектора событий во входных данных:

1. Найти ближайшую запись листа
2. Добавить вектор к листовой записи и обновить CF
3. Если диаметр входа > λ, то лист расщепляется, и, возможно, также родители

CF-дерево: Дерево балансировки высот, которое содержит CFs



Алгоритм BIRCH строит дерево признаков кластеризации (CF) из точек данных — сбалансированное по высоте дерево с двумя параметрами: коэффициентом ветвления и пороговым значением.

Фаза I: Кластеризация событий безопасности [3/4]

Предлагаемая оптимизация порога BIRCH :

$$\max(CIC) = \max(CHI \times (1 - DBI)),$$

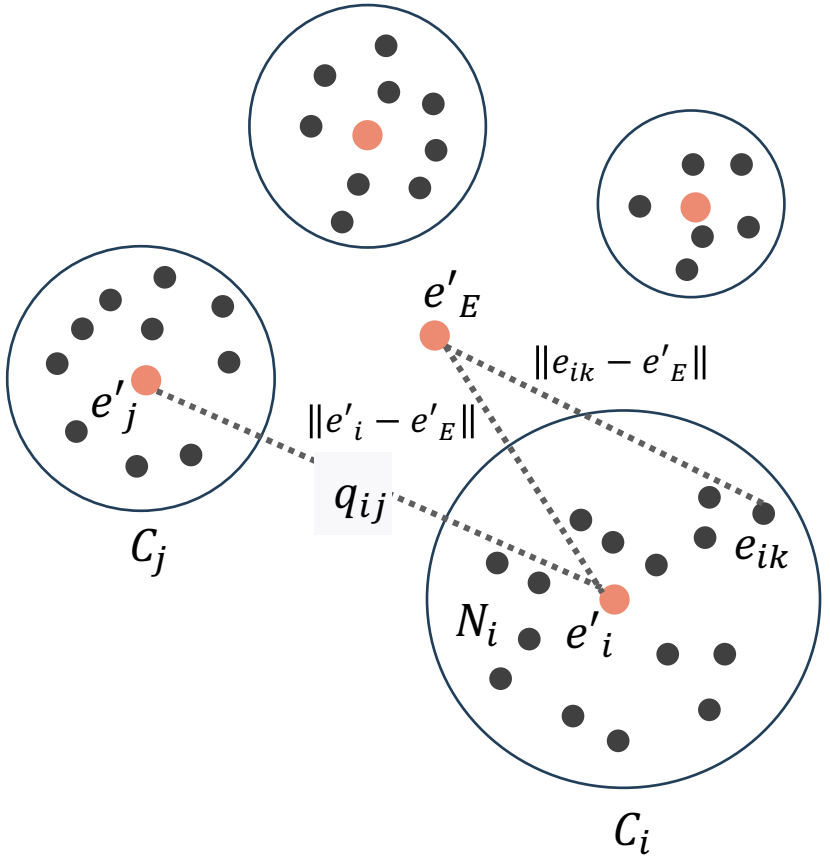
CIC – предлагаемая интегральная оценка кластера

CHI – индекс Калински-Харабаша (сумма межкластерной дисперсии и внутрикластерной дисперсии):

$$CHI = \frac{\sum_{i=1}^{m} N_i \times \|e'_i - e'_E\|^2}{\sum_{i=1}^{m} \sum_{k=1}^{N_i} \|e_{ik} - e'_E\|^2} \times \frac{T - m}{m - 1}$$

DBI – индекс Дэвиса-Боулдина (среднее сходство между кластерами):

$$DBI = \frac{1}{m} \sum_{i=1}^{m} \max_{i \neq j} \frac{D_i - D_j}{q_{ij}}$$



m - количество кластеров

N_i – количество событий в кластере C_i

e'_i – центроид кластера C_i

e'_E – центроид набора данных событий

D_i и D_j – *средние расстояния между каждой точкой в Ci и Cj*

T – длина потока событий безопасности

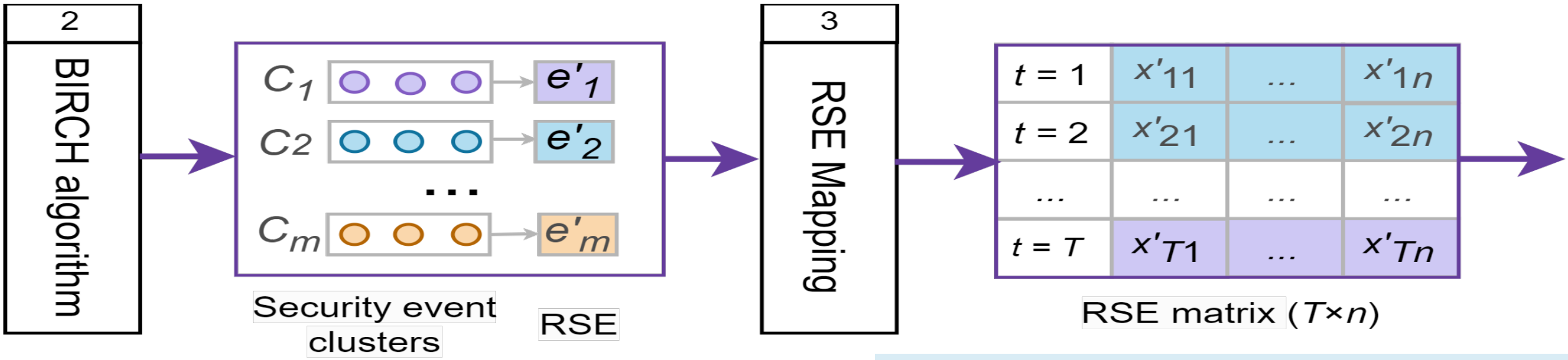
qij – расстояние между e'_i и e'_j

$\| \|$ – Евклидово расстояние

Фаза I: Кластеризация событий безопасности [4/4]

RSE (representative security event, репрезентативное событие безопасности) – центроид кластера (определенная группа подобных событий)

$$e' = \{x'_1, x'_2, \ldots, x'_n\}, \qquad x'_i = \frac{\sum_{j=1}^{N_i} x_{ij}}{N_i}$$



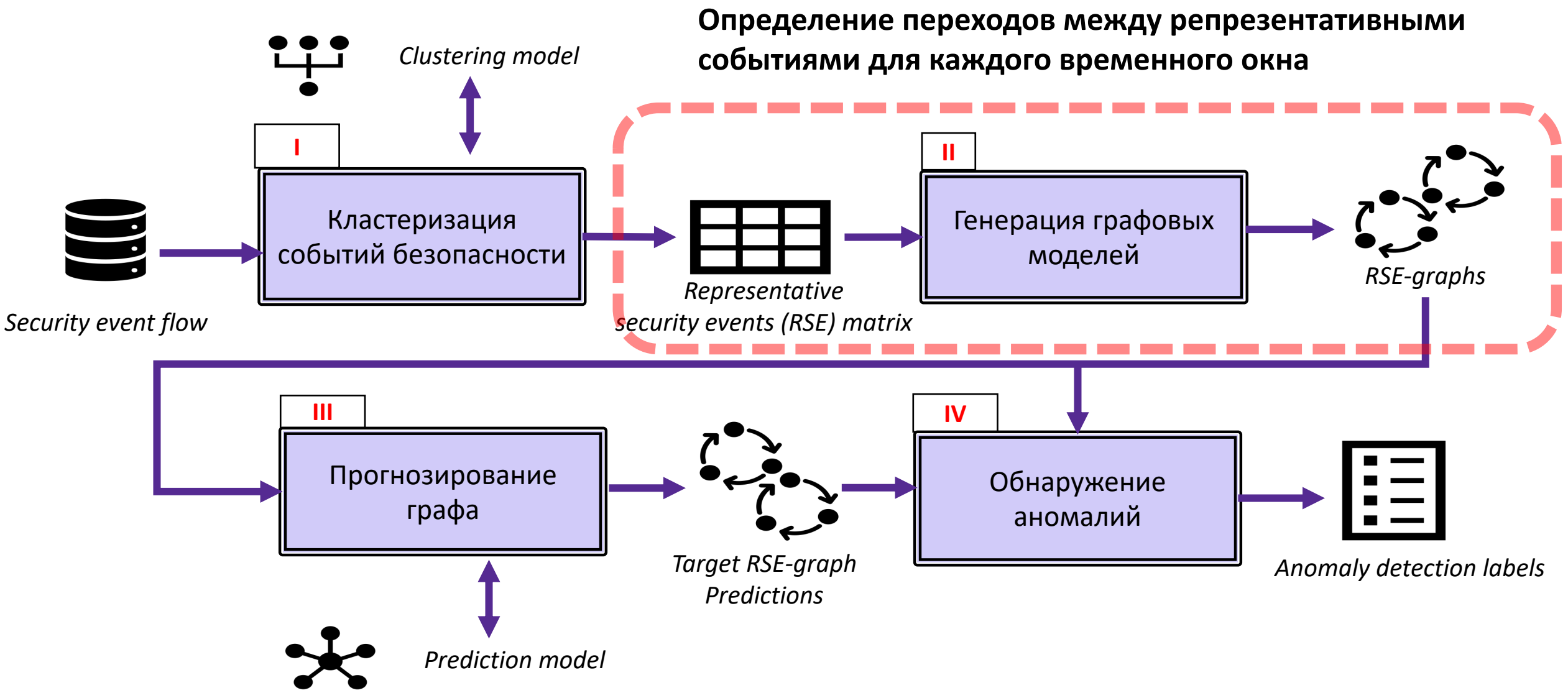
Отображение RSE (шаг 3) представляет собой сопоставление матрицы векторов событий безопасности с полученными репрезентативными событиями (матрицей RSE).

m – количество кластеров
n – количество признаков
T – длина потока событий безопасности
x'_{ti} – значение i-го признака RSE в момент времени *t*

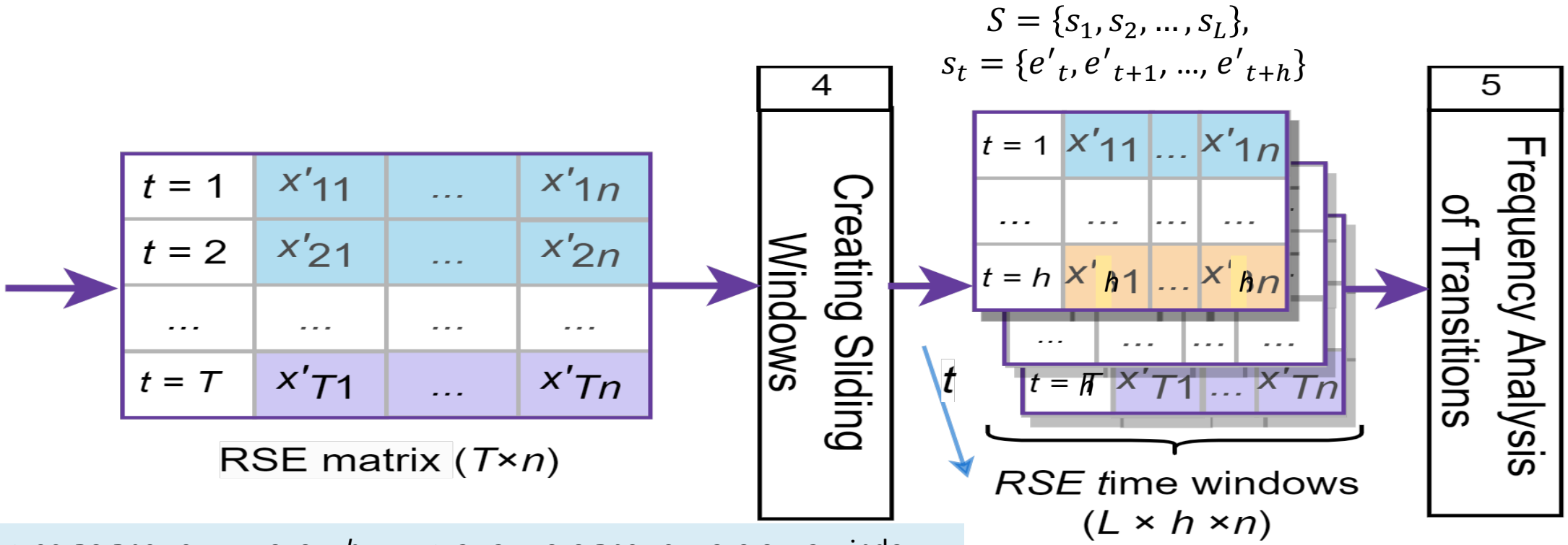
Содержание

- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Схема реализации предлагаемого подхода



