УДК 004.732.056 **doi: 10.26906/SUNZ.2025.4.144-150**

В. В. Челак, О. А. Горносталь

Національний технічний університет “Харківський політехнічний інститут”, Харків, Україна

НЕЧІТКИЙ АНСАМБЛЬ ДЕРЕВ РІШЕНЬ ДЛЯ ІДЕНТИФІКАЦІЇ СТАНУ КОМП’ЮТЕРНИХ СИСТЕМ

Анотація. Об'єктом дослідження є процес ідентифікації стану комп’ютерних систем. **Предметом дослідження** є методи нечіткого ансамблю дерев з багатовимірними вузлами рішень для ідентифікації стану комп’ютерних систем. **Метою дослідження** є розробка та оцінка ефективності нечіткого ансамблю дерев рішень для підвищення точності ідентифікації стану комп’ютерних систем за умов наявності невизначеності, шумових впливів та неповних даних. **Методи, що використовуються**: методи машинного навчання, методи попередньої обробки даних, ансамблеві класифікатори, стекінгові підходи, методи комбінування та вибору ознак КС. **Отримані результати**: досліджено ефективність класичних та розроблених методів для ідентифікації стану комп’ютерних систем у складних умовах, що включають дисбаланс даних та наявність аномальних станів. Запропоновано комплексний підхід із використанням Fuzzy Stacking з MDT, що забезпечує високу точність і стабільність класифікації. Найкращі результати отримано саме для стекінгового підходу, який поєднує базові класифікатори та нечіткі дерева рішень, дозволяючи мінімізувати помилки першого та другого роду та досягти високих показників узагальнюючої здатності (MCC, F1-score, TS, LN(DOR)). **Висновки**. За результатами дослідження запропоновано удосконалений підхід до ідентифікації стану комп’ютерних систем, який поєднує стекінговий метод із Fuzzy MDT та оптимізацію вибору ознак. Комплексне використання цих методів дозволяє значно покращити точність класифікації, стабільність результатів та стійкість моделей до дисбалансу даних, а також забезпечує високу якість класифікації навіть у випадках появи нових аномальних станів.

Ключові слова: ідентифікація стану, комп’ютерні системи, машинне навчання, ансамбль, стекінг, дерева рішень, нечітка логіка.

Вступ

Сучасні комп’ютерні системи є складними багаторівневими утвореннями, що інтегрують програмні, апаратні та мережеві компоненти. Вони забезпечують критично важливі функції в промисловості, транспорті, енергетиці, фінансовій та державній сферах. В умовах зростання складності їхньої структури та взаємозв’язків підвищується ймовірність виникнення збоїв, нештатних режимів роботи та кібератак, що можуть призвести до порушення стабільності або повного припинення функціонування системи. Тому завдання своєчасної ідентифікації стану комп’ю-терних систем набуває особливої актуальності.

Традиційні методи діагностики та моніторингу, що базуються на жорстких правилах і статистичних моделях, не завжди здатні адекватно реагувати на динамічні та невизначені зміни параметрів. Вони часто не враховують багатофакторний характер впливу різних процесів та складність взаємозв’язків між параметрами, що описують стан системи. У зв’язку з цим зростає інтерес до використання інтелектуальних методів аналізу даних, зокрема машинного навчання, нечіткої логіки та ансамблевих моделей, які дозволяють підвищити точність і надійність оцінювання станів у складних і невизначених умовах. Дерева рішень є одними з найбільш інтерпретованих і зручних моделей для аналізу стану систем, однак їх окреме застосування часто призводить до перенавчання або зниження узагальнюючої здатності. Ефективним підходом до подолання цих недоліків є використання ансамблевих методів, які об’єднують результати кількох моделей. Водночас нечітка інтерпретація результатів дозволяє зменшити вплив похибок вимірювання та неповноти даних, забезпечуючи більш гнучке прийняття рішень.

Таким чином, актуальним є розробка підходу, який би поєднував переваги ансамблевих методів машинного навчання з можливістю нечіткої оцінки станів комп’ютерних систем. Це дозволить підвищити достовірність і стійкість ідентифікації при наявності шумових або суперечливих даних.

