DOI https://doi.org/10.32689/maup.it.2026.2.12
УДК 004.421:004.65:004.8
JEL Classification: C38, C45, C63

ОБРОБКА ТЕЛЕМЕТРИЧНИХ ДАНИХ ТА ВИЗНАЧЕННЯ РЕЛЕВАНТНИХ ОЗНАК ДЛЯ РОБОТИ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ У КОНТЕКСТІ ОЦІНКИ ФУНКЦІОНАЛЬНОГО СТАНУ РЕЛЯЦІЙНИХ БАЗ ДАНИХ

О. С. Чикалов, О. Б. Берест

TELEMETRY DATA PROCESSING AND IDENTIFICATION OF RELEVANT FEATURES FOR A DECISION SUPPORT SYSTEM IN THE CONTEXT OF ASSESSING THE FUNCTIONAL STATE OF RELATIONAL DATABASES

Oleksandr Chykalov, Oleh Berest

Анотація

Об'єктом дослідження є процес оброблення телеметричних даних Oracle Database 21c та формування релевантного простору ознак для системи підтримки прийняття рішень щодо оцінювання функціонального стану реляційної бази даних. Проблема полягає у тому, що великий початковий набір метрик містить одночасно діагностично значущі, контекстні, дублюючі та вузькоспеціалізовані показники, а їх безпосереднє використання у моделі ускладнює інтерпретацію та може погіршувати стійкість оцінювання. Запропоновано методологію, яка поєднує потокові подання V$ та історичні таблиці Automatic Workload Repository, побудову шляхів з'єднання і реляційних дерев, синхронізацію за UTC, агрегування, нормалізацію та формування часових характеристик. Відбір ознак виконується за доменно обґрунтованими критеріями діагностичної значущості, недубльованості, інтерпретованості та придатності до часового узгодження. У результаті десять груп кандидатних метрик розподілено на шість базових, дві контекстні та дві надлишкові: до базового набору віднесено CPU, Database Wait Time Ratio, Buffer Cache Hit Ratio, Active Sessions, wait events / wait classes та I/O waits; Executes і Physical Reads враховано частково, а DB Block Gets і Background CPU Time вилучено. Мітки інцидентів формуються поєднанням детерміністичних правил і нечітких функцій належності. Сформований простір ознак використано для побудови часово впорядкованої матриці та класифікації функціонального стану бази даних за допомогою нейромережевої моделі екстремального навчання з інформаційно-керованою селекцією прихованих нейронів. Експеримент на тижневому наборі приблизно з 10080 одновилинних записів дав точність близько 0,64 у вибраній робочій точці; аналіз ROC-кривої та калібраційної діаграми засвідчив потребу в додатковому налаштуванні порогу класифікації. Практичне значення полягає у відтворюваному переході від неоднорідної телеметрії до компактного, пояснюваного та придатного для СППР простору ознак.

Ключові слова: *телеметрія, відбір ознак, шляхи з'єднання, екстремальне навчання, інцидент, матриця ознак, СППР.*

Abstract

The object of the study is the processing of Oracle Database 21c telemetry data and the formation of a relevant feature space for a decision support system intended to assess the functional state of a relational database. The problem is that a large initial set of metrics simultaneously contains diagnostically significant, contextual, redundant, and narrowly specialized indicators, while their direct use in the model complicates interpretation and may reduce assessment stability. A methodology is proposed that combines streaming V$ views and historical Automatic Workload Repository tables, the construction of join paths and relational trees, UTC-based temporal synchronization, aggregation, normalization, and temporal feature generation. Feature selection is performed using domain-informed criteria of diagnostic significance, non-redundancy, interpretability, and suitability for temporal alignment. As a result, ten groups of candidate metrics were divided into six core, two contextual, and two redundant groups: CPU, Database Wait Time Ratio, Buffer Cache Hit Ratio, Active Sessions, wait events / wait classes, and I/O waits were included in the core set; Executes and Physical Reads were partially included, whereas DB Block Gets and Background CPU Time were excluded. Incident labels are generated by combining deterministic rules with fuzzy membership functions. The resulting feature space was used to construct a time-ordered matrix and classify the functional state of the database using an extreme learning neural network model with information-guided hidden-neuron selection. An experiment conducted on a one-week dataset containing approximately 10,080 one-minute observations yielded an accuracy of approximately 0.64 at the selected operating point; analysis of the ROC curve and calibration plot indicated the need for additional tuning of the classification threshold. The practical significance of the study lies in providing a reproducible transition from heterogeneous telemetry to a compact, explainable feature space suitable for use in a decision support system.

Key words: *telemetry, feature selection, join paths, extreme learning, incident, feature matrix, decision support system.*

1. Вступ. Реляційні бази даних (РБД) є базовими компонентами корпоративних інформаційних систем, а їхній функціональний стан безпосередньо визначає доступність сервісів, швидкість виконання транзакцій і стабільність залежних програмних компонентів. За умов змінного навантаження традиційне спостереження за окремими показниками вже не забезпечує своєчасного виявлення короткочасних відхилень, прихованих перевантажень і взаємопов'язаних ресурсних обмежень. Тому

© Чикалов О. С., Берест О. Б., 2026
Стаття поширюється на умовах ліцензії відкритого доступу CC BY 4.0

сучасне оцінювання функціонального стану РБД потребує інтеграції потокової та історичної телеметрії, формування узгодженого простору ознак і застосування інтелектуальних засобів підтримки прийняття рішень.

Практична цінність таких досліджень полягає у можливості перейти від реактивного адміністрування до раннього виявлення інцидентних станів, обґрунтованого вибору діагностичних метрик і формування рекомендацій щодо оптимізації ресурсів. Для цього телеметричні дані мають бути синхронізовані у часі, очищені, перетворені у фіксовані вектори та забезпечені інтерпретованими мітками станів.

Сучасні розподілені реляційні системи, зокрема Spanner, OceanBase і TiDB, демонструють розвиток високодоступної інфраструктури даних [1-3]. Паралельно розвиваються методи виявлення аномалій за системними журналами. Репозиторій Loghub [4] став основою для моделей RobustLog, DeepLog, LogAnomaly, компонентно-орієнтованого аналізу, LogBERT та інших підходів [5-13]. Такі моделі дають змогу виявляти нетипові послідовності подій, однак переважно працюють із текстовими журналами, а не з доменно-специфічною телеметрією СУБД.