Фаза II: Генерация графовых моделей [1/3]



L – количество временных окон, *h* – длина каждого временного окна window

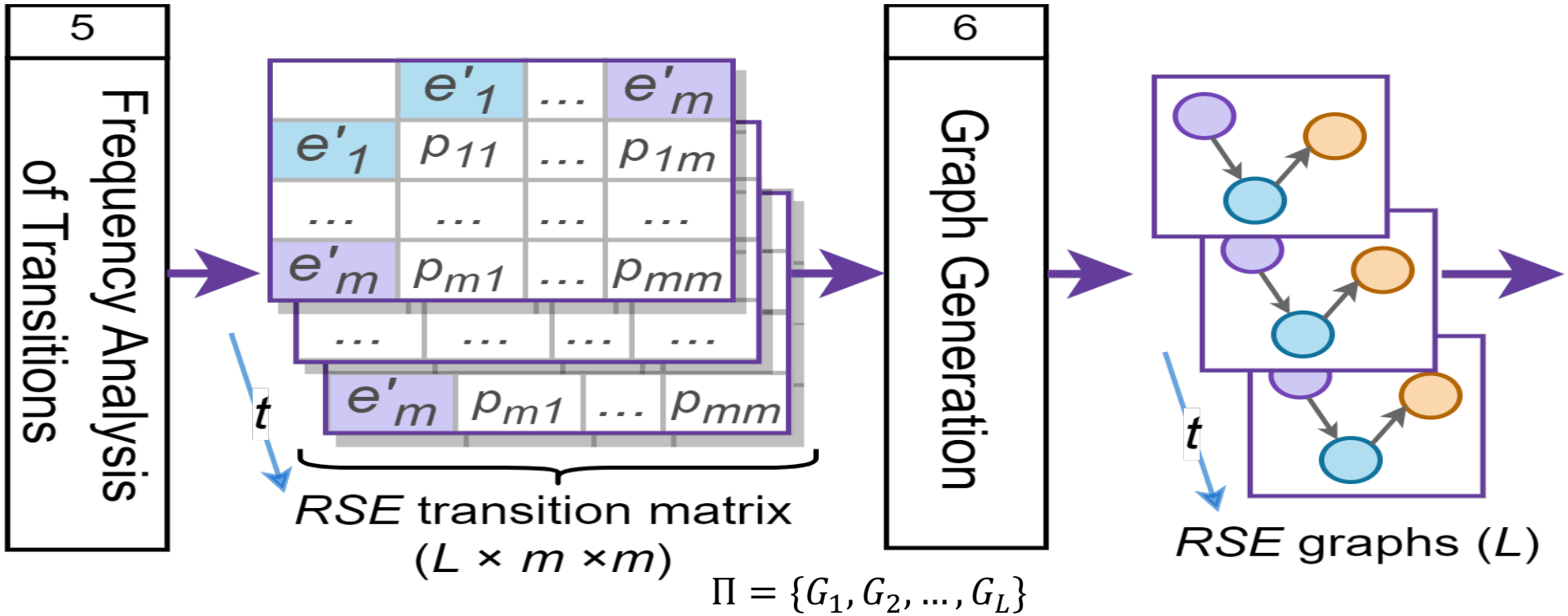
- На этапе II строится временная последовательность RSE-графов. Каждый такой граф определяет переходы между RSE в заданном временном окне. Переходам соответствуют вероятности переходов.
- Для расчета частоты переходов между парой репрезентативных событий безопасности используется метод скользящего окна (шаг 4).

Фаза II: Генерация графовых моделей [2/3]

- На шаге 5 вычисляются вероятности переходов между репрезентативными событиями в последовательности.
- Результатом этого шага является набор матриц перехода RSE для каждого временного окна.
- Граф формируется для каждого временного окна (шаг 6), другими словами, для каждой матрицы.

$$P = \{P_1, P_2, \dots, P_L\} \qquad p_{ij} = \frac{\sum r(e'_i, e'_j)}{\sum_{k=1}^{K} \sum r(e'_i, e'_j)}$$

$$P_i = (p_{ij})_{i=1,j=1}^{m,m}$$



$$\Pi = \{G_1, G_2, \dots, G_L\}$$

$r(e'_i, e'_j)$ – переход между e'_i и e'_j
$\sum r(e'_i, e'_j)$ – число переходов между e'_i и e'_j
K – число событий, в которые e'_i входит в заданной последовательности, $K \le m$

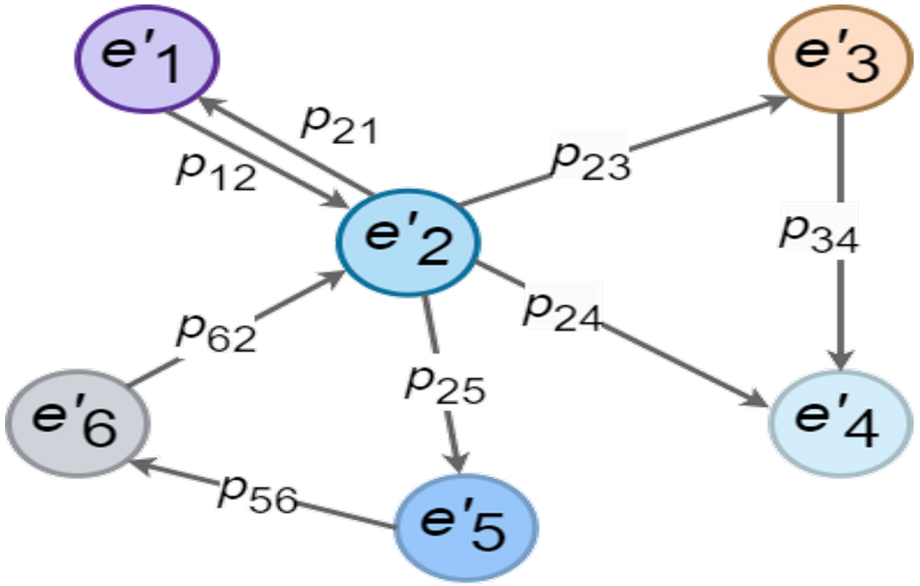
Фаза II: Генерация графовых моделей [3/3]

RSE-граф: $G =< V, R, f >,$

$V = \{e'_1, e'_2, .., e'_m\}$ – набор репрезентативных событий в виде узлов графа

$R = \{r(e'_i, e'_j)\}_{i=1,j=1}^{m,m}$ – набор возможных переходов между репрезентативными событиями в виде ребер графа

$f: r(e'_i, e'_j\} \rightarrow p_{ij}$ – функция, которая сопоставляет ребра с их весами (вероятностью перехода)

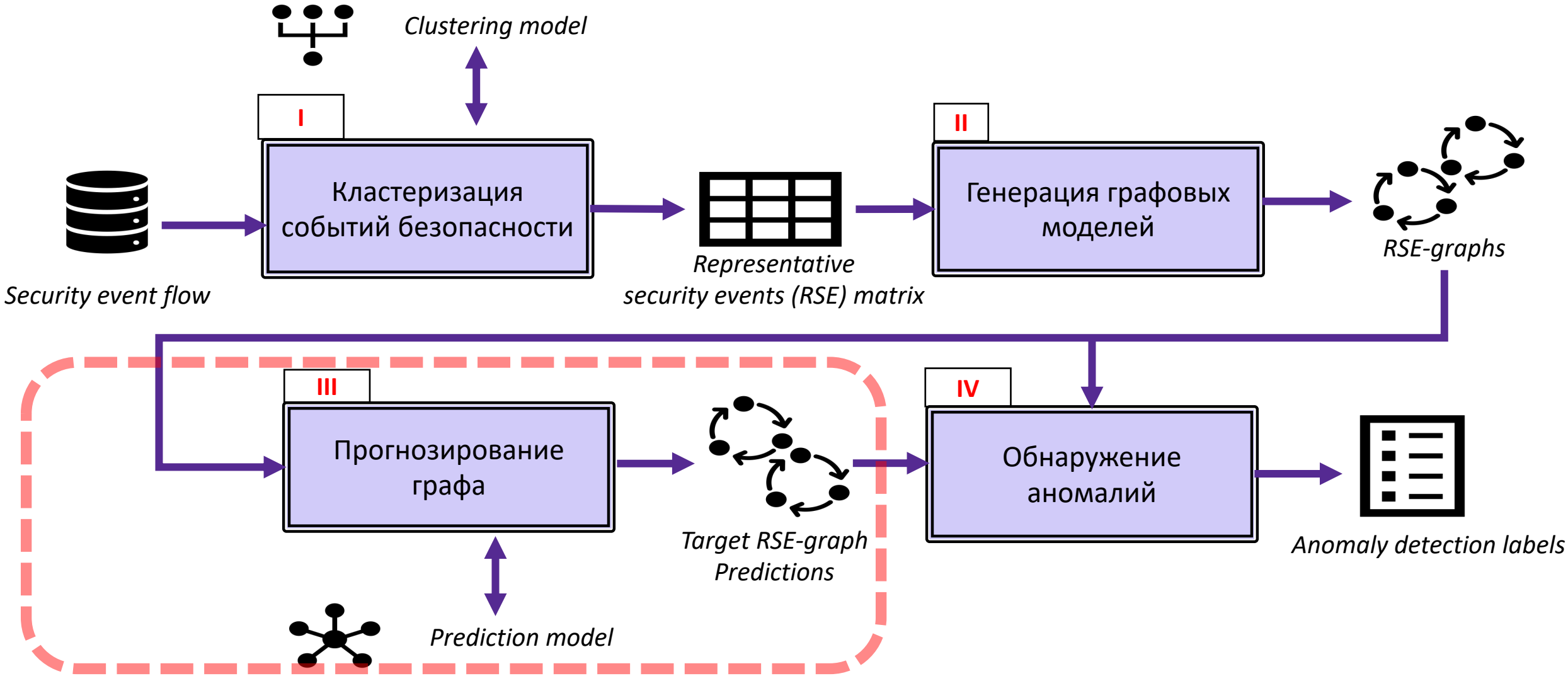


- RSE-граф можно представить как направленный граф.
- **Выбор этого типа графа** основан на идее, что типичное событие представляет собой группу сопоставимых событий, а отслеживание переходов между такими событиями позволяет фиксировать поведение системы.
- Отклонения от нормальных вероятностей переходов между репрезентативными событиями могут отражать **аномалии**.

Содержание

- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Схема реализации предлагаемого подхода



Обучение сверточной LSTM на кадрах RSE-графа

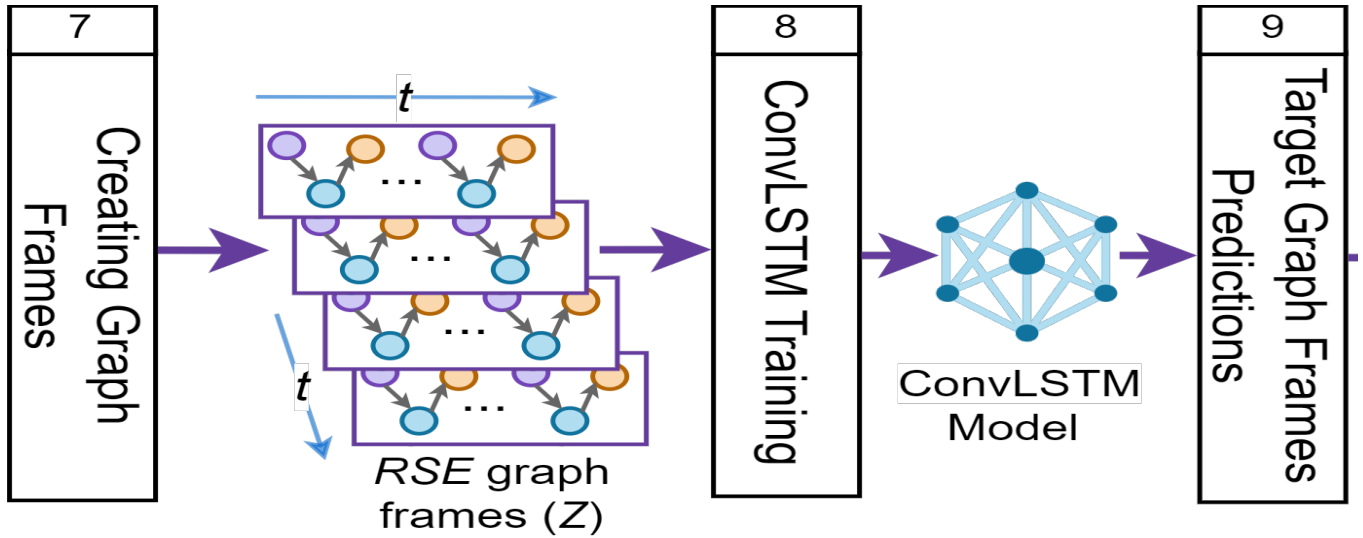
Фаза III: Прогнозирование графа [1/4]