Об’єктом дослідження є процес ідентифікації стану комп’ютерних систем.

Предметом дослідження є методи нечіткого ансамблю дерев рішень для ідентифікації стану комп’ютерних систем.

Огляд пов'язаних наукових публікацій. Ідентифікація стану комп’ютерних систем є одним з сучасних напрямків досліджень у сфері забезпечення їхньої надійності та безпеки. Сучасні методи контролю стану базуються на аналізі великої кількості технічних параметрів, що описують роботу компонентів системи, їхню взаємодію та реакцію на зовнішні впливи. Традиційні підходи до моніторингу, зокрема на основі експертних правил або порогових значень [1, 2], забезпечують швидке реагування на відхилення, однак характеризуються низькою гнучкістю та не здатні ефективно працювати у випадках, коли параметри змінюються під впливом стохастичних процесів або неповноти даних.

Для подолання цих обмежень активно застосовуються методи машинного навчання, які дозволяють виявляти приховані залежності між параметрами та формувати адаптивні моделі поведінки системи [3, 4]. Найпоширенішими серед них є алгоритми класифікації, зокрема дерева рішень, опорні вектори (SVM), штучні нейронні мережі та ансамблеві методи, такі як Random Forest або Gradient Boosting [5, 6]. Дерева рішень залишаються популярними завдяки їхній простоті, інтерпретованості та здатності працювати з гетерогенними даними. Проте окремі дерева часто мають проблеми з перенавчанням, особливо у випадку високої кореляції ознак або значної варіативності вхідних даних.

Розвиток ансамблевих методів став природним етапом еволюції алгоритмів машинного навчання. Такі моделі поєднують результати кількох базових класифікаторів, що дозволяє підвищити стійкість та точність прийняття рішень. Методи типу Random Forest [7] або XGBoost [8] демонструють високу ефективність у задачах прогнозування технічних станів, проте вони залишаються чутливими до шуму та потребують чітких меж між класами. У реальних комп'ютерних системах, де наявна невизначеність і нечіткість даних, такі межі часто відсутні або змінюються динамічно. У зв'язку з цим перспективним напрямом досліджень є інтеграція апарату нечіткої логіки в ансамблеві методи машинного навчання [9, 10]. Нечіткі системи дозволяють описувати невизначеність у вигляді функцій належності, що дає змогу моделювати проміжні стани між «нормальним» та «аномальним» режимом. Поєднання дерев рішень із нечіткою логікою забезпечує більш гнучку інтерпретацію результатів класифікації, а створення нечітких ансамблів підвищує їхню стійкість до неповних або суперечливих даних.

У роботах [11–13] показано, що використання нечітких ансамблів дозволяє досягати кращої узагальнюючої здатності моделей у порівнянні з класичними ансамблями за рахунок вагового врахування ступеня достовірності окремих класифікаторів. Такі підходи особливо ефективні в умовах наявності шуму в телеметричних даних, апаратних збоях та інших невизначених станах. Однак більшість існуючих рішень фокусуються лише на певному типі даних або окремих характеристиках системи, не забезпечуючи комплексного підходу до оцінювання станів.

Отже, аналіз наукових джерел свідчить про необхідність створення узагальненого підходу, що поєднує різні методи ідентифікації станів комп'ютерних систем у межах єдиної нечіткої ансамблевої структури. Це дозволить врахувати специфіку кожного з методів, зменшити вплив шуму та підвищити точність класифікації при роботі в умовах невизначеності.

Постановка проблеми. Основна мета дослідження полягає в розробки та оцінці ефективності нечіткого ансамблю дерев рішень для підвищення точності ідентифікації стану комп'ютерних систем за умов наявності невизначеності, шумових впливів та неповних даних. Проблема ускладняється тим, що сучасні комп'ютерні системи характеризуються високою динамічністю, складною взаємодією компонентів та значною кількістю параметрів, які не завжди мають чітко визначені межі між нормальним та аномальним станом. У зв'язку з цим постає потреба в інтеграції апарату нечіткої логіки в ансамблеві структури, що дозволить здійснювати гнучке зважування результатів окремих класифікаторів та формувати узагальнене рішення з урахуванням ступеня впевненості кожного з них. Об'єднання декількох різних методів класифікації в єдиний нечіткий ансамбль забезпечить більш високу стійкість до шумів, здатність до адаптації в умовах неповних або суперечливих даних та покращення точності розпізнавання складних станів.

