e-ISSN 1607-3274 Радіоелектроніка, інформатика, управління. 2021. № 1
p-ISSN 2313-688X Radio Electronics, Computer Science, Control. 2021. № 1

УДК 004.8

РОЗРОБКА МЕТОДУ ІДЕНТИФІКАЦІЇ СТАНУ КОМП'ЮТЕРНОЇ СИСТЕМИ НА ОСНОВІ АЛГОРИТМУ «ISOLATION FOREST»

Гавриленко С. Ю. – д-р техн. наук, доцент, професор кафедри «Обчислювальна техніка та програмування», Національний технічний університет «Харківський політехнічний інститут», Харків, Україна.

Шевердін І. В. – аспірант кафедри «Обчислювальна техніка та програмування», Національний технічний університет «Харківський політехнічний інститут», Харків, Україна.

АНОТАЦІЯ

Актуальність. Розглянуто задачу ідентифікації стану комп'ютерної системи. Об'єктом дослідження є процес ідентифікації стану комп'ютерної системи. Предметом дослідження є методи та засоби ідентифікації стану комп'ютерної системи.

Мета. Метою роботи є розробка методу ідентифікації стану комп'ютерної системи.

Метод. Розроблено метод ідентифікації стану комп'ютерної системи на основі комплексного використання процедури групування нерозмічених вихідних даних та технології машинного навчання на основі алгоритму «Isolation Forest», який надає можливість ідентифікувати стан комп'ютерної системи і виділити назву процесу, який спричинив аномальний стан. Для цього запропоновано процедуру та розроблено програмний додаток для збору статистичних даних у вигляді подій функціонування операційної системи та виконано їх аналіз. Отримано, що найбільш інформативними є операції читання та запису. Для формування єдиного датасету, операції читання та запису зіставлено з назвою процесу та об'єднано в один масив груп подій, що надалі дозволяє виділити процес, який спричиняє аномальний стан комп'ютерної системи. За результатами дослідження, у якості складової методу ідентифікації стану комп'ютерної системи використано ансамблевий алгоритм «Isolation Forest». Проведено оцінку точності та оперативності розробленого методу ідентифікації стану комп'ютерної системи .

Результати. Розроблений метод реалізований програмно і досліджений під час розв'язання задачі ідентифікації аномалій функціонування комп'ютерної системи.

Висновки. Проведені експерименти підтвердили працездатність запропонованого методу, що надає можливість рекомендувати його для практичного використання з метою підвищення оперативності ідентифікації стану комп'ютерної системи та використання його у якості експрес-методу. Перспективи подальших досліджень можуть полягати в розробці ансамблю нечітких дерев рішень на основі запропонованого методу, оптимізації його програмних реалізації.

КЛЮЧОВІ СЛОВА: комп'ютерна система, події операційної системи, аномальний стан, ідентифікація, машинне навчання, алгоритм «Isolation Forest».

АББРЕВІАТУРИ

КС – комп'ютерна система;

ОС – операційна система;

NB – метод Байєса (Naive Bayes);

KNN – метод *k* найближчих сусідів (k Nearest Neighbors);

DT – метод дерев рішень (Decision Trees);

SVM – метод опорних векторів (Support Vector Machine);

RF – метод випадкового лісу (Random Forest);

IF – метод ізолюючого лісу (Isolation Forest);

J48 – алгоритм C4.5, реалізований на мові програмування Java;

iTree – ізолююче дерево ухвалення рішень.

НОМЕНКЛАТУРА

X – вихідні дані (події ОС);

x_{ij} – об'єкт події ОС із набору вихідних даних;

m– кількість показників об'єкту;

k – кількість класів;

C – множина класів;

f – алгоритм класифікації з областю визначення X та областю значення C;

AS – показник аномальності (Anomaly Score).

K – статистичне ядро, симетрична, але не обов'язково додатна функція з інтегралом рівним одиниці;

h – параметр згладжування, $h > 0$;

X – вихідні дані у вигляді нерозміченого масиву груп подій;

$\hat{f}_h(x)$ – функція щільності розподілу імовірності випадкової величини;

$H(i)$ – гармонічне число;

γ = 0,5772156649 (константа Ейлера);

$h(x)$ – глибина гілки, що містить це спостереження, що еквівалентно кількості розщеплень, необхідних для ізоляції цієї точки;

$E(h(x))$ – середнє значення $h(x)$ з набору Isolation Tree;

$c(n)$ – середнє значення $h(x)$ n-спостережень.

ВСТУП

При вирішенні завдань, пов'язаних з діагностикою та захистом комп'ютерних інформаційних ресурсів, центральною є задача оперативного виявлення аномальної поведінки комп'ютерної системи в умовах зовнішніх впливів.

Проведені дослідження існуючих комп'ютеризованих систем ідентифікації станів дозволили ви-

DOI 10.15588/1607-3274-2021-1-11

явити ряд обмежень їх використання. Так при появі аномалій, породжених вторгненнями в КС з невстановленими або нечітко визначеними властивостями, сучасні методи не завжди залишаються ефективними і вимагають тривалих часових та програмно-апаратних ресурсів для їх відповідної адаптації, що призводить до зниження показників оперативності ідентифікації стану КС.

На сьогодні комп'ютерна система характеризується великим обсягом показників її функціонування. Це призводить до наявності труднощів з адекватного відбору показників для ідентифікації стану КС в умовах зовнішніх впливів і розробки критерію оцінки, що відповідає обраним показникам.

Об'єктом дослідження є процес ідентифікації стану комп'ютерної системи.

Предметом дослідження є методи та засоби ідентифікації стану КС.

Існує безліч методів ідентифікації, які використовують різний математичний апарат і різні підходи при реалізації [1–3]. Одним із найбільш поширених методів аналізу великих обсягів даних (data mining) є методи машинного навчання (machine learning). Однак, ефективність цих методів залежить від конкретної розв'язуваної задачі.

Метою роботи розробка методу ідентифікації стану комп'ютерної системи.

1 ПОСТАНОВКА ЗАДАЧІ

Будемо вважати, що функціонування комп'ютерної системи є сукупністю подій операційної системи. Виконати аналіз подій операційної системи та сформувати вихідні дані X у вигляді нерозміченого масиву груп подій, зіставлених з назвою процесу, тобто кожен об'єкт $x_i \in X$ задати у вигляді деякого вектору $C = \{c_{i1}, c_{i2}, ..., c_{im}\}$. Тоді постановка завдання ідентифікації стану комп'ютерної системи визначається наступним чином. Нехай $C = \{c_1, c_2, ..., c_k\}$ – кінцева множина класів. Існує невідоме відображення f: $X \to C$, причому його значення відомі тільки на елементах кінцевої сукупності $T = \{(x_1, c_1), ..., (x_n, c_k) \subset X \times C$. Потрібно побудувати алгоритм f: $X \to C$, здатний класифікувати довільний стан КС $x_i \in X$. Для цього, для кожного елементу кінцевої сукупності (x_i, c_i) визначити критерій класифікації у вигляді функції *AS*. Визначити поріг бінаризації. Поріг бінаризації задається якщо відома приблизна частка аномалій в даних (для цього вибирається відповідний квантиль) або розраховується за умови щоб дисперсія між класами була мінімальною, $D \to \min$.

Використовуючи значення функції *AS* та значення порогу, детектувати наявність аномалій: в заданій множині X для кожного елемента $x_i \in X$ видати 0, якщо цей об'єкт відноситься до класу нормальних даних, і 1, якщо цей об'єкт є аномальним. За наявності аномального стану, визначити процес, що його спричинив.

2 ОГЛЯД ЛІТЕРАТУРИ

Функціонування КС характеризується великою кількістю процесів. Для аналізу цих даних і їх класифікації використовуються складні математичні алгоритми, що базуються на машинних методах навчання. Найбільш популярні алгоритми машинного навчання наведено в [4,5]. Так, прикладом імовірнісного методу класифікації є метод Байєса [6]. Перевагою методу є: висока швидкість роботи, легка інтерпретація результатів роботи алгоритму, проста реалізація алгоритму у вигляді програми. Незважаючи на наведені переваги, метод Байеса має не достатню точність класифікації і нездатний враховувати залежність результату класифікації від поєднання ознак.