- **Сверточная модель LSTM (ConvLSTM – Convolutional Long Short-Term Memory)** обучена для предсказания следующего кадра RSE-графа (предсказания того, какие графы будут следующими, учитывая последовательность прошлых графов).
- **Кадр графа** представляет собой короткую последовательность *w* RSE-графов. Последовательность кадров формируется аналогично с использованием метода скользящего окна (**шаг 7**):

$$Q = \{q_1, q_2, \dots, q_Z\}, \;\; q_t = \{G_t, \dots, G_{t+\omega-1}\},$$

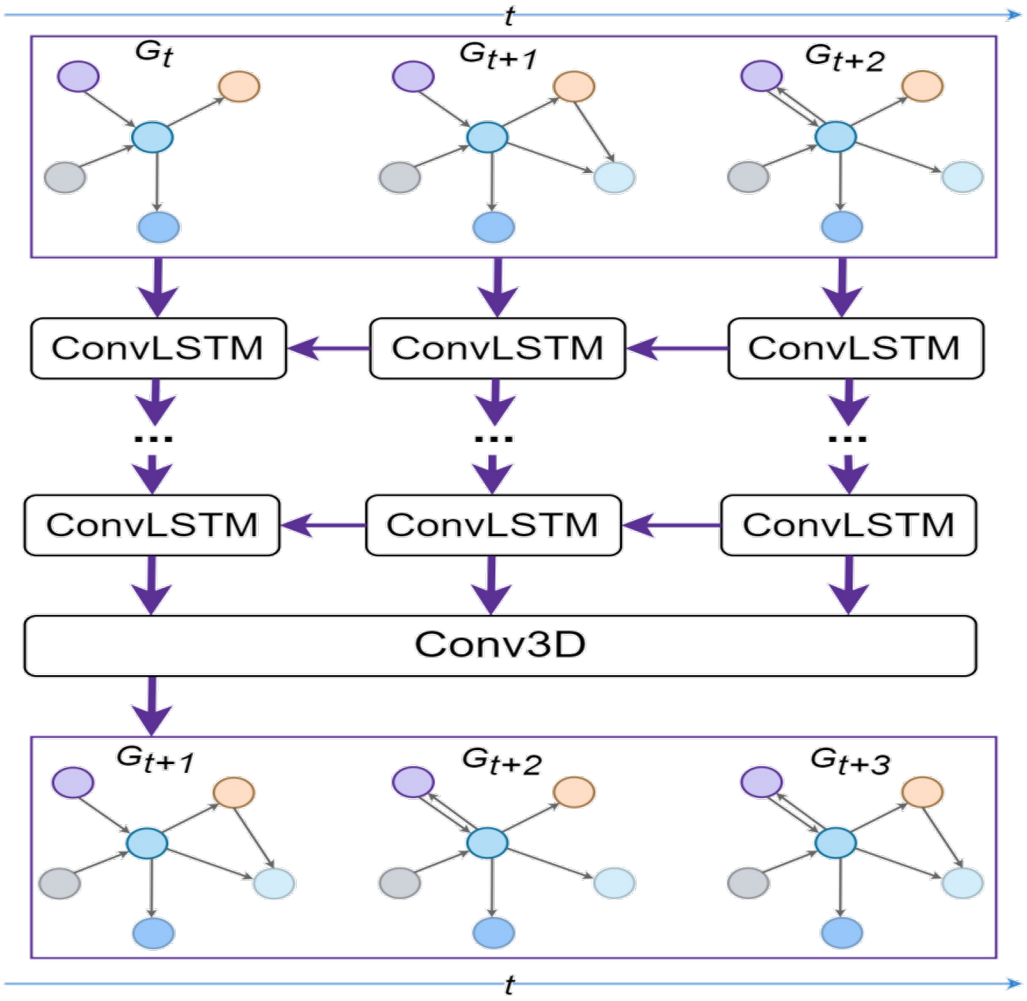
Z = L − w – длина последовательности кадров графа

- Таким образом, данные состоят из **последовательностей кадров графа**, каждый из которых используется для прогнозирования следующего кадра.
- В этом случае рассматривается **последовательность *w* графов для прогнозирования следующей последовательности *w* графов**, и графы предсказываются не по одному. Это позволяет рассматривать более длительные временные связи между RSE-графами.



Фаза III: Прогнозирование графа [2/4]

- При обучении **модели ConvLSTM** входными данными является репрезентативное описание структуры каждого RSE-графа в матричной форме (**шаг 8**).
- **Входные и выходные данные, а также скрытые состояния и вентили ConvLSTM** являются 3-мерными тензорами: (w, m, m).
- Модель ConvLSTM **определяет будущее состояние** (**шаг 9**) конкретной ячейки в матрице графа из входных данных и прошлых состояний ее локальных соседей.
- Предсказанный 3-мерный кадр выводится через несколько слоев ConvLSTM и, наконец, через слой 3D-свертки (Conv3D).



Фаза III: Прогнозирование графа [3/4]

Формулы для одной ячейки ConvLSTM:

Input gate:

$$i_t = \sigma_g(W_{xi} * G_t + W_{H_i} * H_{t-1} + W_{ci} \circ c_{t-1} + b_i)$$

Forget gate:

$$f_t = \sigma_g(W_{xf} * G_t + W_{H_f} * H_{t-1} + W_{cf} \circ c_{t-1} + b_f)$$

Output gate:

$$o_t = \sigma_g(W_{xo} * G_t + W_{H_o} * H_{t-1} + W_{co} \circ c_{t-1} + b_o)$$

State of the current moment:

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_{xc} * G_t + W_{H_c} * H_{t-1} + b_c)$$

Final output:

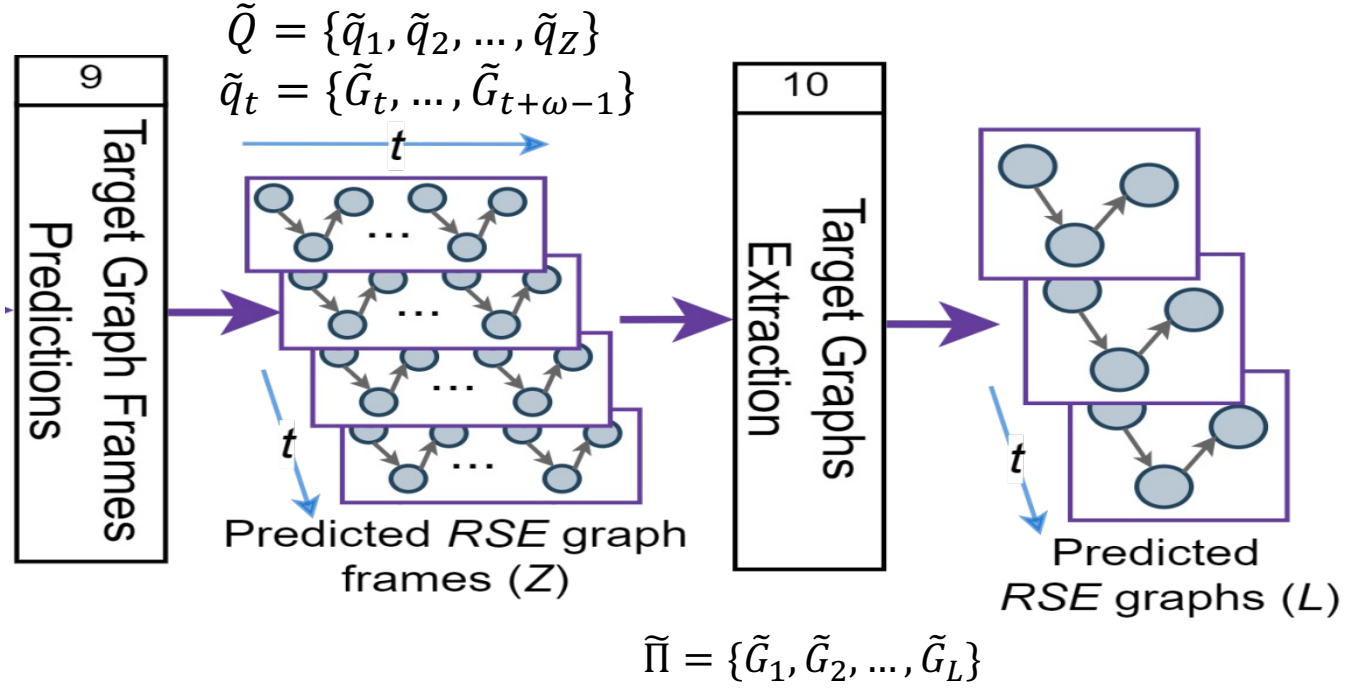
$$H_t = o_t \circ \sigma_c(c_t)$$

- **Разница между обычной LSTM и ConvLSTM** заключается в замене умножения матриц (произведение Адамара, обозначается «◦») на операцию свертки (обозначается «∗»).
- Модель ConvLSTM **определяет будущее состояние конкретной ячейки в матрице графа** из входных данных и прошлых состояний ее локальных соседей. Предсказанный 3-мерный кадр выводится через несколько слоев ConvLSTM и, наконец, через слой 3D-свертки (Conv3D).

c_{t-1} – состояние предыдущего момента
W – весовой коэффициент для заданного вентиля
σ_g – функция активации на основе сигмоиды
$\circ$ – произведение Адамара
b – коэффициент смещения для данного вентиля
σ_c – функция активации на основе гиперболического тангенса
$*$ – операция свертки

Фаза III: Прогнозирование графа [4/4]

- Модель ConvLSTM хорошо **подходит для прогнозирования графов**, поскольку она объединяет пространственные характеристики, полученные в результате операции свертки, с временной изменчивостью, полученной из LSTM.
- Это позволяет **эффективно представлять как пространственные, так и временные характеристики** последовательности RSE-графа.
- Результат прогнозирования (**шаг 10**) - последовательность графов $\tilde{\Pi}$.



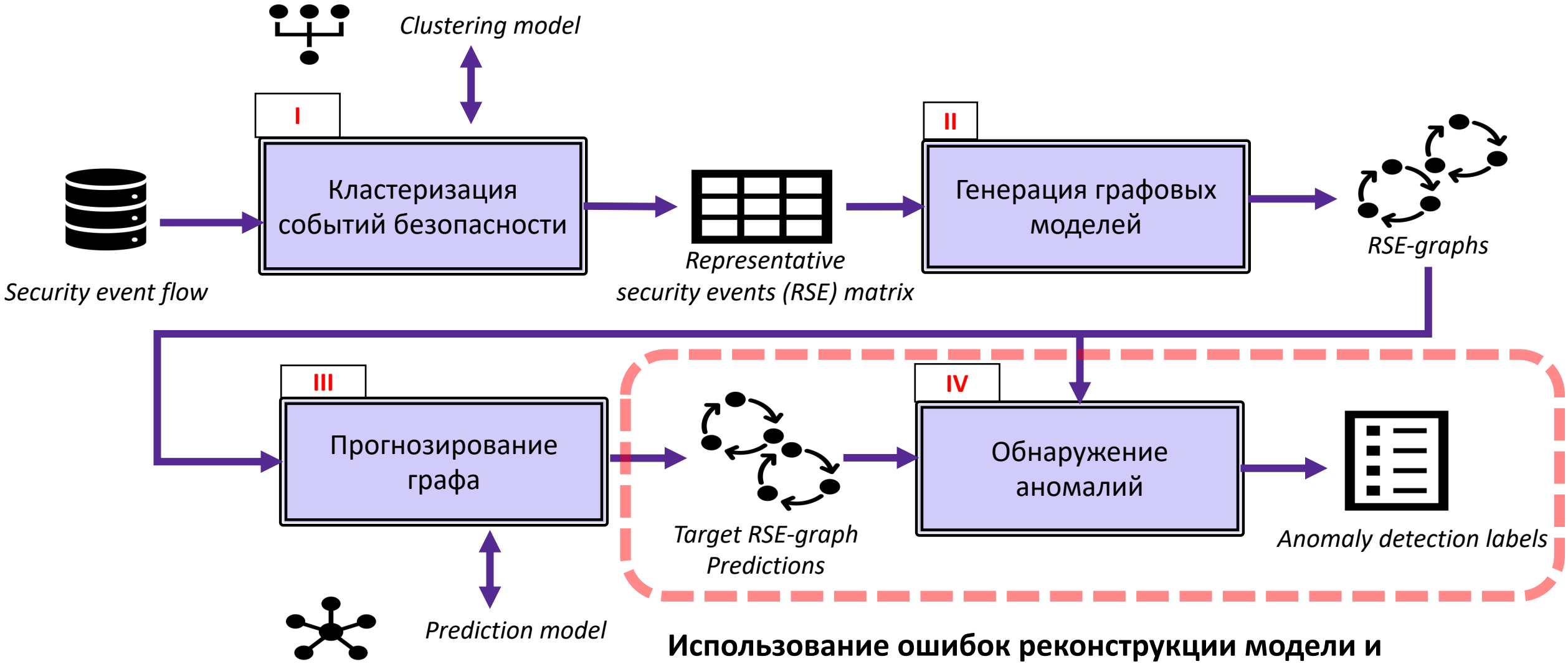
$\tilde{q}_t$ – кадр графика, предсказанный из кадра графа q_t