Для реалізації поставленої мети необхідно дослідити такі напрями підвищення ефективності систем ідентифікації:

1. Аналіз наявних методів ідентифікації стану комп'ютерних систем і визначення їхніх сильних та слабких сторін.
2. Розробка структури нечіткого ансамблю дерев рішень, що поєднує переваги різних підходів до класифікації.
3. Визначення механізму формування функцій належності та вагових коефіцієнтів для агрегування рішень окремих класифікаторів.
4. Проведення експериментальної оцінки точності запропонованого ансамблю на реальних або модельних наборах даних, що описують різні стани комп'ютерних систем.
5. Порівняння отриманих результатів із класичними ансамблевими методами без використання нечіткої логіки для оцінки доцільності впровадження запропонованого підходу.

Виконання цих етапів дозволить сформувати науково обґрунтовані висновки щодо доцільності застосування нечітких ансамблевих моделей у задачах ідентифікації стану комп'ютерних систем та розробити рекомендації для їх подальшої оптимізації й практичного використання.

Огляд підходів та методів

В основі запропонованого методу лежить алгоритм побудови дерев з багатовимірними вузлами рішень, які були вперше представлені в [14]. Теоретична основа дозволяє створювати вузли, які зможуть описувати багатовимірної області будь-якими фігурами та формами. На рис. 1, 2 представлені приклади двовимірних та тривимірних вузлів рішень відповідно.

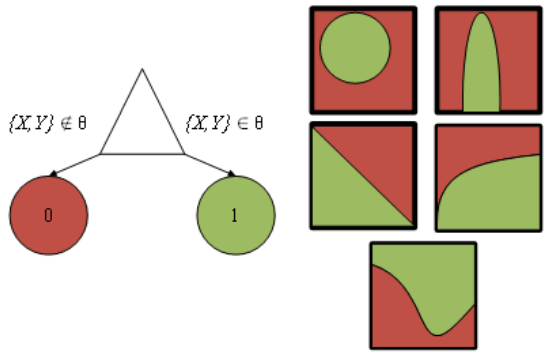

Рис. 1. Двовимірний вузол рішень з різними варіаціями фігури θ

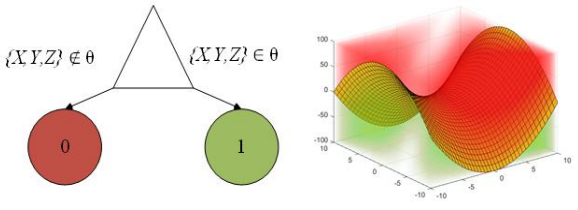

Рис. 2. Тривимірний вузол рішень з заданою фігурою θ

В поточній реалізації алгоритму використовується обмеження, щодо форми розподілу фігур, а саме використання гіперсфер: задаються координати центру (потрібної розмірності) та радіуса.

Першою частиною є побудова базових класифікаторів. Розглянемо модифіковану версію алгорит-

му побудови дерев з одновимірними вузлами рішень, на базі яких будуються багатовимірні вузли (при наявності покращення результатів):

Крок 1. Сформувати тренувальну вибірку даних.

Крок 2. Для кожної ознаки з найвищою інформативністю A_i:

2.1. Визначити мінімальне та максимальне значення ознаки.

2.2. Розрахувати середнє значення між ними та встановити його як поточний поріг θ_i.