Метод k найближчих сусідів відноситься до метричних методів і вважається найпростішим класифікатором [4, 7]. Перевагою даного методу є проста реалізація, наявність гарної теоретичної бази, адаптація під потрібне завдання вибором метрики або ядра. До недоліків відносяться: недостатня продуктивність в реальних завданнях, так як число сусідів, які використовуються для класифікації, буде досить великим; труднощі в наборі відповідних ваг і визначенням, які ознаки необхідні для класифікації; залежність від обраної метрики відстані між об'єктами.

Одним із найкращих методів класифікації є метод опорних векторів [8]. Недоліки методу опорних векторів полягають в наступному: неможливість калібрування ймовірності попадання в певний клас, підходить тільки для вирішення завдань з 2 класами, параметри моделі складно інтерпретувати.

Нейронні мережі також активно використовуються у зв'язку з появою великих обсягів даних і великих обчислювальних можливостей [9]. Їх ефективність досить висока, тому що вони генерують фактично велике число регресійних моделей (які використовуються в рішенні задач класифікації статистичними методами). Однак, будь-який метод, заснований на нейронних мережах, ніколи не дасть класифікатор потрібної якості, якщо набір навчальної вибірки не буде достатньо повним для того завдання, з якою доведеться працювати в системі.

Метод дерев рішень відноситься до логічних методів класифікації [10]. Головною перевагою методу є висока продуктивність навчання і прогнозування, такі дерева рішень можна легко візуалізувати і інтерпретувати. Недоліком методу є відносно невисока точність прогнозів, так як побудова класифікатору істотно залежать від вхідних параметрів [11]; структури даних, природи їх виникнення [12]. За умови відсутності розмітки даних, має місце проблема проведення відбору моделей та перевірки якості їх роботи (за допомогою кроссвалідації або тестування на відкладеній вибірці [13]. Для подолання вищенаведених недоліків розроблено методи, засновані на використанні ансамблів з декількох класифікаторів (сотень і навіть ти-

DOI 10.15588/1607-3274-2021-1-11

сяч). Ансамблі покращують якість, знижують залежність моделей від досліджених даних та вхідних параметрів, підвищуючи стабільність результатів. За якістю одержуваних прогнозів, ансамблі з декількох моделей часто перевершують інші методи [14–19].

Складовими ансамблевих алгоритмів можуть бути класифікатори з учителем та без учителя. Класифікатори без учителя є більш оперативними, оскільки не потребують навчання. Такі методи не потребують розмічених даних і намагаються самостійно знайти шаблони безпосередньо з вихідних даних. При цьому в більшості практичних додатків розмітка нормальних та аномальних класів даних відсутня, у зв'язку з чим проблема виявлення аномалій розглядається як завдання навчання без учителя [15, 16]. Використання дерев рішень у якості базових класифікаторів ансамблів рішень дозволяє, автоматично виконати відбір інформативних предикатів з урахуванням можливості взаємодії між ними.

Таким чином, проведені дослідження надали можливість виявити ряд обмежень використання існуючих методів, що у сукупності з наявністю різних типів даних, що характеризують стан функціонування комп'ютерної системи, призводить до суттєвої розбіжності якості та слабкої практичної придатності окремих класифікаторів. Крім того відомі методи виконують тільки ідентифікацію стану КС та не визначають процес, який спричинив аномальний стан.

У зв'язку з цим особливої актуальності набувають питання удосконалення та розробки нових методів ідентифікації стану КС.

3 МАТЕРІАЛИ ТА МЕТОДИ

Відповідно до постановки задачі, в рамках даного дослідження розроблено метод ідентифікації стану КС, який відрізняється від відомих методів використанням у якості класифікатора ансамблю iTree та наявністю процедури ідентифікації процесу, що спричинив цей аномальний стан.

Для формування вихідних даних виконано аналіз подій операційної системи Windows 10, а саме: ім'я процесу, вид операції, шлях до файлу виконання (табл. 1).

Аналіз подій ОС показав, що всі процеси ОС взаємодії з апаратною пам'яттю, так чи інакше можливо звести до операцій читання та запису. До операції запису слід також віднести операцію видалення чи створення, файлу або ключа реєстру.

Назва процесу, операції читання та запису і шлях до файлу виконання є важливим індикатором роботи ОС та надає можливість визначити, який процес і скільки раз ініціював операції та над якими ключами чи файлами він виконував дію.

Отримано, що статистика операцій читання та запису є закономірною для окремих процесів та характеризує стан комп'ютерної системи. Як правило, системні події мають відносно невелику кількість операцій запису та велику кількість операцій читання. Окремі вірусні події характеризуються великою кількістю операцій запису (в тому числі, запису системних конфігурацій налаштування ОС), які при нормальному стані функціонування ОС виконуються зрідка.

У якості вихідних даних було виділено найбільш розповсюджені операції читання файлів та ключів реєстру (RegOpenKey, RegQueryValue, RegQueryKey, ReadFile, QueryDirectory, RegEnumKey, QueryBasicInformationFile, QueryStandardInformationFile, RegEnumValue, QueryNameInformationFile, QuerySecurityFile, RegCreateKey, FileSystemControl, QueryRemoteProtocolInformation, QueryNetworkOpenInformationFile, RegQueryKeySecurity, QueryAllInformationFile, QueryAttributeTagFile, QueryNormalizedNameInformationFile, QueryEAFile, QueryIdInformation, NotifyChangeDirectory, QueryPositionInformationFile, QueryStreamInformationFile, RegQueryMultipleValueKey, QueryEaInformationFile, QueryFileInternalInformationFile, QueryLinks та операції запису (WriteFile, RegSetValue, CreateFile, SetEndOfFileInformationFile, RegCreateKey, FileSystemControl, RegDeleteValue, SetDispositionInformationFile, SetRenameInformationFile, SetAllocationInformationFile, RegDeleteKey, SetBasicInformationFile, SetPositionInformationFile, SetSecurityFile, SetValidDataLengthInformationFile, SetLinkInformationFile, SetEAFile).

Так як присутня сукупність параметрів, то для формування вхідного датасету запропоновано процедуру, яка базується на багатофакторному групуванні даних (рис. 1). Для комбінації двох категорій у один датасет, об'єднано операції читання та запису в один масив груп подій, та зіставлено з назвою процесу.

На рис. 1 наведено приклад результату роботи програмного додатку групування вихідних даних за назвою процесу, за шляхом до файлів операційної системи, операціями запису та читання, котрі були описані вище. У результаті отримано масив даних, який містить назву процесу, шлях до файлу виконання чи ключа реєстру, назву функції та інформацію про кількість операцій читання або запису для неї.