Критичний аналіз джерел показує три локальні невирішені питання. По-перше, наявні набори даних часто містять обмежений перелік інцидентів, логи окремих вузлів або синтетично сформовані послідовності [14-16]. По-друге, навіть рішення для часових і розподілених баз даних [17, 18] не формують єдиного механізму поєднання потокових метрик Oracle з історичними зрізами Automatic Workload Repository (AWR). По-третє, статичні набори ознак і фіксовані правила недостатньо враховують зміну робочого навантаження та не завжди забезпечують інтерпретованість отриманих рішень.

Подання продуктивності Oracle і механізми AWR надають широкий набір показників для аналізу роботи екземпляра [19, 20], однак сам факт доступності метрик не вирішує питання їхнього часового узгодження, відбору, документування та перетворення у навчальну матрицю. Автоматизація формування ознак у реляційних структурах [21] і принципи FAIR та словників даних [22, 23] створюють методичну основу для такого перетворення, але потребують адаптації до задачі оцінювання функціонального стану РБД.

Отже, загальна невирішена проблема полягає не лише в інтеграції потокових та історичних джерел Oracle, а й у методично обґрунтованому переході від широкого набору доступних метрик до структурованого простору ознак. Такий простір має відокремлювати базові діагностичні показники від контекстних характеристик, що впливають лише у поєднанні з іншими метриками, а також від дублюючих або вузькоспеціалізованих показників. Одночасно необхідно забезпечити часове узгодження, пояснюване призначення міток та підготовку даних для алгоритмів СППР. Тому дослідження, присвячені обробленню телеметричних даних і відбору релевантних ознак для оцінювання функціонального стану реляційних баз даних, є актуальними.

Метою дослідження є розроблення методології оброблення телеметричних даних і формування релевантного простору ознак для оцінювання функціонального стану Oracle Database 21c. Наукова частина мети передбачає шляхово-реляційну інтеграцію джерел, доменно обґрунтоване оцінювання кандидатних метрик з їх поділом на базові, контекстні та надлишкові, а також комбіноване призначення міток інцидентів. Практична частина полягає у побудові часово узгоджених навчальних і тестових матриць та перевірці придатності сформованого простору ознак у прототипі СППР із застосуванням нейромережевої моделі - машини екстремального навчання (Extreme Learning Machine, ELM) – та інформаційно-екстремального алгоритму навчання (Information-Extreme Learning Algorithm, IELA).

Для досягнення мети було визначено наступні завдання:

1. розробити спосіб інтеграції та часового узгодження потокових подань V$ і історичних таблиць AWR;
2. розробити процедуру формування й доменного оцінювання простору ознак та розподілити кандидатні метрики на базові, контекстні й надлишкові;
3. розробити комбінований механізм призначення міток інцидентів на основі детерміністичних правил і нечітких функцій належності;
4. побудувати навчальні й тестові матриці на основі відібраного простору ознак та перевірити його придатність для класифікації за допомогою ELM/IELA;
5. оцінити якість класифікації, визначити межі достовірності результатів, умови практичного використання та напрями розвитку методології.

2. Матеріали і методи.

2.1. Об'єкт, гіпотеза, припущення та спрощення дослідження. Об'єктом дослідження є процес оброблення телеметричних даних і формування релевантних ознак для оцінювання функціонального стану реляційної бази даних Oracle Database 21c. Досліджуване середовище містить один основний вузол та один екземпляр БД, що працює за звичайного змінного навантаження.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Основна гіпотеза дослідження полягає в тому, що інтеграція потокових метрик реального часу та історичних даних AWR за умови їх часового узгодження і шляхово-реляційного вилучення дає змогу сформувати цілісне представлення функціонального стану реляційної бази даних. Оцінювання інформативності телеметричних показників, відбір релевантних ознак і комбіноване призначення міток на основі детерміністичних та нечітких правил забезпечують формування інформативного й інтерпретованого простору ознак, придатного для розмежування штатних та інцидентних станів бази даних. Застосування моделі IELA з інформаційно-керованою селекцією прихованих нейронів дає змогу перевірити придатність сформованого простору ознак для класифікації функціонального стану реляційної бази даних.

Прийнято такі припущення: часові позначки всіх джерел можуть бути приведені до UTC; потокові показники збираються з інтервалом 60 с; історичні агрегати AWR формуються переважно з інтервалом близько однієї години; доступні подання мають узгоджені ідентифікатори екземпляра, знімка та метрики. Для експерименту використано тижневий набір даних обсягом близько 10080 записів.

До спрощень належать бінарне подання стану «штатний/інцидентний», фіксоване ковзне вікно W = 10 хв, розгляд насамперед інцидентів паралельності та I/O, застосування одного робочого порогу та відсутність перевірки на багатовузловій або RAC-конфігурації. Ці спрощення визначають межі узагальнення результатів.

Для забезпечення цілісності процесу оцінювання функціонального стану реляційної бази даних запропоновано методологію оброблення телеметричних даних, що охоплює етапи від інтеграції джерел даних до оцінювання результатів класифікації. Загальну послідовність етапів методології наведено на рис. 1.



Рис. 1. Послідовність оброблення телеметричних даних

Як показано на рис. 1, запропонована методологія включає послідовні етапи: інтеграцію потокових та історичних даних, формування і перетворення ознак, документування набору даних, призначення міток інцидентів, побудову навчальної вибірки та подальше оцінювання результатів класифікації. Кожний етап формує вхідні дані для наступного, що забезпечує узгоджений перехід від первинної телеметрії до формування інформативного простору ознак і перевірки його придатності для розмежування штатних та інцидентних станів реляційної бази даних.

2.2. Джерела телеметричних даних. Джерела телеметрії поділено на дві групи: показники реального часу з динамічних подань V$ та історичні агрегати Automatic Workload Repository. Потокові подання використовуються для короткострокового аналізу, тоді як DBA_HIST_* забезпечують ретроспективний контекст, узгодження знімків та відновлення довгострокової динаміки. Основні джерела [19, 20] наведено в таблиці 1.