2.3. Обчислити значення функції помилки класифікації $E(A)$ для лівої та правої підмножини:

$$E(A) = \sum_{i=1}^{N} [y_i \neq t_i]'; \quad (1)$$

$$[y_i \neq t_i]' = \begin{cases} 1.5, & y_i \neq t_i \ \cap \ t_i = 1; \\ 1, & y_i \neq t_i \ \cap \ t_i = 0; \\ 0, & y_i = t_i. \end{cases} \quad (2)$$

2.4. Якщо $E(A)$ для лівої підмножини більше, ніж для правої, встановити поточний поріг як нове максимальне значення; інакше – як нове мінімальне.

2.5. Повторювати бінарний пошук порогу θ_i, доки $E(A)$ не перестане змінюватися.

У якості оптимального порогу θ_i приймається значення, при якому $E(A)$ мінімальне.

Крок 3. Створити вузол дерева з пороговим значенням θ_i, сформувати підмножини зразків для лівої та правої гілки.

Крок 4. Перевірити умови зупинки (досягнута максимальна глибина, мінімальна кількість зразків тощо). Якщо хоча б одна виконується – перейти до кроку 6.

Крок 5. Повторити кроки 2–3 для тієї гілки, у якої значення $E(A)$ є більшим.

Якщо обидві гілки мають однакову помилку, пріоритет надається гілці з більшою кількістю екземплярів. Якщо $E(A)$ мінімальне для обох гілок, перейти до вузла вищого рівня та повторити крок 5.

Крок 6. Завершити побудову дерева рішень.

Алгоритм базується на ітераційному поділі вибірки даних за допомогою порогових значень, що визначаються методом бінарного пошуку. На кожному етапі обирається ознака з найвищою інформативністю, а поріг розділення θ_i підбирається так, щоб мінімізувати функцію помилки класифікації $E(A)$. Таким чином, формується вузол дерева, який поділяє простір ознак на дві області. Подальше розгалуження здійснюється рекурсивно для тієї частини, де залишкова помилка є найбільшою. Процес продовжується до виконання умов зупинки – досягнення заданої глибини дерева або мінімальної похибки.

Такий підхід дозволяє створювати адаптивну структуру дерева, що забезпечує гнучке розділення простору ознак та високу точність класифікації навіть при наявності неоднорідних даних.

Для пошуку гіперсфер, що зможуть об'єднувати велику кількість зразків з мінімальною помилкою використовується **алгоритм кластерізації DBSCAN**.

В основі алгоритму покладена ідея, згідно з якою всередині кожного кластера щільність об'єктів є суттєво вищою, ніж щільність зовні кластера, тоді як у шумових областях щільність нижча за щільність будь-якого з кластерів.

Першим кроком алгоритму є побудова матриці відстаней за формулою квадрата евклідової відстані:

$$d(x_i, x_j) = \sum_{k=1}^{m} (x_{ik} - x_{jk})^2, \quad (3)$$

де $d(x_i,x_j)$ – функція відстані між двома зразками; x_{ik} – значення k-ої ознаки i-го зразка; m – кількість ознак.

Для випадків, коли значення ознак є дискретними, використовується мангеттенська метрика:

$$d(x_i, x_j) = \sum_{k=1}^{m} |x_{ik} - x_{jk}|, \quad (4)$$

На наступному етапі визначається, чи є об'єкт x_i сусідом для об'єкта x_j. Результатом цього кроку є побудова матриці сусідства, яка базується на матриці відстаней:

$$Neighbour_{ij} = \begin{cases} 0, & d(x_i, x_j) > \varepsilon; \\ 1, & d(x_i, x_j) \leq \varepsilon; \end{cases} \quad (5)$$
$$i = \overline{1 \dots N}, j = \overline{1 \dots N},$$

де ε – гіперпараметр, що визначає радіус гіперсфери у багатовимірному просторі ознак.

Далі виконується ініціалізація міток для кожного об'єкта:

$$Label_i = -1, i = \overline{1 \dots N}. \quad (6)$$

На основі матриці сусідства формується початковий набір кластерів. Усі об'єкти на початку вважаються невизначеними. Ітераційна процедура починається з довільного об'єкта x_i, який ще не був розглянутий. Для кожного поточного об'єкта формується список сусідів – усі x_j, для яких у матриці $Neighbour_{ij}$=1. Підраховується кількість сусідів K і порівнюється з порогом $MinPts$. Якщо $K < MinPts$, об'єкт позначається як можливий викид. Якщо $K \geq MinPts$, об'єкт вважається ядровим, а його сусіди – досяжними за щільністю. Поточний об'єкт x_i разом із сусідами формують новий кластер, що отримує унікальну мітку.