Таблиця 1 – Події операційної системи Windows 10

Назва атрибуту	Тип даних	Опис події	Приклад події
ProcessName	Строкове значення	Назва процесу	Explorer.exe …
Operation	Строкове значення (Категорія)	Вид операції	RegCloseKey; ReadFile; RegOpenKey …
Path	Строкове значення	Шлях до файлу виконання або ключ реєстру	C:\Windows\System32\NgcCtnrSvc.dll; HKCU\Software\Classes …

DOI 10.15588/1607-3274-2021-1-11

	ProcessName	ImagePath	Operation_x	ReadCount	Operation_y	WriteCount
0	Explorer.EXE	C:\Windows\Explorer.EXE	RegOpenKey	48	RegCreateKey	4
1	Explorer.EXE	C:\Windows\Explorer.EXE	RegOpenKey	48	RegSetValue	4
2	Explorer.EXE	C:\Windows\Explorer.EXE	RegQueryKey	4	RegCreateKey	4
3	Explorer.EXE	C:\Windows\Explorer.EXE	RegQueryKey	4	RegSetValue	4
4	Explorer.EXE	C:\Windows\Explorer.EXE	RegQueryValue	37	RegCreateKey	4
5	Explorer.EXE	C:\Windows\Explorer.EXE	RegQueryValue	37	RegSetValue	4
6	MsMpEng.exe	C:\ProgramData\Microsoft\Windows Defender\plat...	FileSystemControl	2	WriteFile	15
7	MsMpEng.exe	C:\ProgramData\Microsoft\Windows Defender\plat...	QueryAllInformationFile	4	WriteFile	15
8	MsMpEng.exe	C:\ProgramData\Microsoft\Windows Defender\plat...	QueryNameInformationFile	6	WriteFile	15
9	SearchFilterHost.exe	C:\Windows\system32\SearchFilterHost.exe	QueryNameInformationFile	3	0	0
10	SearchFilterHost.exe	C:\Windows\system32\SearchFilterHost.exe	RegOpenKey	3	0	0

Рисунок 1 – Приклад результату групування

Аналіз результату групування дозволив отримати статистику масиву груп подій. Статистика результату включає кількість операцій читання і запису (count), середнє значення (mean), середнє квадратичне відхилення (std), мінімальне (min) та максимальне (max) значення, а також значення для 25-го (25%), 50-го (50%), 75-го (75%) перцентилів (рис. 2), які надалі можуть бути використані для аналізу даних, що надходять до класифікатору та вибору типу класифікатору.

Для оцінки отриманого масиву груп подій на предмет наявності аномалій було використано непараметричний метод оцінки функції щільності імовірності випадкової величини за вибіркою, а саме метод ядрової оцінки щільності розподілу:

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i).$$

Результати аналізу розподілу даних у вигляді графіку розподілу наведено на (рис. 3). Як видно із графіків для операцій читання і запису, розподіли даних в обох випадках далекі від нормального закону розподілу. Операції читання та запису мають позитивний довгий хвіст, основна частина розподілу зосереджена зліва. Хвостовий розподіл набагато перевищує піки справа, що потенційно може сигналізувати про наявність аномальних викидів. Присутність невеликих хвостів є індикатором аномальної зміни стану окремих процесів.

	ReadCount	WriteCount
count	273.000000	44.000000
mean	1530.347985	21.681818
std	6167.488208	20.658657
min	1.000000	3.000000
25%	9.000000	8.000000
50%	55.000000	15.000000
75%	415.000000	17.000000
max	34847.000000	61.000000

Рисунок 2 – Статистика розподілу операцій читання і запису

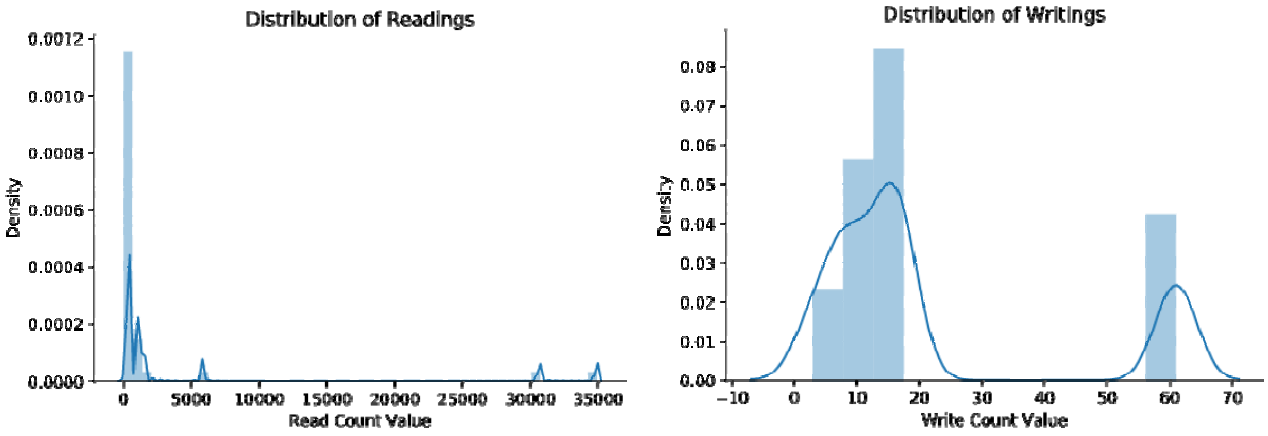

Рисунок 3 – Графіки розподілу операцій читання та запису

DOI 10.15588/1607-3274-2021-1-11

Отримані результати розподілу вихідних даних показали, що вони можуть бути використаними у якості вихідних даних методу ідентифікації аномалій функціонування комп'ютерної системи.

У якості складової методу ідентифікації стану КС використано алгоритм на основі ізолюючого лісу. IF – це ансамблевий алгоритм без учителя, який є варіацією ідеї випадкового лісу [17–19] та замість спроби побудувати модель звичайних екземплярів, ізолює аномальні точки в наборі даних.

IF є ансамблем iTree, де секції дерев створюються шляхом першого випадкового вибору об'єкту, а потім вибору випадкового значення поділу між мінімальним і максимальним значеннями обраного об'єкта.

Мірою нормальності спостереження за даним деревом є глибина гілки $h(x)$, що містить це спостереження, що еквівалентно кількості розщеплень, необхідних для ізоляції цієї точки. Аномальні значення потрапляють в листя на невеликій глибині дерева і тим самим детектуються. Показником аномальності є функція *AS*, яка для даного об'єкту видає деякий «рейтинг» аномальності.

Оцінка аномалії *AS* екземпляра *x* визначається як [19]:

$$AS(x,n) = 2^{-\frac{E(h(x))}{c(n)}},$$

де $c(n)$ визначається наступним чином [20]:

$$c(n) = 2H(n-1) - (2(n-1)/n),$$

де $H(\mathrm{i})$ визначається як [20]:

$$H(i) = \ln(i) + \gamma.$$

Чим більше значення аномалії *AS*, тим більше вірогідність того, що досліджуваний об'єкт є аномальним.

Відповідно до алгоритму IF будується необхідне число дерев та проводиться класифікація об'єкту. Об'єкт класифікації буде віднесено кожним деревом до одного з двох класів: нормального чи аномального. Прийняття рішення відносно класу об'єкту виконується методом простого голосування, тобто на основі мета-алгоритму беггінгу

Перевагою цього методу є можливість якісної обробки, як безперервних, так і дискретних даних з великим числом ознак і класів, в тому числі з пропущеними значеннями ознак. Складність ізолюючого дерева – $O(n \log n)$, що ефективніше більшості інших алгоритмів. Метод не потребує істотних затрат пам'яті, на відміну від, наприклад, метричних методів, які часто потребують побудови матриці попарних відстаней, стійкий до прокляття розмірності.

Для порівняння, у якості складової методу ідентифікації стану КС з метою виявлення аномалій також було досліджено алгоритм KNN. KNN. – це простий непараметричний алгоритм, де для класифікації використовуються відстані (зазвичай евклідові), порахована до усіх інших об'єктів [4, 7].

4 ЕКСПЕРИМЕНТИ

Результати використання алгоритму IF для ідентифікації стану КС наведено на рис. 4–7. Рис. 4–5 відображають залежність *AS* від значення кількості операцій читання або запису (Readings or Writings) для стану КС у режимі простою без запуску будь яких процесів, який можливо трактувати як нормальний стан. Як видно із рис 4–5 за результатами ідентифікації аномального стану функціонування КС не виявлено.