Таблиця 1

Джерела телеметричних даних Oracle, використані у дослідженні

№	Подання	Тип та інтервал	Призначення	Ключові метрики / параметри
1	V$SYSMETRIC	Реальний час, ≈1 хв	Поточні показники продуктивності	Throughput, Response Time, Logical/Physical Reads, Redo Generated
2	V$SYSSTAT	Реальний час, накопичувальний	Кумулятивні лічильники	DB Block Gets, Consistent Gets, Physical Reads, Executes
3	V$SYSTEM_EVENT	Реальний час, ≈1 хв	Статистика подій очікування	Waits, Time Waited, Average Wait, Event Name
4	V$WAITCLASSMETRIC	Реальний час, ≈1 хв	Затримки за класами	User I/O, System I/O, Commit, Concurrency, Application
5	V$SYS_TIME_MODEL	Реальний час, ≈1 хв	Часові витрати процесів	DB Time, DB CPU, Background CPU Time, Parse Time
6	DBA_HIST_SYSMETRIC_SUMMARY	AWR, ≈1 год	Історичні агрегати системних метрик	Average, Min, Max
7	DBA_HIST_SYSSTAT	AWR, ≈1 год	Історія системних статистик	Кумулятивні лічильники системних подій
8	DBA_HIST_SYSTEM_EVENT	AWR, ≈1 год	Історія подій очікування	Event Name, Total Waits, Time Waited
9	DBA_HIST_ACTIVE_SESS_HISTORY	AWR, ≈1 год	Історія активних сесій	Session State, SQL ID, Wait Class, CPU Usage
10	DBA_HIST_SNAPSHOT	AWR, ≈1 год	Синхронізація знімків	SNAP_ID, BEGIN_INTERVAL_TIME, END_INTERVAL_TIME

2.3. Вилучення, узгодження та первинне формування ознак. Вилучення даних виконується за шляхами з'єднання між таблицею-джерелом основних метрик і пов'язаними поданнями продуктивності. Формально шлях подано у вигляді:

$$p = \left(T_0 - -c_1 \to T_1 - -c_2 \to \ldots - -c_k \to T_k : c\right), \tag{1}$$

де T_0 є початковим джерелом метрик, T_i – пов'язаними поданнями продуктивності, c_i – стовпчиками з'єднання, зокрема METRIC_ID, SNAP_ID та INSTANCE_NUMBER, c – цільовим стовпчиком у кінцевій таблиці T_k. Для історичних даних застосовується, наприклад, шлях DBA_HIST_SYSMETRIC_SUMMARY(SNAP_ID) → DBA_HIST_SNAPSHOT, а для потокових даних – узгодження V$SYSMETRIC і V$SYS_TIME_MODEL за METRIC_NAME/METRIC_ID та часовими інтервалами BEGIN_TIME, END_TIME.

Після вилучення значення узгоджуються за UTC. Одинарні шляхи перетворюються на скалярні ознаки, категоріальні атрибути кодуються, а багатозначні шляхи агрегуються у статистики або вкладення часових рядів. Для кожної групи можуть застосовуватися нормалізація до інтервалу [0; 1], z-score, експоненційне ковзне середнє та ковзне стандартне відхилення. На цьому етапі формується первинний набір кандидатних ознак; остаточне рішення щодо їх включення до моделі приймається за процедурою, наведеною у підрозділі 2.4.

Приклад SQL-шаблону для вилучення агрегованих історичних показників:

```
String sql =
    "SELECT begin_time, end_time, metric_name, average, metric_unit " +
    "FROM dba_hist_sysmetric_summary " +
    "WHERE begin_time >= ? AND end_time < ? " +
    "AND metric_name IN (?, ?, ?)";

try (Connection c = ds.getConnection();
     PreparedStatement ps = c.prepareStatement(sql)) {
    ps.setTimestamp(1, fromTs); // UTC
    ps.setTimestamp(2, toTs);
    ps.setString(3, "Database Wait Time Ratio");
    ps.setString(4, "Host CPU Utilization (%)");
    ps.setString(5, "Buffer Cache Hit Ratio");
    try (ResultSet rs = ps.executeQuery()) {
        while (rs.next()) {
            // Перетворення запису на рядок матриці ознак
        }
    }
}
```


ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Словник ознак містить назву, джерело, одиницю вимірювання, спосіб агрегації, часовий інтервал і правила інтерпретації. Метадані документуються за принципами FAIR у форматах JSON-LD або Schema.org [22, 23]. Кожний синхронізований снапшот зберігається як окремий JSON-об'єкт:

```
{
  "ts": "2025-08-13T10:20:00Z",
  "host_cpu_pct": 62.1,
  "buf_cache_hit": 96.7,
  "enqueue_waits_s": 0.2
}
```

2.4. Критерії оцінювання та відбору ознак. Первинний простір кандидатних ознак формується з телеметричних показників, доступних через подання, наведені в таблиці 1. Одиницею оцінювання є не окреме поле таблиці, а змістовна група метрик, що характеризує певний механізм функціонування БД: використання процесора, ефективність пам'яті, інтенсивність I/O, активність сесій, події очікування або інтенсивність SQL-операцій.

Рішення щодо кожної групи приймається за чотирма доменними критеріями:

1) прямий зв'язок із проявами деградації продуктивності або інцидентним станом;

2) недубльованість відносно більш узагальнених показників;

3) інтерпретованість для адміністратора БД та можливість пояснити вплив на підсумкову оцінку;

4) придатність до стабільного вилучення, агрегування й часового узгодження у потоковому та історичному режимах.

За результатом оцінювання ознаці надається один із трьох статусів. «Відібрано» означає включення до базового діагностичного простору. «Частково враховано» позначає контекстну ознаку, яка не є достатньою для самостійного висновку, але може уточнювати інтерпретацію базових показників в агрегованому або умовному вигляді. «Не відібрано» означає, що показник є надто вузьким, дублюючим або слабко пов'язаним із цільовим станом у межах поставленої задачі.

Зазначена процедура є доменною якісною селекцією телеметричних ознак і не ототожнюється з інформаційною селекцією прихованих нейронів IELA. Спочатку формується та обґрунтовується вхідний простір ознак, а вже після цього IELA оцінює інформаційний внесок елементів прихованого шару. Такий поділ потрібний для коректної інтерпретації наукового результату.

2.5. Призначення міток інцидентів. Часткові нормалізовані оцінки об'єднуються в інтегральну функцію виявлення F(x):

$$F(x)=\sum_i w_i s_i(x), \sum_i w_i = 1, \tag{2}$$

де $s_i(x)$ – нормалізована оцінка i-ї ознаки, а w_i – її ваговий коефіцієнт; сума ваг дорівнює одиниці. Функція F(x) використовується як інтерпретований критерій функціонального стану РБД.