$\tilde{G}_t$ – RSE-граф, предсказанный из графа G_t

Содержание

- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Схема реализации предлагаемого подхода

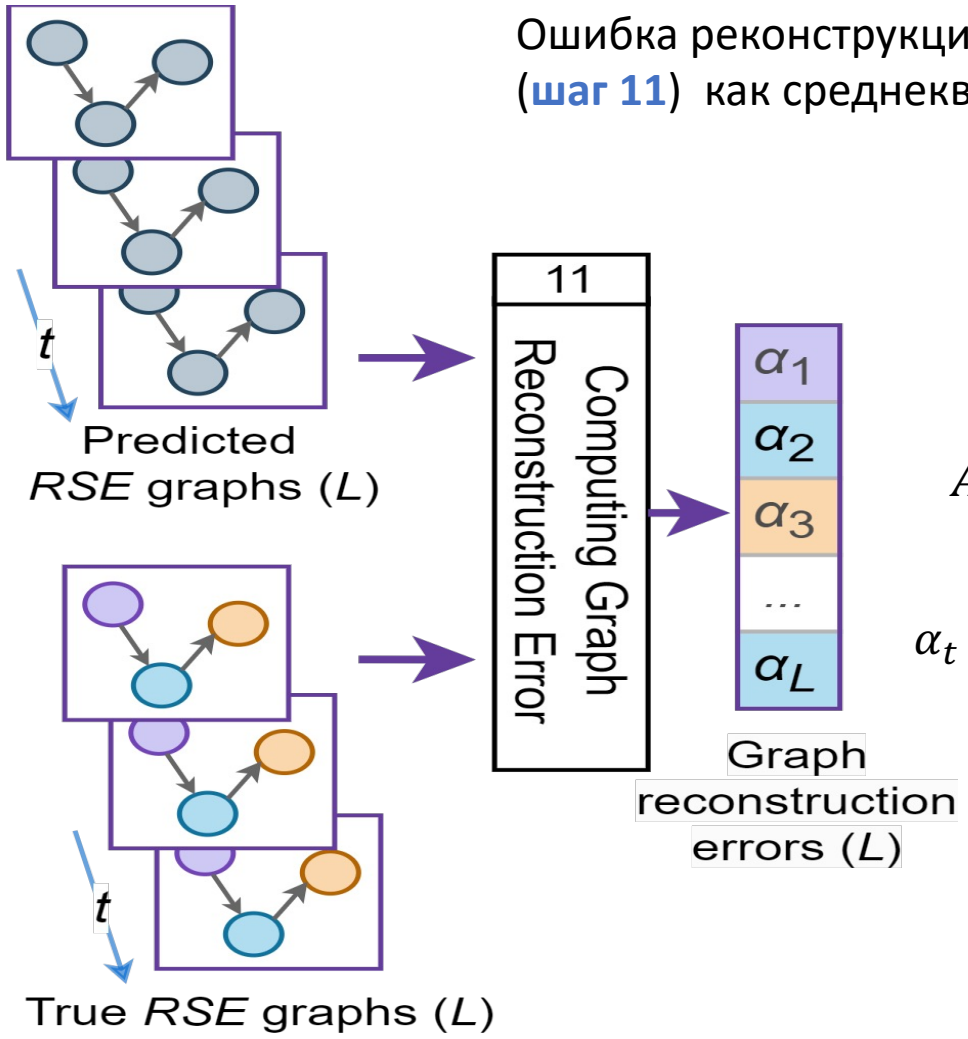


Использование ошибок реконструкции модели и векторных расстояний между событиями и центрами кластеров для поиска отклонений

Фаза IV: Обнаружение аномалий [1/3]

$\tilde{\Pi} = \{\tilde{G}_1, \tilde{G}_2, \dots, \tilde{G}_L\}$

$\Pi = \{G_1, G_2, \dots, G_L\}$



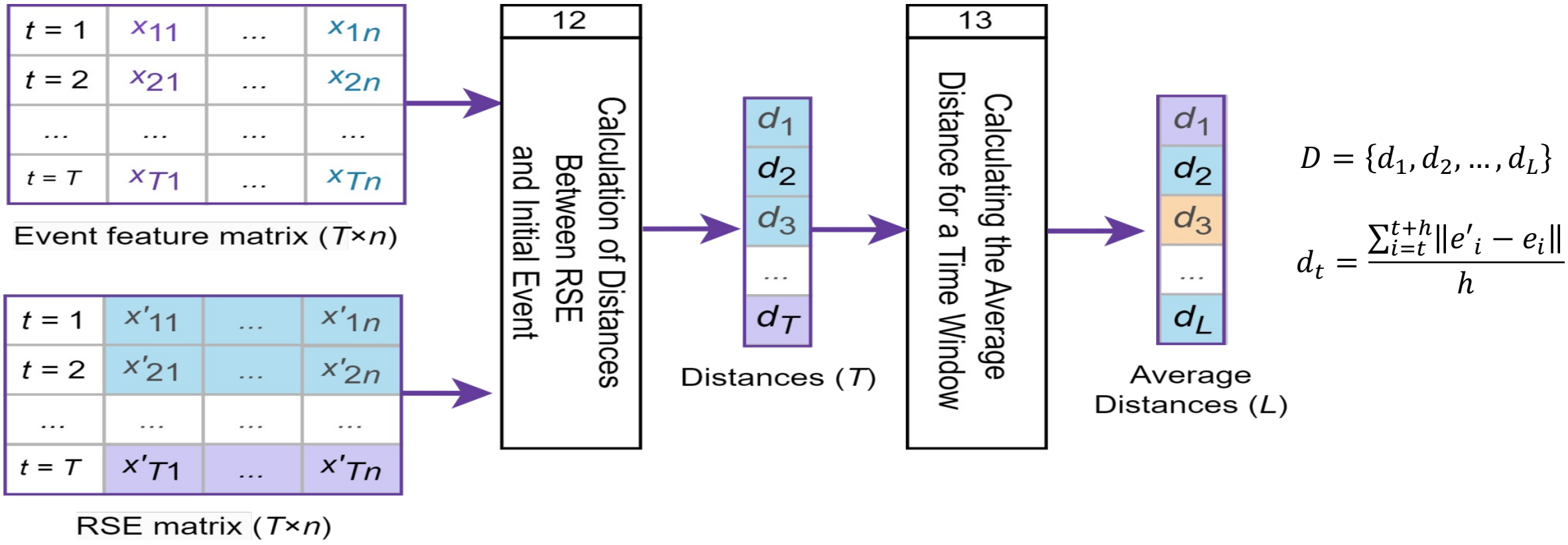
Ошибка реконструкции прогноза графа рассчитывается (**шаг 11**) как среднеквадратическая ошибка.

$A = \{\alpha_1, \alpha_2, \dots, \alpha_L\}$

$$\alpha_t = \frac{\sum_{t=1}^{L}\left(G_t - \tilde{G}_t\right)^2}{L}$$

Фаза IV: Обнаружение аномалий [2/3]

При расчете меток аномалий используются расстояния между событиями и центрами кластеров, к которым принадлежат события (**шаги 12, 13**). Чем меньше расстояние, тем более репрезентативным можно считать событие. Это также помогает определить новизну события.

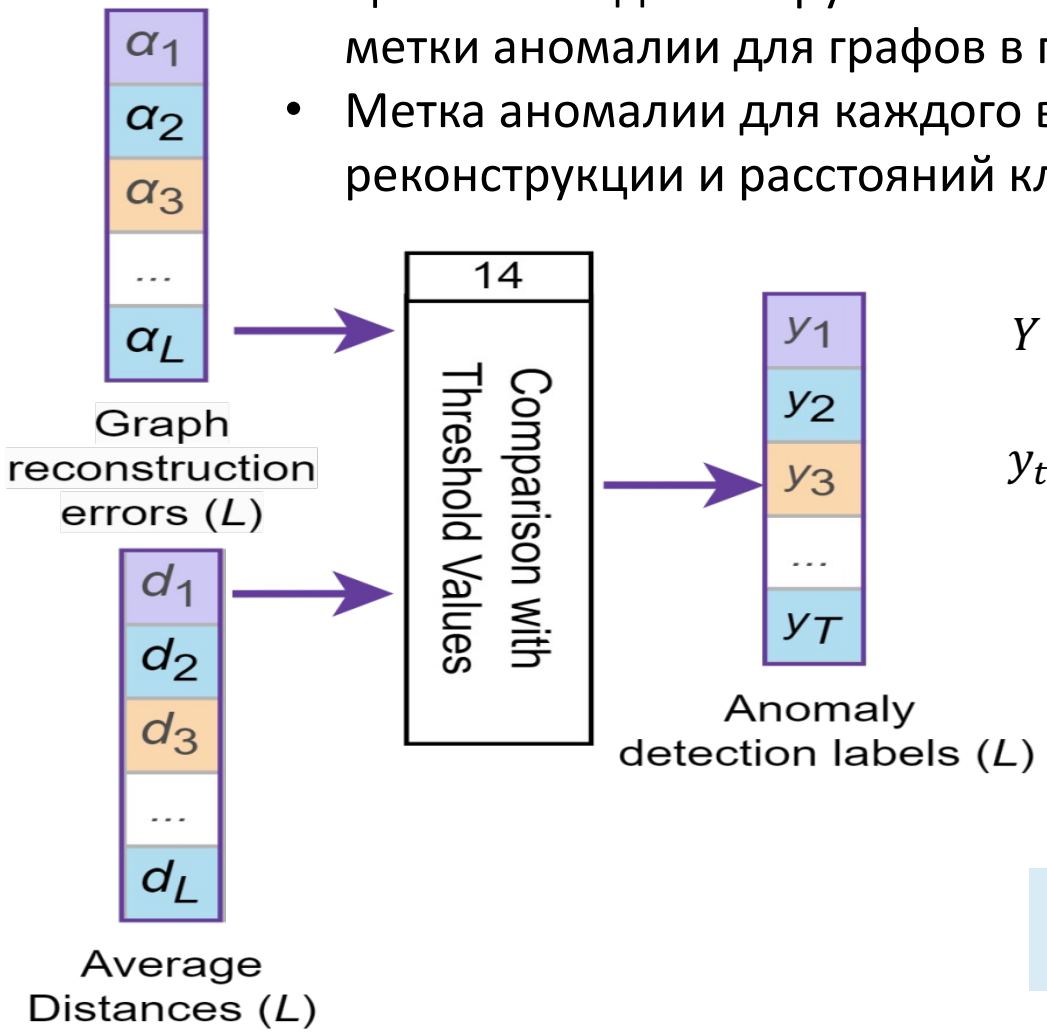


$$D = \{d_1, d_2, \dots, d_L\}$$

$$d_t = \frac{\sum_{i=t}^{t+h} \|e'_i - e_i\|}{h}$$

Фаза IV: Обнаружение аномалий [3/3]

- Целью метода обнаружения аномалий (**шаг 14**) является прогнозирование бинарной метки аномалии для графов в последовательности.
- Метка аномалии для каждого временного окна получается путем сравнения ошибок реконструкции и расстояний кластеризации с заданными пороговыми значениями.



$$Y = \{y_1, y_2, \dots, y_L\}$$

$$y_t = \begin{cases} 1, & if\ (\alpha_t > \epsilon)\ or (d_t > \delta) \\ 0, & overwise \end{cases}$$

ϵ – порог ошибки реконструкции
δ – порог расстояния кластеризации

Содержание

- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Наборы данных, цифровые двойники, стенды