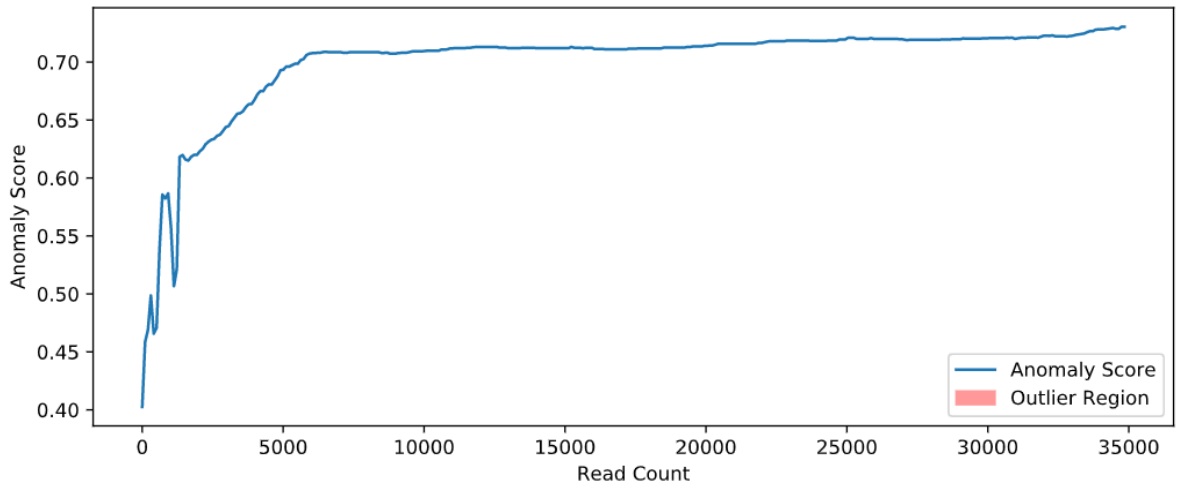

Рисунок 4 – Результат ідентифікації нормального стану КС алгоритмом IF для операції читання

DOI 10.15588/1607-3274-2021-1-11

На рис. 6–7 наведено результати ідентифікації стану функціонування КС за умови запуску великої кількості процесів, в тому числі шкідливого програмного забезпечення, що призводить до аномального стану функціонування КС. Графіки відображають залежність *AS* від значення кількості операції читання, які надалі зіставляються з назвою процесу, що дозволяє визначити ініціюючий події процес з масиву груп подій. На рис. 6 виділені дві світлі області для групи подій зі значенням кількості операцій читання (Readings) 2700–3400, які відповідають піку оцінки *AS*.

На рис. 7 піком оцінки *AS* є кількість операцій запису (Writings), яка приймає значення 50–60. За звичай, це групи подій одного аномального ініціюючого процесу.

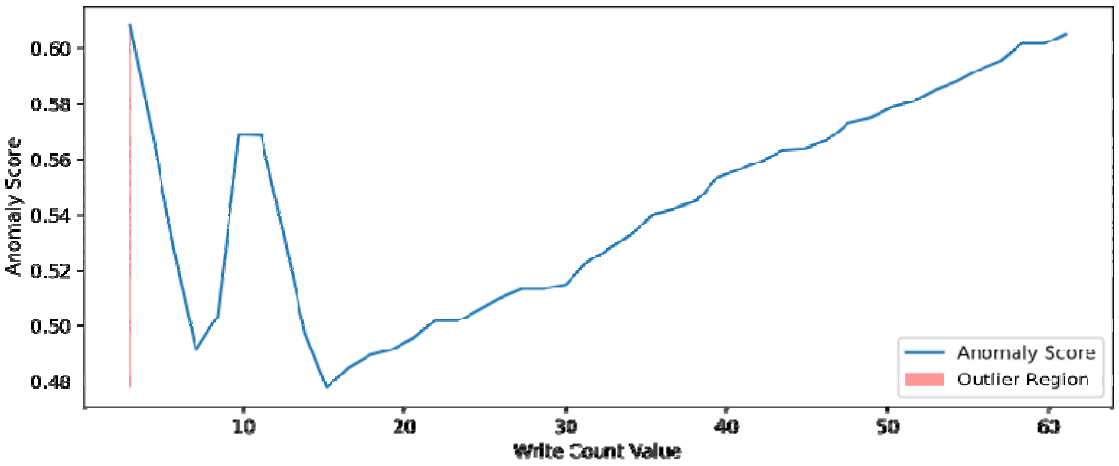

Рисунок 5 – Результат ідентифікації нормального стану КС алгоритмом IF для операції запису

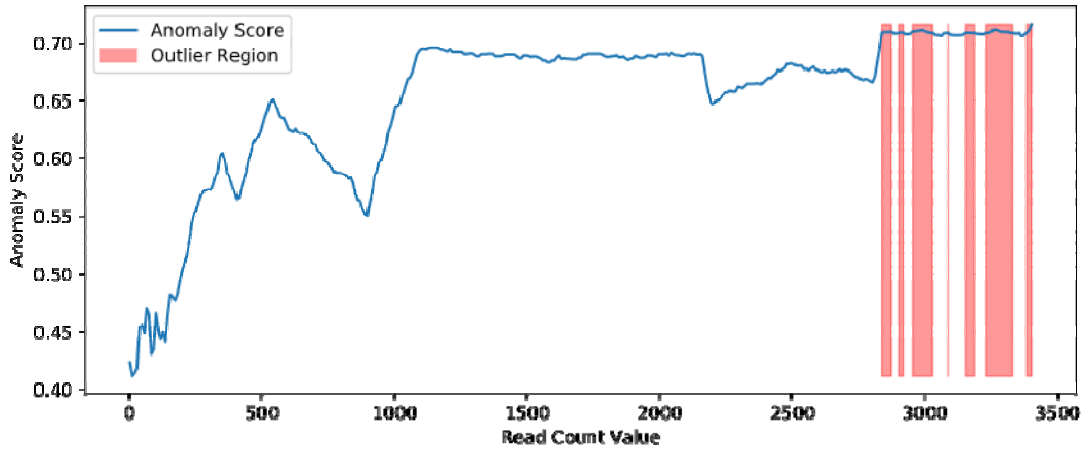

Рисунок 6 – Результат ідентифікації аномального стану КС алгоритмом IF для операції читання

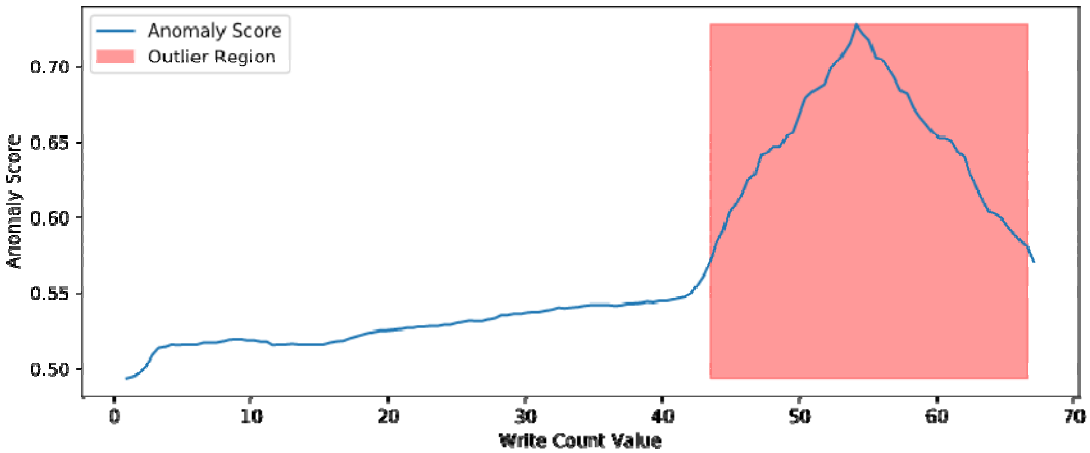

Рисунок 7 – Результат ідентифікації аномального стану КС алгоритмом IF для операції запису

DOI 10.15588/1607-3274-2021-1-11

Приклад результату роботи процедури ідентифікації процесу, який є причиною аномального стану наведено на (рис. 8). Результат містить назву аномального процесу (ProcessName), шлях до файлу виконання (ImagePath), тип операції читання (Operation_x), кількість операцій читання (ReadCount), тип операції запису (Operation_y), кількість операцій запису (WriteCount).

```
ProcessName                              7zG.exe
ImagePath          C:\Program Files\7-Zip\7zG.exe
Operation_x              QueryBasicInformationFile
ReadCount                                     17
Operation_y                             WriteFile
WriteCount                                    66
```

Рисунок 8 – Приклад результату ідентифікації

Подальші дослідження пов'язані з використанням багатовимірного аналізу системних подій, а саме одночасним використанням двох параметрів: читання і запису (змінні ReadCount і WriteCount), що дозволяє збільшити точність ідентифікації стану КС.