Бінарна мітка визначається пороговим правилом:

$$y = 1\left[F(x) \geq \tau\right], \tag{3}$$

де τ – робочий поріг. Значення $F(x) \geq \tau$ відповідає інцидентному стану, а $F(x) < \tau$ – штатному. Початкові ваги задаються рівномірно з доменними коригуваннями; зокрема, до ваг CPU та активності ASH додається 0,05.

Детерміністичні правила охоплюють два типи інцидентів. Інцидент паралельності фіксується, коли середня кількість затримок через блокування перевищує 5 і понад 40 % часу БД-сеансу припадає на очікування через конкурентний доступ. Інцидент I/O фіксується, коли понад 60 % часу сеансу припадає на очікування I/O, а інтенсивність логування транзакцій перевищує 95-й перцентиль базового рівня; у такому разі F(x) збільшується на 0,2.

Для врахування плавного переходу між штатним і критичним станами використовується сигмоїдна функція належності:

$$\mu^{l}_{\text{крит}}(x) = \frac{1}{1+\exp\left(-k(x-c)\right)}, \tag{4}$$

де c – критична точка, а k – коефіцієнт крутизни, налаштований відповідно до мінливості метрики. Агрегована нечітка оцінка обчислюється за формулою:

$$\sum_i w_i \mu_{\text{крит},i}(x). \tag{5}$$

Значення агрегованої нечіткої оцінки не менше 0,6 розглядається як ознака аномального стану. Таким чином, мітка формується поєднанням інтерпретованих правил і плавних функцій належності.

2.6. Формування вибірки, модель та критерії оцінювання. Первинні часові ознаки перетворюються у матрицю $X \in R^{Nxd}$, де кожний рядок відповідає стану системи у момент часу, а вектор $y \in \{0; 1\}^N$ містить мітки. Для часової динаміки застосовується ковзне вікно W = 10 хв, що охоплює десять послідовних однохвилинних снапшотів. У фактичному експерименті послідовності розподілено у співвідношенні 70/15/15 для навчання, валідації та тестування без перемішування часових відрізків.

Базовою моделлю є Extreme Learning Machine – одношарова нейронна мережа прямого поширення, у якій параметри вхідного та прихованого шарів задаються випадково і фіксуються, а навчаються лише вихідні ваги. У запропонованому Information-Extreme Learning Algorithm приховані нейрони оцінюються за інформаційним внеском, після чого зберігаються найбільш інформативні кандидати. Інформаційний внесок визначається як:

$$IG = H(Y) - H(Y \mid Z), \tag{6}$$

де H(Y) – ентропія міток, а H(Y|Z) – умовна ентропія міток за наявності виходу нейрона Z. Такий критерій орієнтує відбір на зв'язок активації нейрона з цільовим станом, а не лише на випадкову структуру прихованого шару.

Для оцінювання використовуються часовий графік спрацювань у підтвердженому інцидентному вікні, ROC-крива, калібраційний графік та довірчі інтервали Вілсона. Для непорогових оцінок передбачено 95 % довірчі інтервали блочного бутстрапу з безперервним блоком розміром 5 хвилин. Тестовий набір, фрагмент якого наведено в Таблиці 2, сформовано з чотирьох інформаційних полів: time_utc, що визначає часову позначку UTC телеметричного спостереження; F, що містить значення інтегрального бала функціонального стану; y_true, що відповідає істинній бінарній мітці штатного або інцидентного стану; та y_pred, що містить прогнозовану моделлю бінарну мітку, сформовану з урахуванням обраного робочого порогу τ.

Таблиця 2

Фрагмент тестового набору даних для оцінювання якості моделі

time_utc	F	y_true	y_pred
2025-08-04T00:00:00Z	0.59625811	1	1
2025-08-04T00:01:00Z	0.253569967	0	0
2025-08-04T00:02:00Z	0.112429378	0	0
2025-08-04T00:03:00Z	0.300775039	0	1
2025-08-04T00:04:00Z	0.589850204	1	1
2025-08-04T00:05:00Z	0.700235738	1	1
2025-08-04T00:06:00Z	0.611445935	1	1
2025-08-04T00:07:00Z	0.346930432	0	1
2025-08-04T00:08:00Z	0.185999865	0	0

3. Результати і обговорення.

3.1. Результати інтеграції та часового узгодження телеметрії. Результатом виконання першої задачі стало формування інтеграційного контуру з десяти подань Oracle, що охоплює потокові системні метрики, кумулятивні статистики, події та класи очікування, модель часу, активні сесії й історичні знімки AWR. Поєднання SNAP_ID, INSTANCE_NUMBER, METRIC_ID та часових меж забезпечило приведення різнорідних джерел до спільної UTC-шкали.

На відміну від моделей, що аналізують переважно текстові послідовності журналів [4-16], отриманий результат використовує числову телеметрію на рівні конкретної СУБД. Це дає змогу безпосередньо пов'язувати ознаки з використанням CPU, пам'яті, I/O та внутрішніми очікуваннями. Проблемна частина, визначена у вступі, закривається за рахунок одночасного використання короткострокових V$-показників і ретроспективного контексту AWR.

Практичне застосування результату потребує доступу до відповідних подань, однакової часової зони, контролю одиниць вимірювання та перевірки частоти оновлення. У разі зміни конфігурації Oracle або переходу до RAC шляхові правила мають бути розширені додатковими ідентифікаторами вузлів та екземплярів.

3.2. Результати формування, оцінювання та відбору релевантних ознак. Результатом другої задачі стало не лише складання переліку метрик, а формування структурованого простору ознак із зафіксованим рішенням щодо кожної кандидатної групи. Із десяти оцінених груп шість включено до базового діагностичного набору, дві залишено як контекстні та дві вилучено як вузькі або надлишкові.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

До базового набору увійшли Host CPU Utilization, Database Wait Time Ratio, Buffer Cache Hit Ratio, Active Sessions, wait events / wait classes та I/O waits. На основі результатів проведених експериментів було оцінено релевантність телеметричних ознак для визначення функціонального стану реляційної бази даних. За результатами оцінювання прийнято рішення щодо включення, часткового врахування або виключення кожної ознаки з кінцевого простору ознак. Узагальнені результати наведено в таблиці 3.