SUTD: SWAT, WADA, EPIC, IoT

Сколтех: SKAB

Harvard: TEP

Numenta: NAB

Dataset Requests (2017 – 2021)

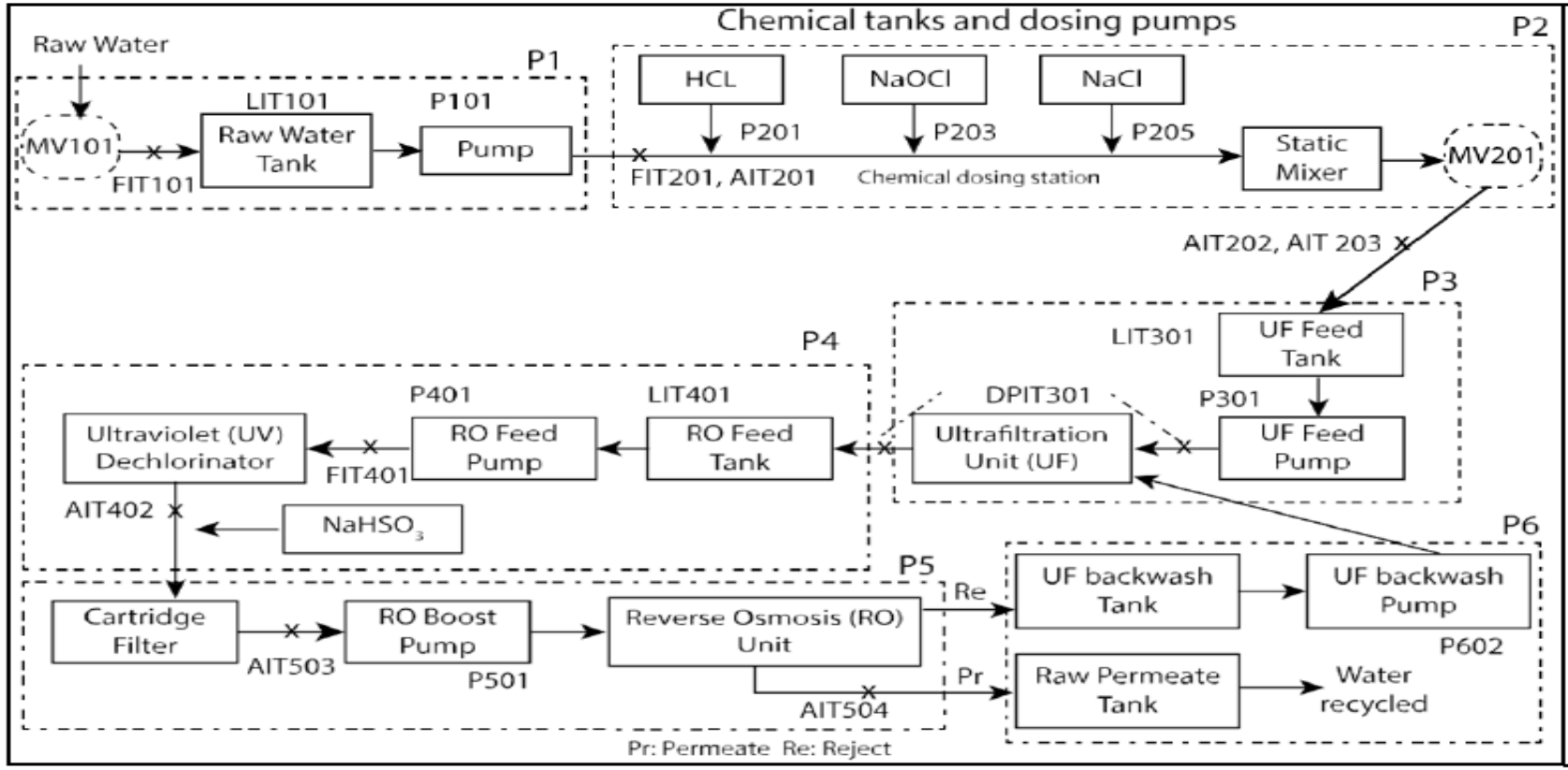
S/N	Name	Organisation	Origin of Request	Date of request	Dataset requested
933	Peng Kang	Southeast University	China	25-Dec-21	SWaT, WADI
1932	Li Yunpeng	Southwestern University of Finance and Economics	China	25-Dec-21	SWaT, WADI
1931	Meng Wei Xie	Fudan university	China	23-Dec-21	SWaT, WADI, IoT
1817	Liudmila Kopeikina	Eötvös Loránd University	Hungary	9-Nov-21	SWaT
1816	Chernyshov Yury	Ural Security Systems Center	Russia	8-Nov-21	SWaT
1815	Nozima Murodova	Inha University in Tashkent	Uzbekistan	8-Nov-21	SWaT
1814	Wu Jihua	Beijing University of Post and Telecommunication	China	8-Nov-21	SWaT, WADI
1813	Francesco Simone	Sapienza University of Rome	Italy	8-Nov-21	SWaT, WADI, CISS, BATADAL
1812	Tianhao Chen	Shandong University	China	8-Nov-21	SWaT, WADI, EPIC, CISS, Blaq_0, BATADAL, IoT
1811	Harsh Gupta	Indian Institute of Information Technology,	India	7-Nov-21	SWaT, WADI

https://itrust.sutd.edu.sg/

https://www.youtube.com/watch?v=i4vCG4clNZQ

Набор данных Safe Water Treatment (SWaT) [1/4]

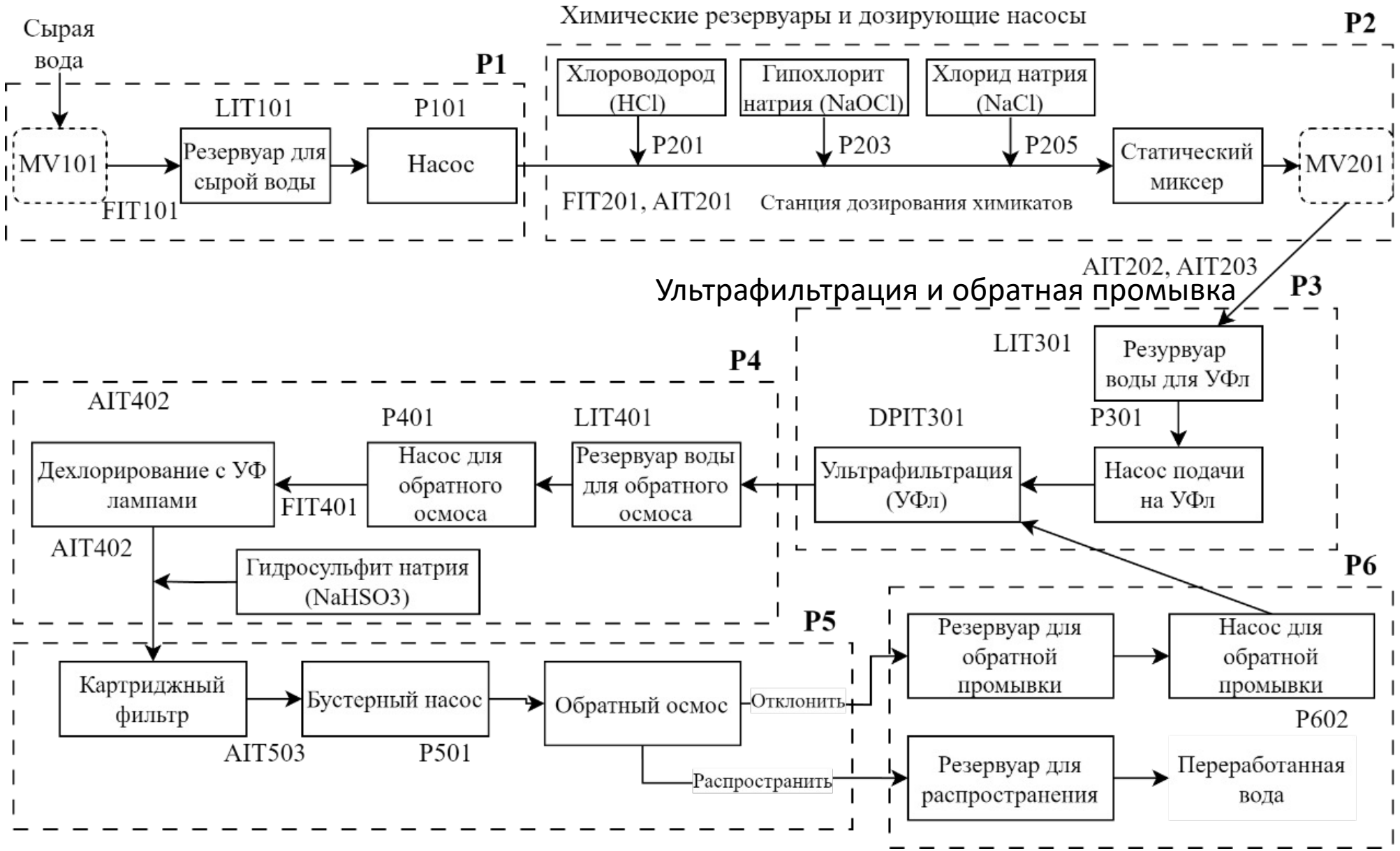
Для эксперимента был выбран набор данных **Safe Water Treatment (SWaT)**. Набор данных описывает работу испытательного стенда SWaT, который имитирует реальную промышленную установку по очистке воды. В этом исследовании используются данные из подпроцесса 3 (P3), поскольку он является целью наиболее значимых атак.



Legend

- P1: Raw water supply & storage
- P2: Chemical dosing
- P3: UF
- P4: Dechlorination
- P5: RO
- P6: RO permeate transfer, UF backwash
- AITx0y: Analyser Indicator Transmitter
- DPITTx0y: Differential Pressure Indicator Transmitter
- FITx0y: Flow Indicator Transmitter
- LITx0y: Level Indicator Transmitter
- MVx0y: Motorised Valve
- Px0y: Pump
- x = component # ; y = process module#

Набор данных Safe Water Treatment (SWaT) [2/4]



Набор данных Safe Water Treatment (SWaT) [3/4]

Фрагмент описания атак на компоненты испытательного стенда SWaT

№ атаки	Описание	Цель	Процессы	Сек.	Тип
1	Открыть MV101	Переполнение резервуара	P1	939	SSSP
2	Включить P102	Повреждение трубы	P1	442	SSSP
3	Увеличить показания LIT101на 1 мм каждую секунду	Недолив резервуара. Повреждение P101	P1	382	SSSP
4	Открыть MV504	Остановить отключение обратного осмоса	P5	389	SSSP
6	Установить показания AIT202 ниже номинального значения	Изменение качества воды	P2	195	SSSP
7	Установить показания LIT301 выше максимального предела	Остановка притока воды. Слив резервуара. Повреждение P301	P3	428	SSSP
8	Установить показания DPIT301 выше номинального значения	Перезапуск процесса обратного осмоса	P3	963	SSSP
10	Установить показания FIT401 ниже номинального значения	Остановка ультрафильтрации	P4	160	SSSP
11	Обнулить показания FIT401	Остановка ультрафильтрации	P4	560	SSSP
13	Закрыть MV304	Остановка процесса P3.	P3	232	SSSP
14	Не дать открыть MV303	Остановка процесса P3	P3	430	SSSP
16	Уменьшить показания LIT301 на 1 мм каждую секунду	Переполнение резервуара	P3	275	SSSP
17	Не дать открыть MV303	Остановка процесса P3	P3	716	SSSP

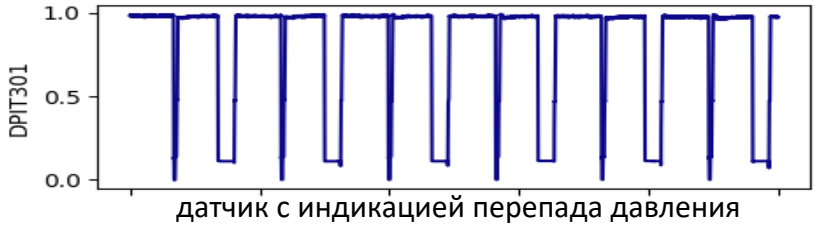
Одношаговые атаки на один компонент (Single Stage Single Point, SSSP), одношаговые атаки на несколько компонентов (Single Stage Multi Point, SSMP), многошаговые атаки на один компонент (Multi Stage Single Point, MSSP), многошаговые атаки на несколько компонентов (Multi Stage Multi Point, MSMP)