Для порівняння, у якості методів ідентифікації стану КС з метою виявлення аномалій було використано два алгоритми: IF та KNN.

На рис. 9–10 наведено результати ідентифікації стану КС у режимі очікування. Функціонування КС у такому режимі можливо зіставити з нормальним станом функціонування. Обидва алгоритми IF та KNN зафіксували кількість аномалій на рівні статистичної похибки (5%). Це означає, що системні події схожі між собою за параметрами запису та читання.

Для моделювання аномального стану КС виконано запуск великої кількості процесів, в тому числі архівування файлів додатком «7Zip» (процес «7zG.exe») та шкідливого програмного забезпечення (рис. 11, 12).

Обидва алгоритми IF та KNN виділили аномальні процеси. Однак, метод на основі алгоритму KNN показав меншу точність класифікації. Як видно з рис. 11 велика кількість аномальних подій, які виділено крапками, не належать виділеній аномальній області (outliers). На противагу, точність методу на основі алгоритму IF є набагато більшою (рис. 12). Метод виділив область не аномальних подій (inliers) та більш точно ідентифікував ділянки аномальних подій (outliers). Порівняльний аналіз ідентифікації стану КС, за умови запуску різних системних процесів, у вигляді проценту виявлених аномальних подій алгоритмами KNN та IF наведено в (табл. 2). Як видно із таблиці, алгоритм IF виявив більшу кількість аномалій для усіх процесів. Такі результати співпадають з результатами ідентифікації стану КС методом на основі ансамблю дерев рішень, побудованих за алгоритмом J48 [20]. Таким чином, можливо зробити висновок, що алгоритм IF є більш якісним та може бути використаним у якості складової методу ідентифікації стану комп'ютерної системи.

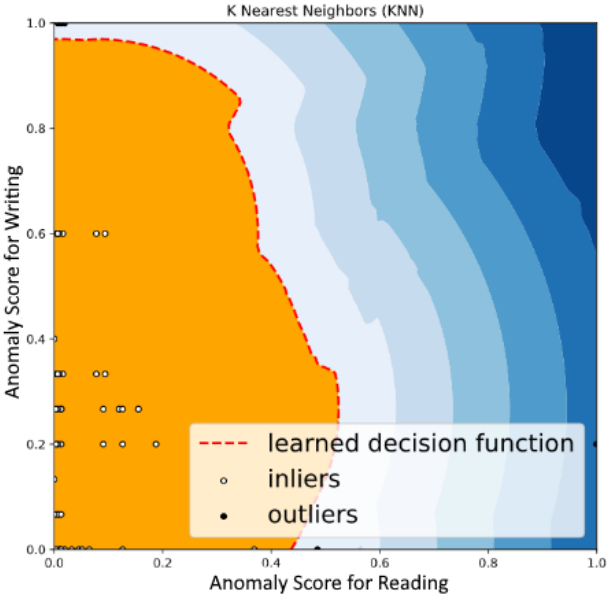

Рисунок 9 – Результати багатовимірного пошуку аномалій для системних подій методом KNN

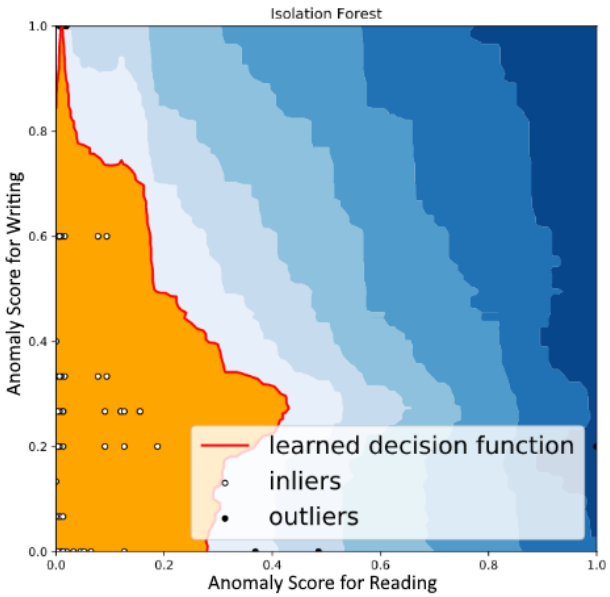

Рисунок 10 – Результати багатовимірного пошуку аномалій для системних подій методом IF

DOI 10.15588/1607-3274-2021-1-11

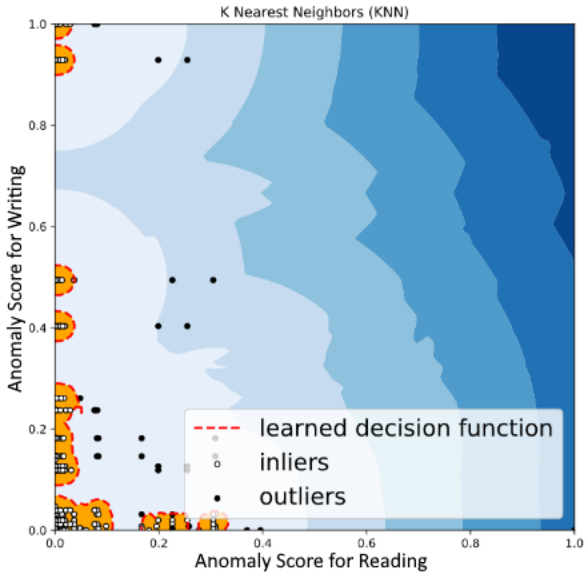

Рисунок 11 – Результати багатомірного пошуку аномалій для подій архівування «7Zip» методом KNN

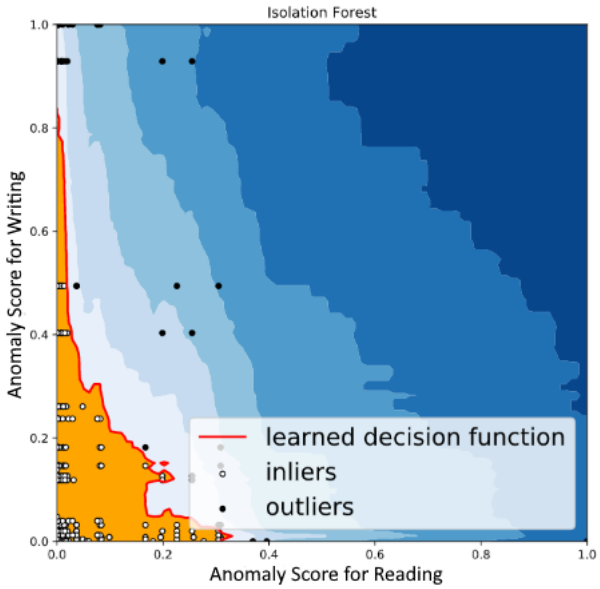

Рисунок 12 – Результати багатомірного пошуку аномалій для подій архівування «7Zip» методом IF

Таблиця 2 – Порівняльний аналіз ідентифікації стану КС, за умови запуску різних системних процесів, у вигляді проценту виявлених аномальних подій алгоритмами KNN та IF

Назва процесу	Процент виявлених аномальних подій алгоритмом IF	Процент виявлених аномальних подій алгоритмом KNN
SystemProcesses	5,00	5,00
ZipFile7Zip	8,93	8,38
ZipFileSystemZipper	8,76	7,23
OpenFoldersAndFiles	8,80	8,51
ExtractFilesSystemZipper	8,91	5,43
ExtractFiles7Zip	8,79	2,77
EditingTxtFile	8,87	4,13
DeleteToRecycleBin	43,36	38,51
DeleteFilesPerm	54,08	43,81
CopyFiles71	8,68	8,18
CopyFiles	9,09	6,85
VirusPetya	24,94	21,27

5 РЕЗУЛЬТАТИ

Якість класифікації методу на основі алгоритму IF оцінено за допомогою ROC-аналізу. Як видно із рис.13 площа під ROC AUC дорівнює 81.43%, тобто даний алгоритм є якісним. Точність класифікації складає 89.83%.