Таблиця 3

Результати оцінювання та відбору телеметричних ознак

Метрика	Джерело даних	Опис	Статус відбору	Обґрунтування
Host CPU Utilization (%)	V$SYSMETRIC DBA_HIST_ SYSMETRIC_SUMMARY	Характеризує завантаження процесорних ресурсів	Відібрано	Безпосередньо відображає ресурсну напруженість системи
Database Wait Time Ratio	V$SYSMETRIC DBA_HIST_ SYSMETRIC_SUMMARY	Відображає частку непродуктивного часу	Відібрано	Є інтегральним індикатором затримок
Buffer Cache Hit Ratio	V$SYSMETRIC DBA_HIST_ SYSMETRIC_SUMMARY	Характеризує ефективність використання буферного кешу	Відібрано	Дає узагальнену оцінку роботи пам'яті та впливу на I/O
Active Sessions	DBA_HIST_ACTIVE_SESS_ HISTORY	Відображає фактичний рівень навантаження на БД	Відібрано	Чутливий до пікових навантажень і черг на ресурси
Wait Events / Wait Class	V$SYSTEM_EVENT / V$WAITCLASSMETRIC / DBA_ HIST_SYSTEM_EVENT	Відображає типи та інтенсивність затримок	Відібрано	Дає змогу виявляти причини деградації продуктивності
I/O waits / User I/O / System I/O	V$WAITCLASSMETRIC / DBA_ HIST_SYSTEM_EVENT	Характеризує затримки вводу-виводу	Відібрано	Важливий для виявлення дискових та мережевих вузьких місць
DB Block Gets	V$SYSSTAT / DBA_HIST_SYSSTAT	Відображає логічні звернення до блоків	Не відібрано	Має вузький технічний зміст і дублює інформацію інших метрик
Background CPU Time	V$SYS_TIME_MODEL	Характеризує навантаження фонових процесів	Не відібрано	Менш інформативний для оцінювання стану з позиції користувацьких запитів
Executes	V$SYSSTAT / DBA_HIST_SYSSTAT	Кількість виконань SQL-операцій	Частково враховано	Може використовуватись як допоміжний контекстний показник
Physical Reads	V$SYSSTAT / DBA_HIST_SYSSTAT	Інтенсивність фізичних читань	Частково враховано	Має діагностичну цінність, але доцільно використовувати в агрегованому вигляді

Базові ознаки безпосередньо відображають основні механізми деградації: ресурсну напруженість, непродуктивний час, ефективність буферного кешу, активне навантаження, внутрішні очікування та затримки I/O. Їх спільне використання дає змогу оцінювати не лише факт відхилення, а й можливу природу проблемного стану.

Окремий результат стосується ознак, вплив яких не є очевидним при ізольованому розгляді. Executes і Physical Reads не включено до базового набору як самостійні критерії, однак їх збережено у статусі «частково враховано». Вони виконують контекстну функцію: характеризують інтенсивність SQL-операцій та фізичного читання і можуть уточнювати інтерпретацію CPU, I/O та wait-показників в агрегованому вигляді. Таким чином, методологія не зводить відбір до бінарного рішення «залишити/видалити», а підтримує проміжний клас ознак, значення яких проявляється у взаємодії з базовими характеристиками.

DB Block Gets і Background CPU Time не відібрано. Перший показник має вузький технічний зміст і частково дублює інформацію інших метрик, а другий менш безпосередньо пов'язаний зі станом,

який спостерігає користувацьке навантаження. Їх вилучення з основного простору зменшує кількість слабкоінформативних або дублюючих змінних, не усуваючи ключових ресурсних та часових характеристик.

Науковий результат цього етапу полягає у відтворюваному переході від широкої телеметрії Oracle до трирівневого простору ознак: базового, контекстного та вилученого. На відміну від загального автоматизованого конструювання ознак у реляційних базах [21], рішення прив'язані до семантики конкретних Oracle-показників і супроводжуються поясненням їхньої діагностичної ролі. Подальші дослідження доцільно спрямувати на перевірку стійкості сформованого простору ознак на розширених наборах телеметричних даних, інших конфігураціях Oracle Database та додаткових типах інцидентів.

3.3. Результати призначення міток інцидентів. Третю задачу вирішено шляхом поєднання порогових правил з нечіткими функціями належності. Детерміністична частина забезпечує явну інтерпретацію двох сценаріїв: паралельність визначається за кількістю блокувальних затримок понад 5 і часткою конкурентних очікувань понад 40 %, а I/O-інцидент – за часткою I/O-очікувань понад 60 % та перевищенням 95-го перцентиля інтенсивності логування. Нечітка частина використовує поріг агрегованої оцінки 0,6.

Отриманий результат є адаптивнішим за повністю фіксовану систему правил, оскільки сигмоїдна належність відображає поступове наближення метрики до критичної області. Водночас він зберігає пояснюваність: для кожної мітки можна вказати метрики, пороги та ваги, що вплинули на рішення. Це важливо для майбутньої СППР, де адміністратор повинен не лише отримати клас стану, а й зрозуміти причину спрацювання.

Недоліком є залежність базової істини від прийнятих правил і критичних точок. Без незалежної експертної або інцидентної розмітки можлива циркулярність: модель навчається відтворювати правила, що одночасно використовуються для оцінювання стану. Тому перед промисловим використанням пороги мають бути перевірені на реальних підтверджених інцидентах.

3.4. Результати формування вибірки та перевірки простору ознак засобами IELA. Результатом четвертої задачі стало перетворення відібраних базових і контекстних характеристик у часово впорядковану матрицю приблизно з 10 080 однохвилинних спостережень. Ковзне вікно з десяти снапшотів забезпечило врахування короткострокової динаміки, а розподіл 70/15/15 зберіг хронологічну цілісність між навчанням, валідацією і тестуванням. Класифікатор на цьому етапі використано як засіб перевірки придатності сформованого простору ознак до розмежування станів, а не як заміну процедурі їх доменного відбору.

Застосування інформаційного критерію IG дало методичну основу для ранжування прихованих нейронів за зв'язком їхніх активацій з мітками стану. На відміну від класичного ELM, у якому весь випадково сформований прихований шар використовується без селекції, IELA дозволяє відкинути кандидати з низьким інформаційним внеском. Важливо, що цей етап виконується після формування телеметричного простору ознак: таблиця 3 описує відбір вхідних характеристик, тоді як IELA виконує селекцію елементів прихованого шару. У цьому експерименті акцент зроблено на перевірці придатності схеми; окрема кількісна оцінка скорочення прихованого шару не наводилася, тому перевага не інтерпретується як доведене зменшення складності для всіх конфігурацій.