Набор данных Safe Water Treatment (SWaT) [4/4]

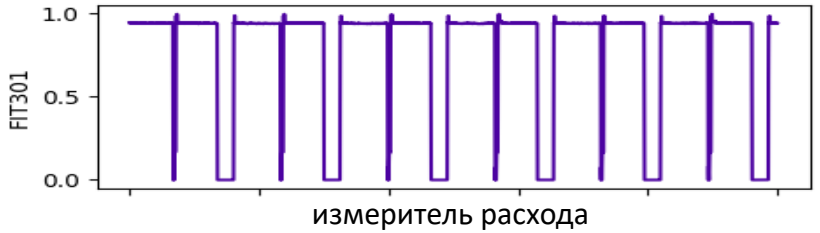
- Моделирование промышленной киберфизической системы
- Год: 2015
- Подпроцесс: P3 (ультрафильтрация и обратная промывка)

- Количество признаков: 9
- Количество обычных событий: 470 000
- Количество событий с атаками: 400 000
- Количество сценариев атак: 36

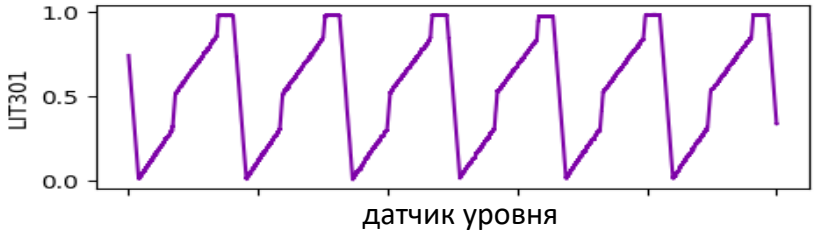
Пример нормальных данных процесса



датчик с индикацией перепада давления



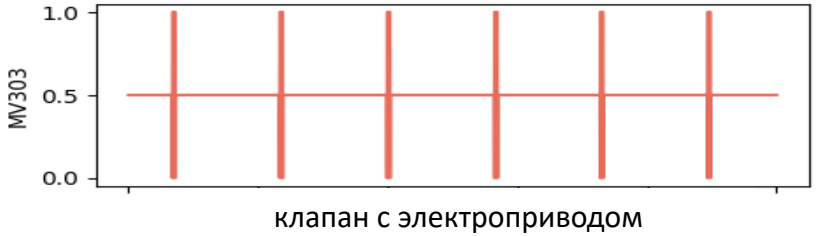
измеритель расхода



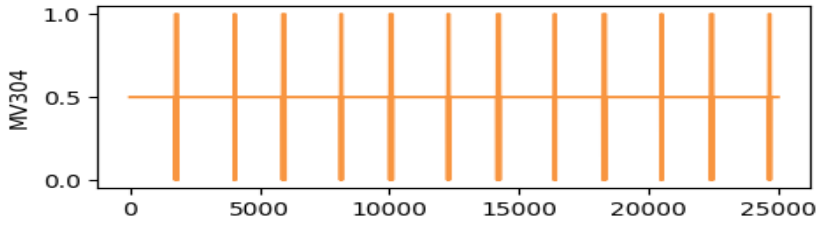
датчик уровня

клапан с электроприводом

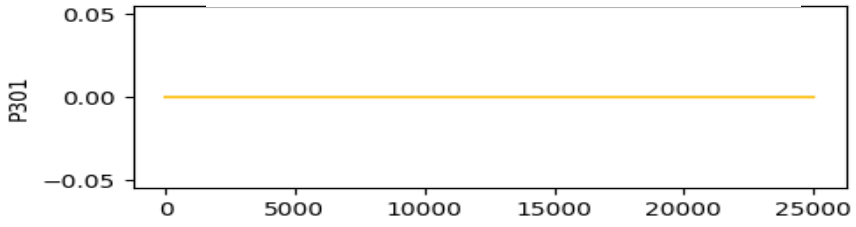
клапан с электроприводом



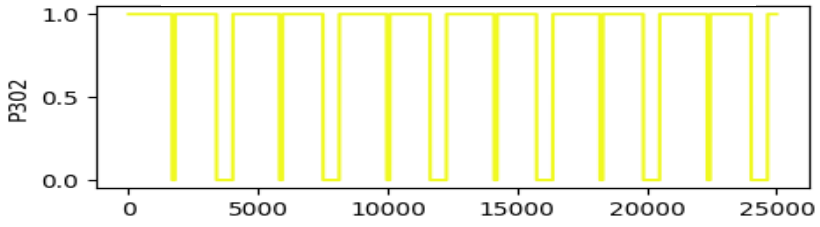
клапан с электроприводом



клапан с электроприводом



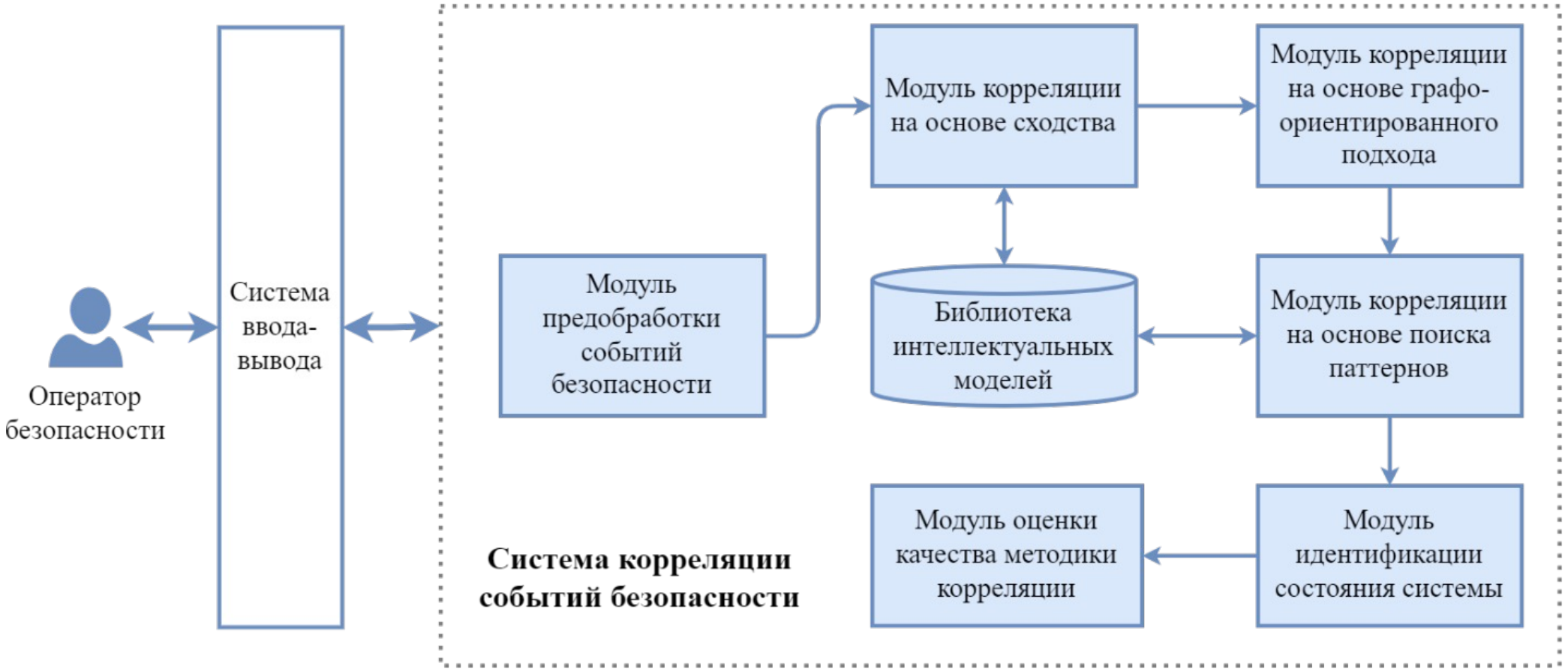
Насос ультрафильтрации



Насос ультрафильтрации

Реализация прототипа

- Прототип программного обеспечения на Python версии 3.8 и библиотеках TensorFlow (версия 2.10), Keras (версия 2.10), Scikit-learn (версия 1.3), SciPy (версия 1.9), NumPy (версия 1.23.5) и других
- Работает с Intel(R) Core(TM) i5, CPU 1.60GHz, 8GB RAM, Windows 10.



Содержание

- Введение
- Предлагаемый подход
- Кластеризация событий безопасности
- Генерация графовых моделей
- Обучение сверточной LSTM на кадрах RSE-графа
- Обнаружение аномалий
- Набор данных и прототип
- Экспериментальная оценка
- Заключение

Основные этапы предлагаемого подхода

I. Кластеризация событий безопасности (выявление репрезентативных событий безопасности RSE):

1. Предварительная обработка данных
2. Реализация алгоритма BIRCH
3. Отображение RSE

II. Генерация графовых моделей (определение переходов между репрезентативными событиями для каждого временного окна):

4. Создание скользящих окон
5. Частотный анализ переходов
6. Генерация графа

III. Подготовка модели прогнозирования графа (обучение сверточной LSTM на кадрах RSE-графа):

7. Создание кадров графа
8. Обучение ConvLSTM
9. Прогнозирование кадров целевого графа
10. Извлечение целевых графов

IV. Обнаружение аномалий (использование ошибок реконструкции модели и векторных расстояний между событиями и центрами кластеров для поиска отклонений):

11. Вычисление ошибки реконструкции графа
12. Расчет расстояний между RSE и исходным событием
13. Расчет среднего расстояния для временного окна
14. Сравнение с пороговыми значениями

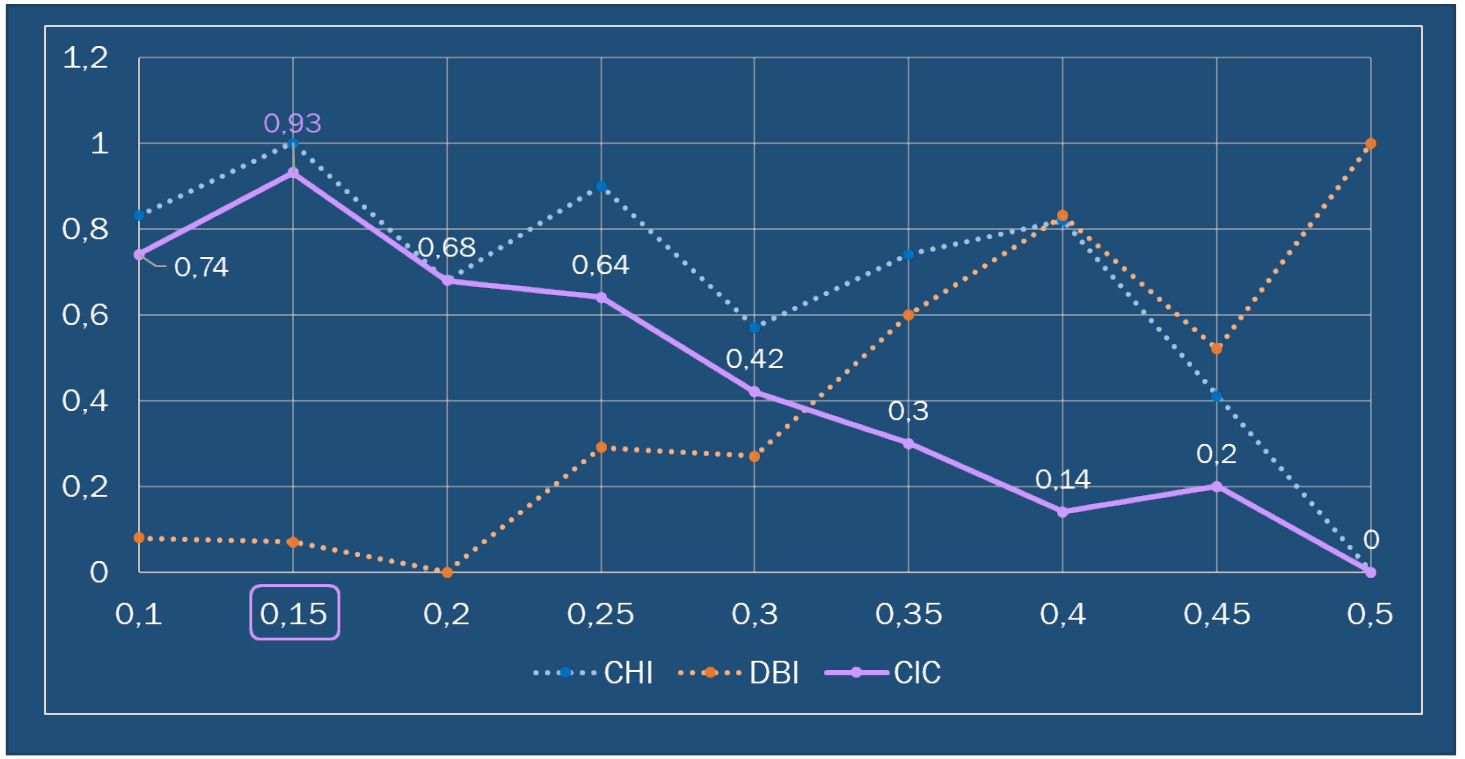
Эксперименты. Шаги 1-2

1. Предобработка данных

Форма матрицы обучения: [423 000 x 9] – нормальные данные
Форма матрицы валидации: [47 000 x 9] – нормальные данные
Форма матрицы тестирования: [400 000 x 9] – данные с атаками

2. Реализация алгоритма BIRCH

Результат: 60 кластеров событий.