Для оцінки оперативності ідентифікації стану КС розробленим методом виконано порівняльний аналіз з методом ідентифікації на основі ансамблю дерев рішень, побудованих за алгоритмом J48 [20].

Отримано, що швидкість ідентифікації стану КС методом на основі алгоритму J48 є, в середньому, майже в 10 раз меншою відносно швидкості методом на основі алгоритму KNN та в 21 раз меншою відносно швидкості методом на основі алгоритму IF (рис. 14).

6 ОБГОВОРЕННЯ

При вирішенні завдань, пов’язаних з діагностикою та захистом комп’ютерних інформаційних ресурсів було виявлено ряд обмежень використання існуючих комп’ютеризованих систем ідентифікації стану КС. Поява аномалій, породжених вторгненнями в КС з невстановленими, або нечітко визначеними властивостям, наявність великого обсягу параметрів функціонування КС, відсутність розмічених даних призводить до труднощів з адекватного відбору показників її аномальної поведінки в умовах зовнішніх впливів і розробки критерію оцінки, що відповідає обраним показникам. Крім того, за умови відсутності розмітки даних, має місце проблема проведенням відбору моделей та перевірки якості їх роботи, що у сукупності з вищеописаними причинами

DOI 10.15588/1607-3274-2021-1-11

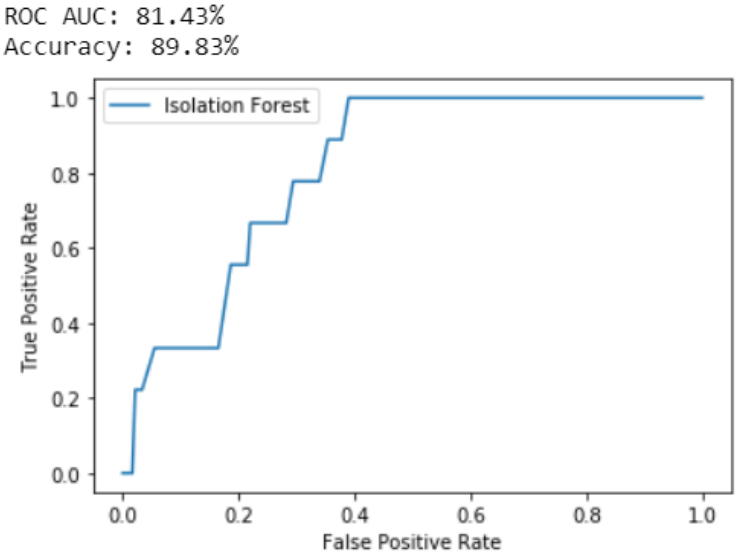

Рисунок 13 – Оцінка якості класифікації алгоритмом IF

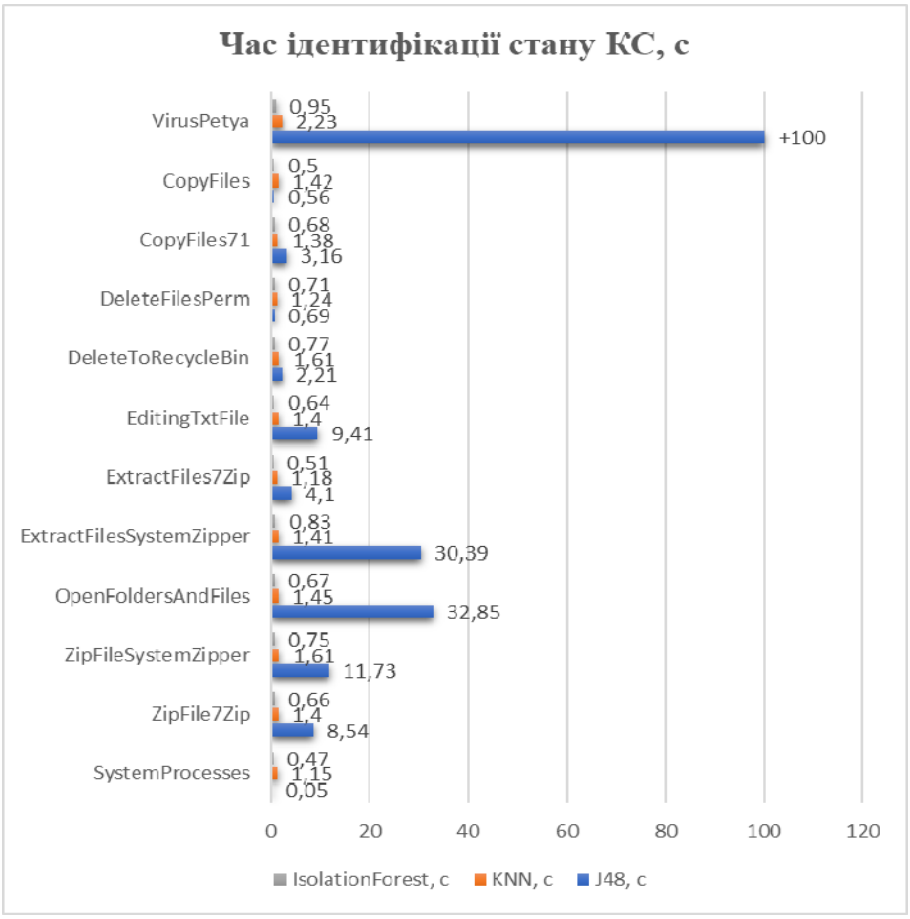

Рисунок 14 – Час ідентифікації стану КС

призводить до суттєвої розбіжності якості та слабкої практичної придатності окремих методів. Більше того, існуючі методи, в основному, тільки ідентифікують стан КС, але не надають можливість виділити назву процесу, який спричинив її аномальний стан.

Саме тому, проведені дослідження дозволили запропонувати метод ідентифікації стану КС на основі процедури групування вихідних даних та використання технології машинного навчання на основі алгоритму IF, який надає можливість не тільки ідентифікувати стан КС, але і виділити аномальні процеси. Проведені експерименти дозволили оцінити

DOI 10.15588/1607-3274-2021-1-11

точність та оперативність ідентифікації стану КС, практичну значимість та перспективи подальших досліджень.

ВИСНОВКИ

Таким чином, у роботі вирішено завдання підвищення оперативності ідентифікації стану функціонування комп'ютерної системи.

Наукова новизна отриманих результатів полягає в тому, що вперше запропоновано метод ідентифікації стану КС, який відрізняється комплексним використанням процедури групування нерозмічених вихідних даних та технології машинного навчання на основі алгоритму «Isolation Forest», що надає можливість ідентифікувати стан комп'ютерної системи і виділити назву процесу, який спричинив аномальний стан.

Для формування вихідних даних розроблено процедуру збору інформації у вигляді подій функціонування операційної системи. Виконано аналіз показників функціонування КС та оцінку їх інформативності. Аналіз подій ОС показав, що найбільш інформативними показниками її функціонування є операції читання та запису. Для формування єдиного датасету, операції читання та запису об'єднано в один масив груп подій та зіставлено з назвою процесу, що надалі надає можливість виділити назву процесу, який спричиняє аномальний стан КС.

У якості складових методу ідентифікації стану КС досліджено алгоритми: IF та KNN.