Умовами відтворення результату є незмінний порядок часових спостережень, однакова процедура нормалізації для навчальної та тестової частин, відсутність витоку майбутніх значень у ковзне вікно та фіксація генератора випадкових параметрів ELM.

3.5. Оцінювання якості, порівняння, обмеження та перспективи. Перший етап оцінювання передбачає аналіз динаміки зміни значень функції виявлення F(x) у випадку підтвердженого інциденту. У межах 45-хвилинного часового вікна наведеного на рисунку 2 за робочого порогу τ=0,60 отримано TP = 5, FP = 10 та FN = 6. При цьому TP (True Positive) відповідає випадкам, коли інцидент фактично мав місце і був правильно виявлений моделлю; FP (False Positive) – випадкам, коли інцидент був відсутній, однак модель помилково класифікувала стан як інцидентний; FN (False Negative) – випадкам, коли інцидент фактично мав місце, але модель його не виявила. Точність позитивних прогнозів (Precision) становить 0,33, а повнота (Recall) 0,45. Загальна точність класифікації (Accuracy) становить 0,64, тобто модель правильно класифікувала приблизно 64 % усіх спостережень у межах аналізованого часового вікна. Водночас це значення слід інтерпретувати разом із показниками точності позитивних прогнозів і повноти, оскільки наявність значної кількості хибнопозитивних та хибнонегативних спрацювань свідчить про необхідність подальшого налаштування робочого порогу та калібрування моделі з урахуванням особливостей розподілу класів.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

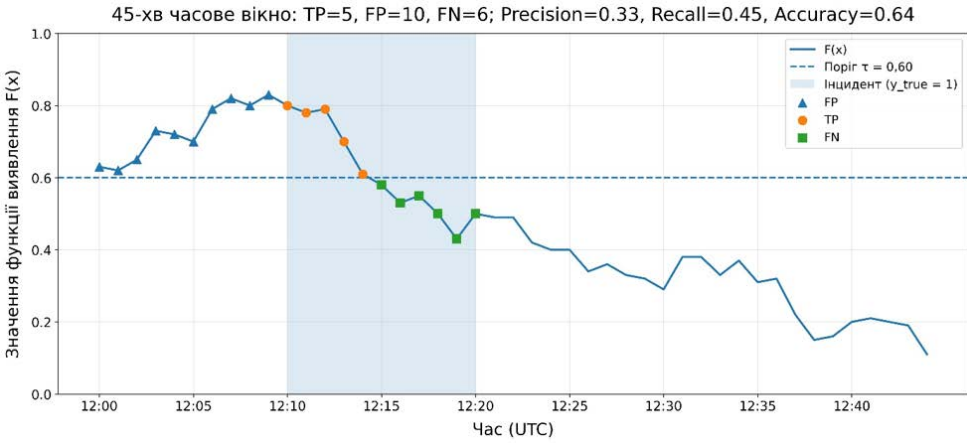


Рис. 2. Відстеження спрацювань у 45-хвилинному інцидентному вікні

Рисунок 3 демонструє аналіз із кривою ROC, де бірюзова крива вказує на сильне розмежування між TP та FP частками порівняно з червоною діагональною референтною лінією, що доводить високу здатність виявляти аномальні явища в залежності від встановлених порогів.

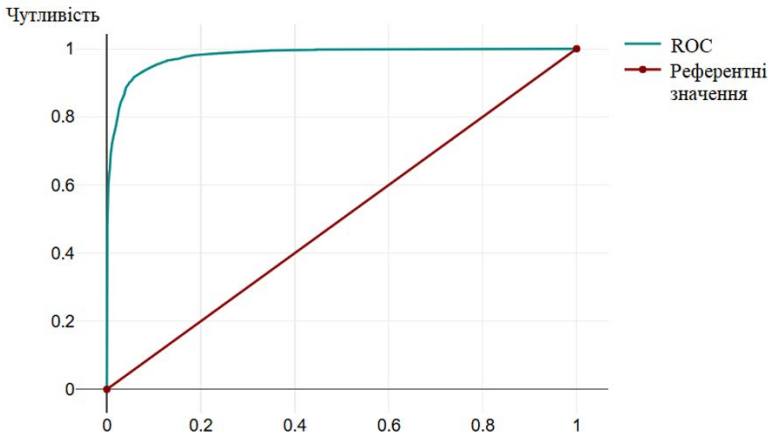


Рис. 3. ROC-крива прототипної моделі у порівнянні з випадковим класифікатором

Калібраційний графік на рисунку 4 демонструє загальне наближення спостережуваних часток позитивних випадків до середніх передбачуваних балів, хоча у верхній частині діапазону помітне завищення прогнозу. Це узгоджується з надлишком хибнопозитивних рішень у робочій точці та підтверджує потребу в калібруванні порогу.

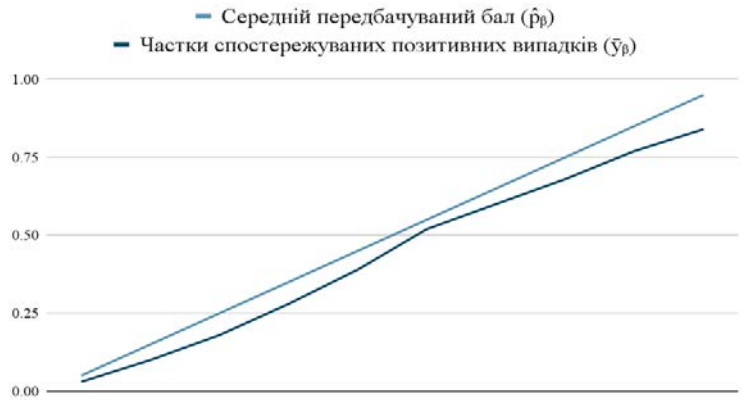


Рис. 4. Калібраційний графік передбачуваних балів і спостережуваних позитивних часток

Таблиця 4

Довірчі інтервали Вілсона для калібраційних груп

n_β	Нижня межа	Верхня межа
120	0.015	0.056
115	0.058	0.165
110	0.118	0.260
105	0.203	0.373
98	0.299	0.489
92	0.418	0.618
85	0.496	0.693
78	0.578	0.771
72	0.664	0.849
65	0.742	0.905

Ширина довірчих інтервалів визначається спільним впливом кількості спостережень n_β у β-й калібраційній групі та величини оцінюваної частки, тому її не можна пояснювати лише зменшенням n_β. Ці інтервали доцільно використовувати як показник невизначеності калібрування, а не як самостійну оцінку точності класифікатора.