Оптимизация порога

- После предварительной обработки данных была получена обучающая матрица, матрицы валидации и матрица тестирования (training matrix, validation matrix, test matrix).
- Вычислен порог кластеризации λ на основе оценки интервала [0,5 . . . 0,1] с шагом 0,05.
- Проведена кластеризация с порогом 0,15 и получено 60 кластеров событий.

Эксперименты. Шаги 3-5

3. Отображение RSE

4. Создание скользящих окон

Sliding window size *h* = 600 (10 min)

Форма матрицы обучения: [422 400 x 600 x 9] – нормальные данные
Форма матрицы валидации: [46 400 x 600 x 9] – нормальные данные
Форма матрицы тестирования: [399 400 x 600 x 9] – данные с атаками

5. Частотный анализ переходов

Форма матрицы обучения: [422 400 x 60 x 60] – нормальные данные
Форма матрицы валидации: [46 400 x 60 x 60] – нормальные данные
Форма матрицы тестирования: [399 400 x 60 x 60] – данные с атаками

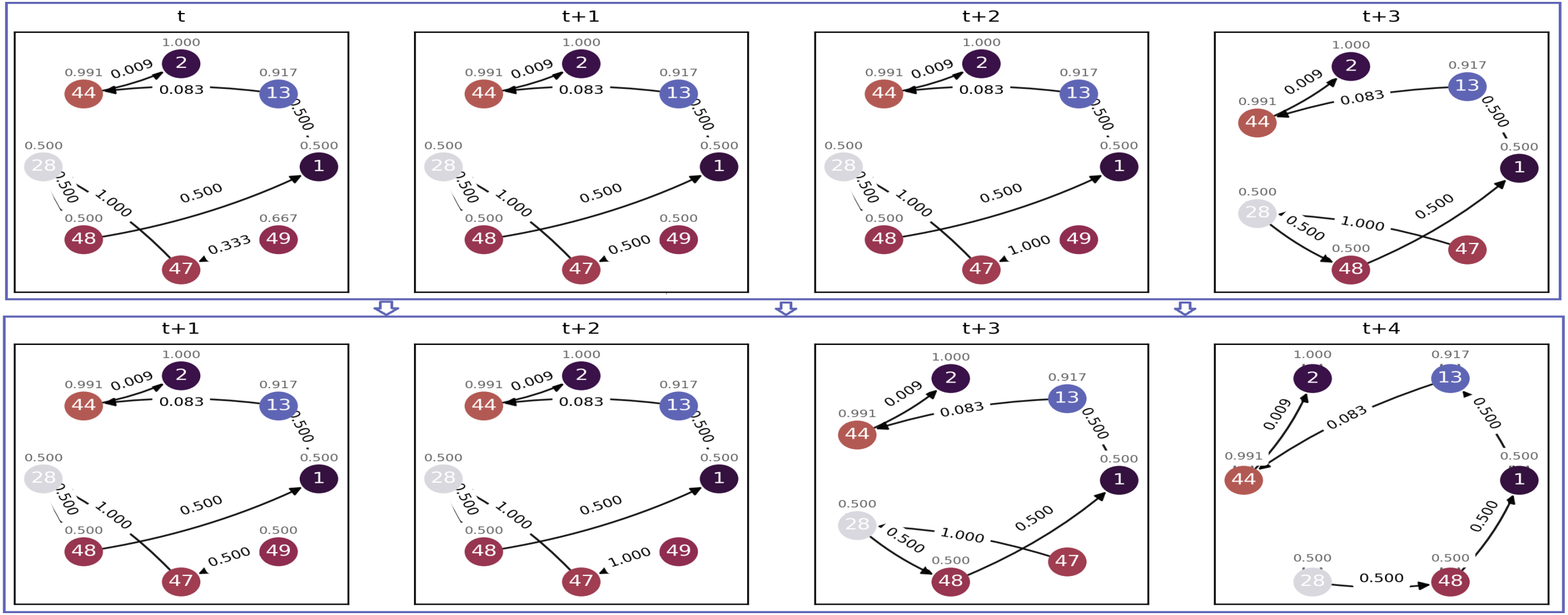
- Алгоритм BIRCH обучается на обычной выборке. Кластеризация проверочных и тестовых выборок выполняется на обученной модели.
- Размер скользящего окна h устанавливается равным 600: для последовательности длительностью 10 минут.
- Для каждого временного окна строится матрица вероятностей перехода. Размер каждой такой матрицы составляет 60 × 60 (количество кластеров).

Эксперименты. Шаги 6-7

6. Генерация графа

7. Создание кадров графа

Размер кадра $\omega = 4$

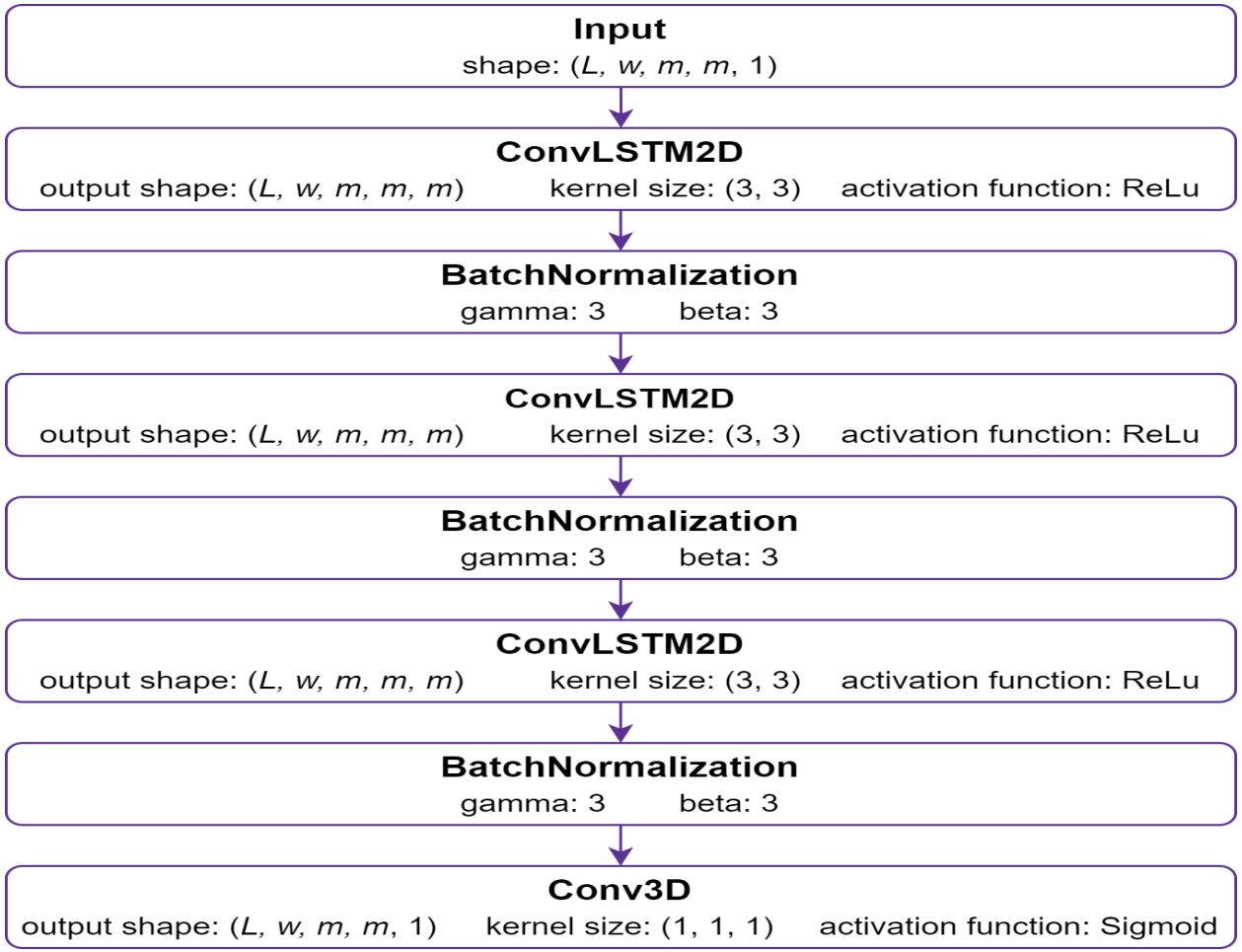


Каждая матрица перехода между репрезентативными событиями преобразуется в граф. Размер кадра графа устанавливается равным 4.

Эксперименты. Шаги 8-11

8. Обучение ConvLSTM

Архитектура модели ConvLSTM



The model is trained using the Adam optimizer with a learning rate of 0.001 in 50.

Функция потерь (Loss function): среднеквадратическая ошибка (mean squared error, MSE)

$$MSE = \frac{\sum_{i=1}^{Z}(q_i - \tilde{q}_i)^2}{Z}$$

Набор обучения: $MSE = 0.221 * 10^{-5}$

Набор валидации: $MSE = 0.328 * 10^{-5}$

9. Прогнозирование кадров целевого графа

10. Извлечение целевых графов

11. Вычисление ошибки реконструкции графа

Выбор порогового значения ошибки реконструкции, равного 90% от MSE валидации (чтобы смягчить влияние выбросов в нормальных данных):

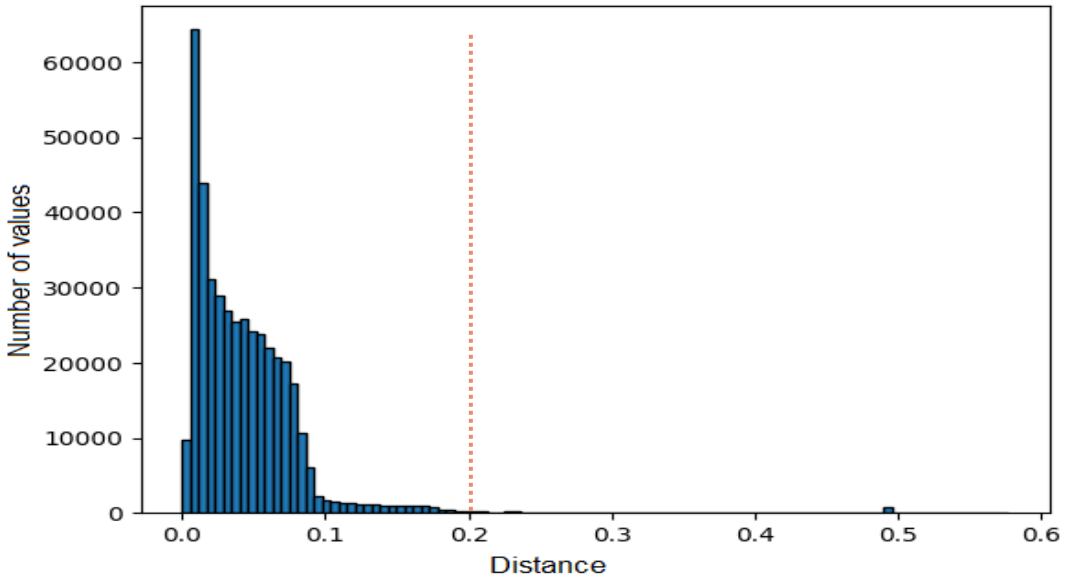
$$\epsilon = 0.9 * 0.328 * 10^{-5} \approx 0.3 * 10^{-5}$$