Отримано, що алгоритм IF є більш якісним та може бути використаним у якості складової методу ідентифікації стану КС.

Проведено оцінку якості запропонованого методу ідентифікації стану КС за допомогою ROC-аналізу та виконано оцінку оперативності. Отримано, що алгоритм є якісним: ROC AUC складає 81,43%, а точність виявлення аномалій складає 89,83%. При цьому, швидкість ідентифікації стану КС на основі процедури групування вихідних даних з використанням технології машинного навчання на основі алгоритму IF є, в середньому, в 21 раз вищою, ніж швидкість ідентифікації на основі ансамблю дерев рішень, побудованих за алгоритмом J48.

Практична значимість полягає в тому, що розроблений метод реалізований програмно і досліджений під час розв'язання задачі ідентифікації стану комп'ютерної системи.

Проведені експерименти підтвердили працездатність запропонованого методу, що надає можливість рекомендувати його для практичного використання у якості експрес-методу аналізу стану КС та не тільки ідентифікувати стан КС, але і виділити назву аномального процесу.

Перспективи подальших досліджень можуть полягати в розробці ансамблю нечітких дерев рішень на основі запропонованого методу, оптимізації його програмної реалізації та підвищення якості класифікації.

ПОДЯКИ

Робота виконана за підтримки міжнародного проекту Ерасмус + «Digital competence framework for Ukrainian teachers and other citizens», dComFra (598236-EPP-1-2018-1-IT-EPPKA2-CBHE-SP).

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Kelleher J. Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples and Case Studies / J. Kelleher, B. Namee, A. Archi // The MIT Pres. – 2015. – 642 p.
2. Identification of the state of an object under conditions of fuzzy input data / [S. Semenov, O. Sira, S. Gavrylenko, N. Kuchuk] // Eastern-European Journal of Enterprise Technologies. – 2019. – Vol. 1, № 4 (97). – P. 22–29. DOI: 10.15587/1729-4061.2019.157085
3. Субботін С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень / С. О. Субботін. – Запоріжжя : ЗНТУ, 2008. – 341 с.
4. Большаков А. С. Обнаружение аномалий в компьютерных сетях с использованием методов машинного обучения. Телекоммуникационные устройства и системы / А. С. Большаков, Е. В. Губанкова // Телекоммуникационные устройства и системы. – 2020. – Т. 10, № 1. – С. 37–42.
5. Линдигрин А. Н. Сравнительный анализ методов машинного обучения в задачах обнаружения сетевых аномалий / А. Н. Линдигрин // Известия Тульского государственного университета. Технические науки. – 2019. – № 12. – С. 400–404.
6. Wang S. Adapting naive Bayes tree classification / S. Wang, L. Jiang, C. Li // Knowledge and Information system. – 2015. – Vol. 44, № 1. – P. 77–89. DOI: 10.1007/s10115-014-0746-y
7. Кокорева Я. Поэтапный процесс кластерного анализа данных на основе алгоритма кластеризации k-means / Я. Кокорева, А. Макаров // Молодой ученый. – 2015. – № 13. – С. 126–128.
8. Catania C. Autonomous Labelling Approach to Support Vector Machine Algorithms for Network Traffic Anomaly Detection / Carlos Catania, Facundo Bromberg, Carlos Garcia Garino // Expert Systems lications: An International Journal Archive. – 2012. – No. 39. – P. 45–49. DOI: 10.1016/j.eswa.2011.08.068
9. Pankaj M. Long Short Term Memory Networks for Anomaly Detection in Time Series / Malhotra Pankaj// ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. – 2015. – P. 89–94.
10. Ben–Gal I. Efficient Construction of Decision Trees by the Dual Information Distance Method / Irad Ben-Gal, Alexandra Dana, Niv Shkolnik, Gonen Singer // Quality Technology & Quantitative Management. – 2014. – Vol. 11, № 1. – P. 133–147. DOI: 10.1080/16843703.2014.11673330
11. Aggarwal C. Theoretical foundations and algorithms for outlier ensembles / C. Aggarwal, S. Sathe // ACM SIGKDD Explorations Newsletter. – 2015. – Vol. 17, № 1. – P. 24–47. DOI: 10.1145/2830544.2830549
12. Zimek A. Ensembles for unsupervised outlier detection: challenges and research questions a position paper / A. Zimek, R. Campello, J. Sander // Acm Sigkdd

DOI 10.15588/1607-3274-2021-1-11

Explorations Newsletter. – 2014. – Vol. 15, № 1. – P. 11–22. DOI: 10.1145/2594473.2594476.
13. Aggarwal C. Outlier ensembles: position paper / C. Aggarwal // ACMSIGKDD Explorations Newsletter. – 2017. – Vol. 14, № 2. – P. 49–58. DOI: 10.1145/2481244.2481252
14. Rafika B. Boosted Decision Trees for Lithiasis Type Identification / Boutalbi Rafika, Chitibi Kheir Eddine // International Journal of Advanced Computer Science and Applications. – 2015. – Vol. 6, № 6. – P. 197–202. DOI: 10.14569/IJACSA.2015.060628.
15. Chandola, V. Anomaly detection: A survey / V. Chandola, A. Banerjee, V. Kumar // ACM Comput. Surv. – 2009. – № 41. – P. 15–58. DOI: 10.1145/1541880.1541882
16. Chowdhury M. Malware Analysis and Detection Using Data Mining and Machine Learning Classification / M. Chowdhury, A. Rahman., Rz. Islam // International Conference on Applications and Techniques in Cyber Security and Intelligence. – 2018. – P. 266–274.
17. Breiman L. Random Forests / L. Breiman // Statistics Department University of California Berkeley: Machine Language. – 2001. – P. 5–32.
18. Шелухин О. И. Применение алгоритма «изолирующий лес» для решения задач обнаружения аномалий / О. И. Шелухин, М. В. Полковников // Решение. – 2019. – С. 186–188.
19. Fei Tony L. Isolation forest / Liu Fei Tony, Ting Kai Ming, Zhou Zhi-Hua // Proceedings of the 2008 Eighth IEEE International Conference on Data Mining. – 2008. – P. 413–422. DOI: 10.1109/ICDM.2008.17
20. Gavrylenko S. The ensemble method development of classification of the computer system state based on decisions trees / S. Gavrylenko, I. Sheverdin, M. Kazarinov // Advanced Information Systems. – 2020. – P. 5–10. DOI: 10.20998/2522-9052.2020.3.01

Стаття надійшла до редакції 12.10.2020.
Після доробки 27.12.2020.

УДК 004.8

РАЗРАБОТКА МЕТОДА ИДЕНТИФИКАЦИИ СОСТОЯНИЯ КОМПЬЮТЕРНОЙ СИСТЕМЫ НА ОСНОВЕ АЛГОРИТМА «ISOLATION FOREST»

Гавриленко С. Ю. – д-р техн. наук, доцент, профессор кафедры «Вычислительная техника и программирование», Национальный технический университет «Харьковский политехнический институт», Харьков, Украина.

Шевердин И. В. – аспирант кафедры «Вычислительная техника и программирование», Национальный технический университет «Харьковский политехнический институт», Харьков, Украина.

АНОТАЦІЯ

Актуальность. Рассмотрена задача идентификации состояния компьютерной системы. Объектом исследования является процесс идентификации состояния компьютерной системы. Предметом исследования являются методы и средства идентификации состояния компьютерной системы.

Цель. Целью работы является разработка метода идентификации состояния компьютерной системы.

Метод. Разработан метод идентификации состояния компьютерной системы на основе комплексного использования процедуры группировки неразмеченных исходных данных и технологии машинного обучения на основе алгоритма «Isolation Forest», который предоставляет возможность идентифицировать состояние компьютерной системы и выделить название процесса, который вызвал аномальный состояние. Для этого предложена процедура и разработано программное приложение для сбора статистических данных в виде событий функционирования операционной системы и выполнен их анализ. Получено, что наиболее информативными являются операции чтения и записи. Для формирования единого датасета, операции чтения и записи сопоставлены с названием процесса и объединены в один массив групп событий, что в дальнейшем позволяет выделить процесс, который вызывает аномальный состояние компьютерной системы. По результатам исследования, в качестве составляющей метода идентификации состояния компьютерной системы использовано ансамблевый алгоритм «Isolation Forest». Проведена оценка точности и оперативности разработанного метода идентификации состояния компьютерной системы.