На повному тижневому наборі частка правильно класифікованих спостережень становила близько 0,64. Це значення характеризує якість класифікації за однієї фіксованої робочої точки, тоді як ROC-крива відображає співвідношення між часткою істинно позитивних і хибнопозитивних результатів у всьому діапазоні порогів. Тому помірне значення accuracy за вибраного порога та візуально виражена ROC-крива не суперечать одне одному. Водночас отримані результати свідчать про необхідність подальшого вибору оптимального порога класифікації та перевірки калібрування моделі. Для повнішого оцінювання її якості доцільно окремо навести AUROC, precision, recall, F1-міру та матрицю помилок для повного тестового набору.

Пряме кількісне порівняння з логовими моделями [5-13, 18] обмежене різними джерелами даних, типами аномалій і способами формування базової істини. Перевага запропонованої методології полягає не у доведеній вищій точності класифікатора, а у цілісному ланцюжку: інтеграція V$ та AWR, формування кандидатних характеристик, пояснюваний поділ на базові, контекстні та надлишкові ознаки, призначення міток і подальша перевірка сформованого простору засобами ELM/IELA.

Основні обмеження дослідження: один тиждень спостережень; один вузол та один екземпляр Oracle 21c; обмежена кількість типів інцидентів; частково синтетичне або правилове формування міток; класовий дисбаланс; відсутність зовнішньої валідації та числового AUROC. Подальший розвиток доцільно спрямувати на незалежну розмітку підтверджених інцидентів, оптимізацію порогу за вартістю помилок, розширення набору середовищ, онлайн-оновлення ваг і нейронів IELA та перевірку на багатовузлових конфігураціях.

Практичне використання можливе за наявності синхронізованої телеметрії, стабільного доступу до V$ та AWR, перевірених порогів для конкретного середовища і процедури регулярного перегляду словника ознак. Особливо важливо повторно оцінювати контекстні та вилучені показники при зміні типу навантаження, конфігурації пам'яті, дискової підсистеми або архітектури Oracle. За цих умов методологія може слугувати основою прототипу СППР для раннього виявлення деградації функціонального стану РБД.

4. Висновки.

1. Сформовано шляхово-реляційний спосіб інтеграції десяти потокових та історичних подань Oracle. Використання METRIC_ID, SNAP_ID, INSTANCE_NUMBER і часових меж дозволило привести V$ та AWR до спільної UTC-шкали та сформувати первинний набір кандидатних характеристик без розриву між короткостроковим і ретроспективним аналізом.

2. Розроблено процедуру доменного оцінювання та відбору телеметричних ознак. Із десяти кандидатних груп шість визначено як базові, дві – як контекстні, що враховуються частково, і дві – як надлишкові або вузькоспеціалізовані. Це дозволило зберегти показники CPU, I/O, очікувань, пам'яті та активного навантаження, не втратити неочевидний контекст Executes і Physical Reads та виключити DB Block Gets і Background CPU Time з основного простору. Відмінною рисою результату є не бінарне відсікання, а трирівневе, пояснюване рішення щодо ролі кожної ознаки.

3. Розроблено комбінований механізм призначення міток, який поєднує правила оцінювання одночасної активності в базі даних і навантаження на підсистему введення-виведення з пороговим

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

значенням інтегральної нечіткої оцінки 0,6. Правила враховують показники активних сеансів, блокувань та очікувань ресурсів, а також інтенсивність і тривалість операцій читання та запису даних. Перевагою механізму є можливість пояснити, які метрики, порогові значення та вагові коефіцієнти вплинули на призначення мітки. Його обмеженням є залежність від обґрунтованості встановлених порогів і ваг, а також необхідність незалежної перевірки на підтверджених інцидентах.

4. На основі відібраного простору ознак сформовано часову вибірку приблизно з 10080 однохвилинних записів, ковзним вікном 10 хв та розподілом 70/15/15. ELM/IELA використано як наступний етап перевірки придатності цього простору до класифікації станів; інформаційний критерій IELA ранжує приховані нейрони, але не підміняє доменний відбір вхідних телеметричних характеристик.

5. У вибраній робочій точці частка правильних класифікацій становила близько 0,64; у 45-хвилинному вікні отримано precision = 0,33 та recall ≈ 0,45. ROC- і калібраційна діагностика показали потенціал сформованого простору ознак для розмежування станів за інших порогів, але велика кількість FP підтвердила необхідність калібрування, розширення незалежної розмітки, статистичної перевірки внеску окремих ознак та зовнішньої валідації.

Конфлікт інтересів. Автори декларують, що не мають конфлікту інтересів стосовно даного дослі-дження, в тому числі фінансового, особистісного характеру, авторства чи іншого характеру, що міг би вплинути на дослідження та його результати, представлені в даній статті.

Фінансування. Дослідження проводилося без фінансової підтримки.

Доступність даних. Дані будуть надані за обґрунтованим запитом.

Використання засобів штучного інтелекту. Автори підтверджують, що не використовували тех-нології штучного інтелекту при створенні представленої роботи.

Внесок авторів. Олександр Чикалов – концептуалізація, аналіз джерел, методологія, написання основного тексту; Олег Берест – рецензування та редагування, формування висновків і перевірка узгодженості структури.

References:

1. James, C., Corbett, J., Dean, J., Epstein, M., Fikes, A., Frost, C., Furman, J.J., Ghemawat, S., Gubarev, A., Heiser, C., Hochschild, P., et al. (2013). Spanner: Google's globally distributed database. *ACM Transactions on Computer Systems (TOCS)*, Vol. 31. PP. 1–22. DOI:10.1145/2491245.

2. Yang, Z., Yang, C., Han, F., Zhuang, M., Yang, B., Yang, Z., Cheng, X., Zhao, Y., Shi, W., Xi, H., et al. (2022). OceanBase: a 707 million tpmC distributed relational database system. *Proceedings of the VLDB Endowment*, Vol. 15. PP. 3385–3397. DOI:10.14778/3554821.3554837.

3. Huang, D., Liu, Q., Cui, Q., Fang, Z., Ma, X., Xu F., Shen, L., Tang, L., Zhou, Y., Huang, M., et al. (2020). TiDB: a Raft-based HTAP database. *Proceedings of the VLDB Endowment*, Vol. 13. PP. 3072–3084. DOI:10.14778/3415478.3415535.