Эксперименты. Шаги 12-13

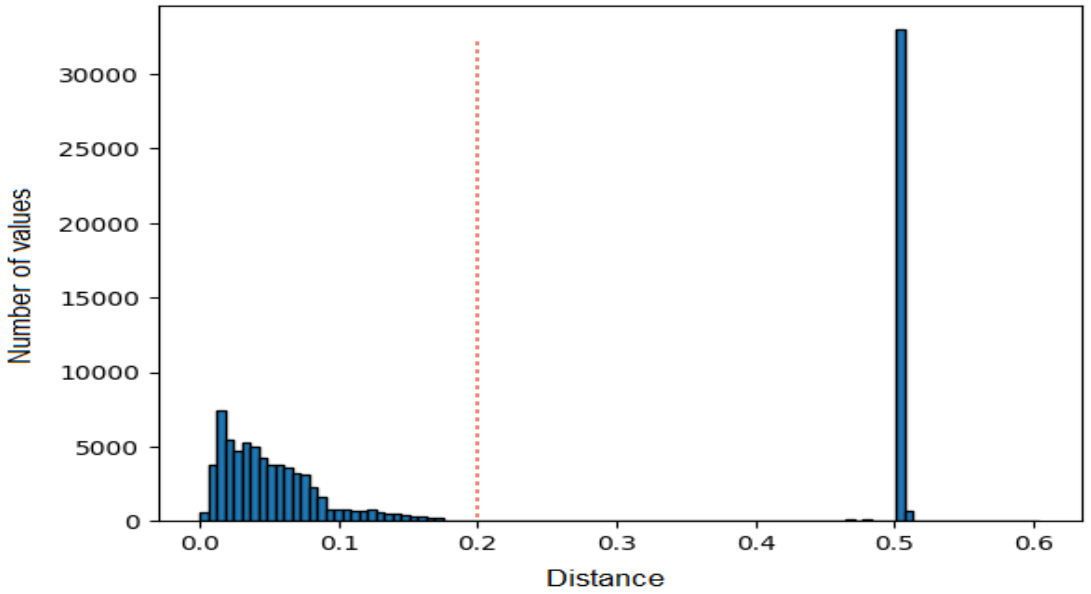
12. Расчет расстояний между RSE и исходным событием

13. Расчет среднего расстояния для временного окна

Распределение средних расстояний на обучающем наборе (нормальные данные)



Распределение средних расстояний на тестовом наборе (данные об атаках)



Выбрано пороговое расстояние $\delta = 0.2$

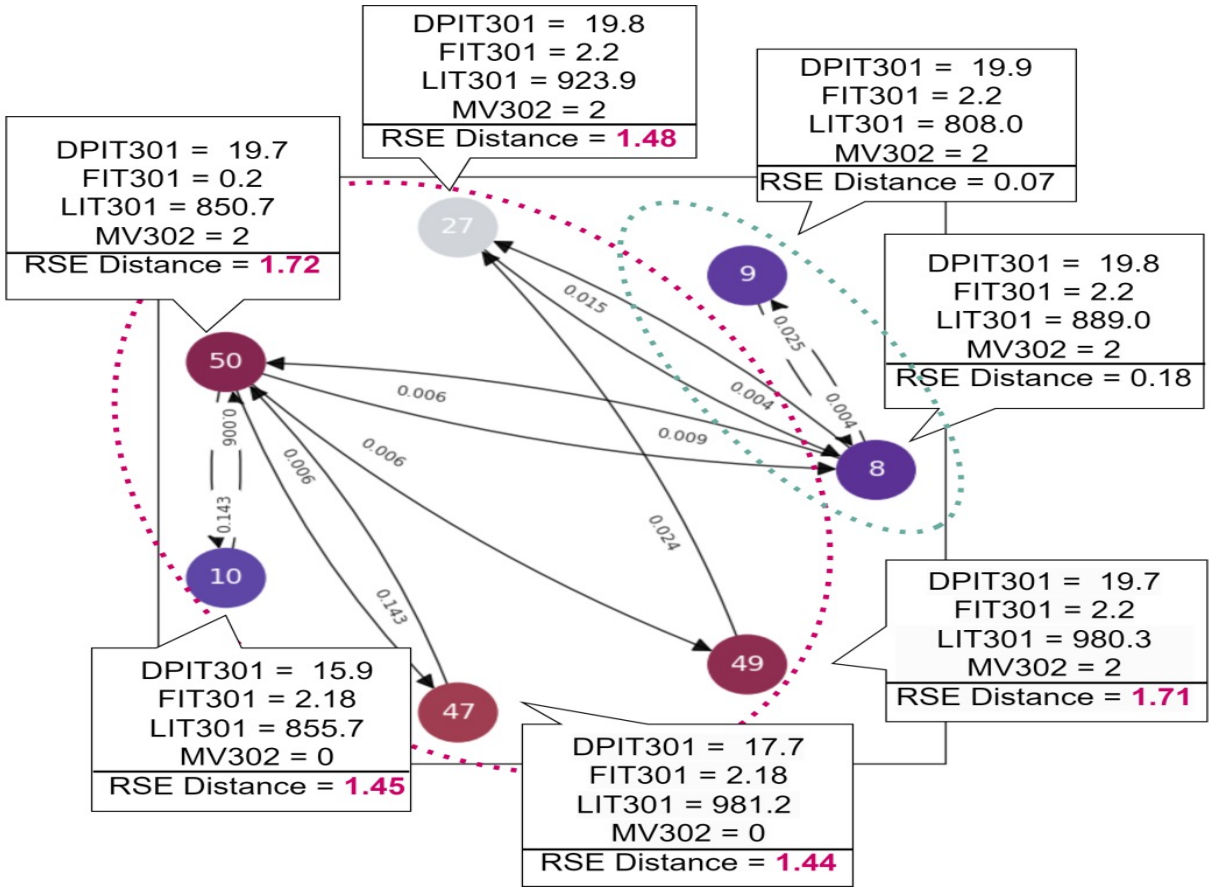
Мы также рассчитали расстояние между событиями в исходном наборе данных и центрами кластеров, к которым принадлежат эти события. Мы установили пороговое значение расстояния, используя распределение расстояний для нормальных данных. Можно отметить, что распределение расстояний в данных атаки имеет явный выброс, что может указывать на аномалию.

Эксперименты. Шаги 14

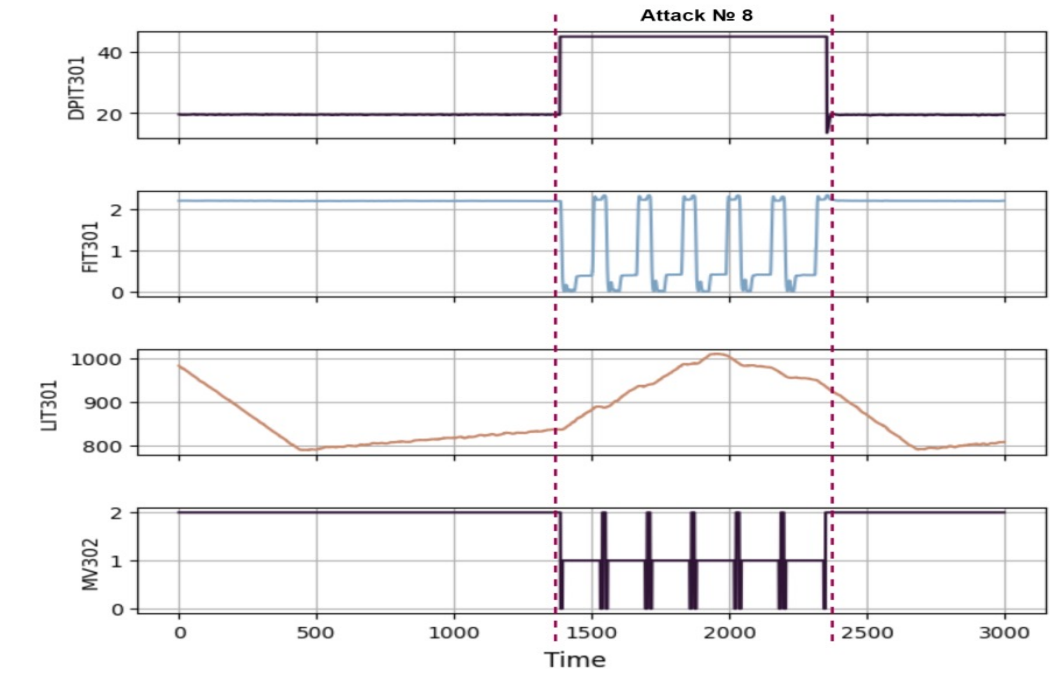
14. Сравнение с пороговыми значениями

Выделены значения характеристик, поскольку они лучше всего отражают характер атаки.

RSE-график для атаки №8



Временной ряд характеристик во время атаки



норма

атака

Атака №8 направлена на датчик DPIT301 и заключается в повышении его значений выше допустимого уровня. После этого процесс обратной промывки запускается снова и снова, о чем свидетельствуют циклические изменения значений. Также наблюдается повышение уровня воды в резервуаре 301 (LIT301).

Результаты экспериментов

Precision $P = \frac{TP}{TP + FP}$

Recall $R = \frac{TP}{TP + FN}$

F-measure $F1 = 2\frac{P \times R}{P + R}$

- *TP* – количество правильно обнаруженных окон атак (true positive);
- *FP* – количество неверно обнаруженных окон атак (false positive);
- *TN* – количество правильно обнаруженных окон без атак (true negative);
- *FN* – количество неверно обнаруженных окон без атак (false negative).

Metric	*Our approach*	*Nedeljkovic & Jakovljevic*[1]	*Kravchik & Shabtai*[2]			*Xie X. et al.*[3]
	ConvLSTM	*CNN*	*LSTM*	*CNN*	*CNN-LSTM*	*CNN*
Precision	0.845	0.900	n/a	n/a	n/a	n/a
Recall	0.894	0.833	n/a	n/a	n/a	n/a
F-measure	**0.869**	0,865	0.666	0.8	0.857	0.854

[1] Nedeljkovic, D., & Jakovljevic, Z. (2022). CNN based method for the development of cyber-attacks detection algorithms in industrial control systems. *Computers & Security, 114*, 102585.

[2] Kravchik, M., & Shabtai, A. (2018). Detecting cyber attacks in industrial control systems using convolutional neural networks. In Proc. of the 2018 workshop on cyber-physical systems security and privacy (pp. 72-83).

[3] Xie, X., Wang, B., Wan, T., & Tang, W. (2020). Multivariate abnormal detection for industrial control systems using 1D CNN and GRU. *IEEE Access, 8,* 88348-88359.

Заключение (1)

Преимущества подхода

- Не требует дополнительных знаний о режимах работы сенсоров в сети IoT
- Использование неконтролируемого обучения (алгоритм BIRCH) для извлечения репрезентативных событий, что решает задачу изучения нормального поведения
- Обладает объяснимостью обнаружения аномалий, поскольку можно точно проследить, какая последовательность событий привела к угрозе
- Подходит для обнаружения многошаговых атак. Позволяет обнаруживать аномальные переходы между репрезентативными событиями безопасности, что учитывает временную последовательность событий. Более информативно, чем обнаружение изолированных аномальных событий в процессах
- Обнаружение аномалий показывает результаты с высокой эффективностью (>84%)

Ограничения подхода

- Необходимо иметь репрезентативные данные для составления RSE-графов
- Зависит от выбранных порогов кластеризации/ошибки реконструкции/расстояния
- Требуется большой объем памяти для обработки и хранения матриц и графов

Заключение (2)

Направления будущих исследований

- Оценить подход при создании графов более высоких измерений
- Оценить подход на других наборах данных IoT и глубоких архитектурах
- Улучшение эффективности обнаружения аномалий
- Оптимизация подхода для снижения вычислительных затрат
- Внедрение подхода в комбинированную систему корреляции событий безопасности

Контакты

Федеральное государственное бюджетное учреждение науки «Санкт-Петербургский Федеральный исследовательский центр Российской академии наук» (СПб ФИЦ РАН), Санкт-Петербургский институт информатики и автоматизации Российской академии наук, Лаборатория проблем компьютерной безопасности,

Адрес: 39, 14 линия В.О., Санкт-Петербург, 199178

Телефон: +7(812)328-71-81

URL: http://comsec.spb.ru

Контакты:

ГНС, д.т.н., проф. Котенко Игорь Витальевич, ivkote@comsec.spb.ru, http://comsec.spb.ru/kotenko

Благодарности

- работа выполнена совместно с Левшун Дианой Альбертовной, СПб ФИЦ РАН
- при частичной финансовой поддержке бюджетной темы FFZF-2025-0016.

СПб ФИЦ РАН