Результаты. Разработанный метод реализован программно и исследован при решении задачи идентификации аномалий функционирования компьютерной системы.

Выводы. Проведенные эксперименты подтвердили работоспособность предложенного метода, позволяет рекомендовать его для практического использования с целью повышения оперативности идентификации состояния компьютерной системы и использования его в качестве экспресс-метода. Перспективы дальнейших исследований могут заключаться в разработке ансамбля нечетких деревьев решений на основе предложенного метода, оптимизации его программных реализации.

КЛЮЧЕВЫЕ СЛОВА: компьютерная система, события операционной системы, аномальный состояние, идентификация, машинное обучение, алгоритм «Isolation Forest».

UDC 004.8

DEVELOPMENT OF METHOD TO IDENTIFY THE COMPUTER SYSTEM STATE BASED ON THE «ISOLATION FOREST» ALGORITHM

Gavrylenko S. Y. – Dr. Sc., Associate Professor, Professor at Department of Computer Engineering and Programming, National Technical University «Kharkiv Polytechnik Institute», Kharkiv, Ukraine.

Sheverdin I. V. – Post-graduate student at Department of Computer Engineering and Programming, National Technical University «Kharkiv Polytechnik Institute», Kharkiv, Ukraine.

DOI 10.15588/1607-3274-2021-1-11

ABSTRACT

Context. The problem of identification a computer system state was investigated. The object of the research is the identification process of the computer system state. The subject of the research is computer system state identifying means and methods.

Objective. The purpose of the work is to develop a method for identifying the computer system state.

Method. The method has been developed for identifying a computer system state based on integrated use the procedure for grouping unlabeled initial data and using machine learning technology based on the «Isolation Forest» algorithm, which provides to identify a computer system state and to distinguished the process name that initiated the abnormal state. Therefore, for collecting statistical data in the form of operating system functioning events, data method has been proposed and developed along with software. The analysis of functioning events has been performed. The result of analysis showed that the most informative are read and write operations. To set up a single dataset, read and write operations compared with the process name and combined into one array of event groups, so that it is possible to single out the process that causes the abnormal state of the computer system. As a result of the research, the «Isolation Forest» algorithm has been selected as a component of the method for identifying the computer system state. An accuracy and efficiency assessment of the developed method of identifying a computer system state has been carried out.

Results. The developed method is implemented and investigated when solving the problem of identifying anomalies in the functioning of computer systems.

Conclusions. The experiments carried out confirmed the efficiency of the proposed method. It allows us recommended the method for practical use in order to improve efficiency of identifying the computer system state and use it as an express method. Areas for further research may lie in the creation of the ensemble of fuzzy trees based on the proposed method and optimization of this software implementation.

KEYWORDS: computer system, operating system events, abnormal state, identification, machine learning, Isolation Forest algorithm.

REFERENCES

1. Kelleher, J., B. Namee, A. Archi Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies, *The MIT Pres*, 2015, 642 p.
2. Gavrylenko S., Semenov S., Sira O., Kuchuk N. Identification of the state of an object under conditions of fuzzy input data. *Eastern-European Journal of Enterprise Technologies*, 2019, Vol. 1, No. 4 (97), pp. 22–29. DOI: 10.15587/1729-4061.2019.157085
3. Subbotin S.O. Podannya j obrobka znan u sistemah shtuchnogo intelektu ta pidtrimki prijnyattya rishen. Zaporizhzhya, ZNTU, 2008, 341 p.
4. Bolshakov A.S., Gubankova E.V. Obnaruzhenie anomalij v kompyuternyh setyah s ispolzovaniem metodov mashinnogo obucheniya. *Telekommunikacionnye ustrojstva i sistemy*, 2020, Vol. 10, No. 1, pp. 37–42.
5. Lindigrin A. N. Sravnitelnyj analiz metodov mashinnogo obucheniya v zadachah obnaruzheniya setevyh anomalij, *Izvestiya Tulskogo gosudarstvennogo universiteta. Tehnicheskie nauki*, 2019, No. 12, pp. 400–404.
6. Wang S., Jiang L., Li C. Adapting naive Bayes tree classification, *Knowledge and Information system*, Vol. 44, No. 1, pp. 77–89. DOI: 10.1007/s10115-014-0746-y
7. Kokoreva Ya., Makarov A. Poetapnyj process klasternogo analiza dannyh na osnove algoritma klasterizacii k-means, *Molodoj uchenyj*, 2015, No. 13, pp. 126–128.
8. Carlos A., Catania, Facundo Bromberg, Carlos Garcia Garino. An Autonomous Labelling Approach to Support Vector Machine Algorithms for Network Traffic Anomaly Detection, *Expert Systems lications: An International Journal Archive*, 2012, No. 39, pp. 45–49. DOI: 10.1016/j.eswa.2011.08.068
9. Malhotra Pankaj, Long Short Term Memory Networks for Anomaly Detection inTime Series, *ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2015.
10. Irad Ben-Gal, Alexandra Dana, Niv Shkolnik, Gonen Singer. Efficient Construction of Decision Trees by the Dual Information Distance Method, *Quality Technology & Quantitative Management*, 2014, Vol. 11, No. 1, pp. 133–147. DOI: 10.1080/16843703 .2014.11673330
11. Aggarwal C. C., Sathe S. Theoretical foundations and algorithms for outlier ensembles, *ACM SIGKDD Explorations Newsletter*, 2015, Vol. 17, No. 1, pp. 24–47. DOI: 10.1145/2830544.2830549
12. Zimek A., Campello R. J. G. B., Sander J. Ensembles for unsupervised outlier detection: challenges and research questions a position paper, *Acm Sigkdd Explorations Newsletter*, 2014, Vol. 15, No. 1, pp. 11–22. DOI: 10.1145/2594473.2594476
13. Aggarwal C. C. Outlier ensembles: position paper, *ACMSIGKDD Explorations Newsletter*, 2017, Vol. 14, No. 2, pp. 49–58. DOI: 10.1145/2481244.2481252
14. Boutalbi Rafika, Chitibi Kheir Eddine. Boosted Decision Trees for Lithiasis Type Identification, *International Journal of Advanced Computer Science and Applications*, 2015, Vol. 6, No. 6, pp. 197–202.
15. Chandola V., Banerjee A., Kumar V. Anomaly detection:survey, *ACM computing surveys (CSUR)*, 2009, No. 41, pp. 15–58. DOI: 10.1145/1541880.1541882.
16. Chowdhury M. Malware Analysis and Detection Using Data Mining and Machine Learning Classification, *International Conference on Applications and Techniques in Cyber Security and Intelligence, ATCI*, 2018, pp. 266–274.
17. Breiman, L. Random Forests, *Machine Language*, 2001, No. 45 (1), pp. 5–32.
18. Sheluhin O. I., Polkovnikov M. V. Primenenie algoritma «izoliruyushij les» dlya resheniya zadach obnaruzheniya anomalij. *Reshenie,* 2019, No. 1, pp. 186–18.
19. Liu, Fei Tony, Ting, Kai Ming and Zhou, Zhi-Hua. Isolation forest, *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, December 2008, pp. 413–422. DOI: 10.1109/ICDM.2008.17
20. Gavrylenko S., Sheverdin I., Kazarinov M. The ensemble method development of classification of the computer system state based on decisions trees, *Advanced Information System*, 2020, Vol. 4, No. 2, pp. 5–10. DOI: 10.20998/2522-9052.2020.3.01

DOI 10.15588/1607-3274-2021-1-11