4. He, P., Liu, J., Lyu, M.R., Zhu, J., He, S. (2023). Loghub: A large collection of system log datasets towards automated log analytics, arXiv:2307.09227.

5. Zhang, X., Xu, Y., Lin, Q., Qiao, B., Zhang, H., Dang, Y., Xie, C., Yang, X., Cheng, Q., Li, Z., et al. (2019). Robust log-based anomaly detection on unstable log data. Proceedings of the 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, PP. 807–817. DOI:10.1145/3338906.3338964.

6. Du, M., Li, F., Zheng, G., Srikumar, V. (2017). Deeplog: Anomaly detection and diagnosis from system logs through deep learning. Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, PP. 1285–1298. DOI:10.1145/3133956.3134015.

7. Meng, W., Liu, Y., Zhu, Y., Zhang, S., Pei, D., Liu, Y., Chen, Y., Zhang, R., Tao, S., Sun, P., et al. (2019). Loganomaly: Unsupervised detection of sequential and quantitative anomalies in unstructured logs. Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI), PP. 4739–4745.

8. Yin, K., Yan, M., Xu, L., Xu, Z., Li, Z., Yang, D., Zhang, X. (2020). Improving log-based anomaly detection with component-aware analysis. 2020 IEEE International Conference on Software Maintenance and Evolution (ICSME), PP. 667–671. DOI:10.1109/ICSME46990.2020.00077.

9. Guo, H., Yuan, S., Wu, X. (2021). Logbert: Log anomaly detection via BERT. Proceedings of the International Joint Conference on Neural Networks (IJCNN). IEEE, PP. 1–8. DOI:10.1109/IJCNN52387.2021.9533689.

10. Jia, T., Li, Y., Yang, Y., Huang, G., Wu, Z. (2022). Augmenting log-based anomaly detection models to reduce false anomalies with human feedback. Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, PP. 3081–3089. DOI:10.1145/3534678.3539393.

11. Jia, T., Wu, Y., Hou, C., Li, Y. (2021). LogFlash: Real-time streaming anomaly detection and diagnosis from system logs for large-scale software systems. Proceedings of the IEEE 32nd International Symposium on Software Reliability Engineering (ISSRE), PP. 80–90. DOI:10.1109/ISSRE52982.2021.00021.

12. Jia, T., Yang, L., Chen, P., Li, Y., Meng, F., Xu, J. (2017). Logsed: Anomaly diagnosis through mining time-weighted control flow graph in logs. Proceedings of the IEEE 10th International Conference on Cloud Computing (CLOUD), PP. 447–455. DOI:10.1109/CLOUD.2017.60.

13. Kim, J., Savchenko, V., Shin, K., Sorokin, K., Jeon, H., Pankratenko, G., Markov, S., Kim, C.-J. (2020). Automatic abnormal log detection by analyzing log history for providing debugging insight. Proceedings of the 42nd International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP), PP. 71–80. DOI:10.1145/3377813.3381363.

14. Lin, Q., Zhang, H., Lou, J., Zhang, Y., Chen, X. (2016). Log clustering based problem identification for online service systems. Proceedings of the 38th International Conference on Software Engineering Companion, PP. 102–111. DOI:10.1145/2889160.2889232.

15. Huo, Y., Li, Y., Su, Y., He, P., Xie, Z., Lyu, M.R. (2023). AutoLog: A log sequence synthesis framework for anomaly detection, arXiv:2308.09324.

16. Landauer, M., Skopik, F., Wurzenberger, M. (2023). A critical review of common log data sets used for evaluation of sequence-based anomaly detection techniques, arXiv:2309.02854.

17. Wang, C., Huang, X., Qiao, J., Jiang, T., Rui, L., Zhang, J., Kang, R., Feinauer, J., McGrail, K.A., Wang, P., et al. (2020). Apache IoTDB: Time-series database for internet of things. *Proceedings of the VLDB Endowment*, Vol. 13. PP. 2901–2904. DOI:10.14778/3415478.3415504.

18. Zhang, L., Jia, T., Jia, M., Li, Y., Yang, Y., Wu, Z. (2024). Multivariate log-based anomaly detection for distributed database. Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, PP. 12. DOI:10.1145/3637528.3671725.

19. Oraculix. Monitoring Oracle performance with AppDynamics, 2013. [Online]. Retrieved from: https://oraculix.com/2013/02/25/monitoring-oracle-performance-with-appdynamics/.

20. Oracle. Instance tuning using performance views. Database Performance Tuning Guide, 2023. [Online]. Retrieved from: https://docs.oracle.com/en/database/oracle/oracle-database/23/tgdba/instance-tuning-using-performance-views.html.

21. Lam, H.T., Thiebaut, J.M., Sinn, M., Chen, B., Mai, T., Alkan, O. (2017). One button machine for automating feature engineering in relational databases, arXiv:1706.00327. DOI:10.48550/arXiv.1706.00327.

22. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L.B., Bourne, P.E., et al. (2016). The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data, Vol. 3. Article 160018. DOI:10.1038/sdata.2016.18.

23. Buchanan, E., Crain, S., Cunningham, A., Johnson, H., Stash, H., Papadatou-Pastou, M., Isager, P., Carlsson, R., Aczel, B. (2019). Getting started creating data dictionaries: How to create a shareable dataset. OSF Preprints, DOI:10.31219/osf.io/vd4y3.

Відомості про авторів

Англ.	Укр.
Oleksandr Chykalov Postgraduate Student at the Department of Computer Science, Sumy State University chikalov2012@gmail.com ORCID: 0009-0004-5132-7430	Чикалов Олександр Сергійович аспірант кафедри комп'ютерних наук, Сумський державний університет hikalov2012@gmail.com ORCID: 0009-0004-5132-7430
Oleh Berest Senior Lecturer in the Department of Computer Science, Sumy State University o.berest@cs.sumdu.edu.ua ORCID: 0000-0003-0183-9871	Берест Олег Борисович старший викладач кафедри комп'ютерних наук, Сумський державний університет o.berest@cs.sumdu.edu.ua ORCID: 0000-0003-0183-9871

Дата надходження статті: 29.05.2026
Дата надходження виправленої версії статті: 08.07.2026
Дата прийняття статті: 10.07.2026
Дата публікації статті: 26.10.2026