Процес розширення кластера продовжується ітераційно: перевіряються необроблені або позначені як можливі викиди об'єкти, які є досяжними для елементів кластера, і приєднуються до нього. Ітерації тривають, доки кластер не перестане розширюватися. Процедура повторюється, поки всі об'єкти не набудуть визначеного статусу. Об'єкти, що залишились поза кластерами, вважаються викидами.

Після завершення кластеризації виконується **пошук параметра прийняття рішення для багатовимірного вузла дерева рішень**. Для цього визначається кластер C з максимальною кількістю елементів:

$$C = \max_{A_i \in A} (|A_i|), \quad (7)$$

де $|A_i|$ – кількість елементів i-го кластера.

Далі обчислюється центр кластера за кожною ознакою:

$$xc_k = \sum_{i=1}^{|C|} x_{ik} \Big/ |C|. \quad (8)$$

Отримавши координати центру кластера, виконується розрахунок відстаней від кожної точки до центру, після чого визначається максимальне значення:

$$\varepsilon = \max_{x_i \in C}\big(d(xc, x_i)\big). \quad (9)$$

Значення ε інтерпретується як радіус гіперсфери для багатовимірного вузла дерева рішень. Остаточним етапом є розрахунок цільових функцій, таких як критерій Джині або функції помилки класифікації. Як результат обираються такі гілки дерева, для яких сумарне значення цільової функції є мінімальним.

Другою частиною є використання нечітких дерев в якості мета-моделі, яка буде приймати загальне рішення стекінг ансамбля.

Нечітке дерево використовується як мета-модель для прийняття інтегрального рішення стекінг-ансамблю. Його побудова базується на фазифікації вхідних даних та аналізі інформаційного приросту для вибору оптимальних ознак розщеплення.

Процес фазифікації відбувається по спеціальній процедурі, яка була розроблена в попередніх дослідженнях [9]. На початку формується база правил та виконується фазифікація навчальної вибірки, у результаті якої отримується тривимірна таблиця нечітких значень $F_{ijk} = \mu_{jk}(TSij)$, де $\mu_{jk}(x)$ – функції належності лінгвістичних змінних. Далі розраховуються суми очікуваних та інверсних результатів і визначається загальна ентропія системи $E(F)$. Для кожної ознаки j, що ще не використана у дереві, обчислюються часткові ентропії нечітких множин E_{jk} та ентропія ознаки $E(j)$. На основі цього визначається інформаційний приріст $G(j) = E(F) - E(j)$.

Ознака з максимальним значенням $G(j)$ обирається як розщеплення вузла дерева. Для кожної гілки створюються підвузли відповідно до кількості функцій належності обраної ознаки. Агрегація ступенів належності між поточним вузлом та підвузлами виконується за мінімальним оператором. Отримані значення використовуються для подальших ітерацій побудови.

Процес рекурсивно повторюється, доки всі ознаки не будуть використані або інформаційний приріст стане незначним. У фіналі зберігаються параметри моделі: коефіцієнти гілок, вибрані ознаки розщеплення, структура вузлів та параметри функцій належності. У результаті формується модель FuzzyDecisionTree, що забезпечує адаптивне прийняття рішень на основі нечітких правил і використовується як узагальнювальний рівень ансамблю класифікаторів.

Формування нечіткого ансамблю дерев рішень

Для підвищення стійкості класифікації та узагальнюючи здатності моделей запропоновано об'єднання декількох різнотипних алгоритмів у стекінг-ансамбль, що поєднує властивості класичних дерев рішень, дерев з багатовимірними вузлами та нечітких дерев рішень. Такий підхід дозволяє врахувати як лінійні, так і нелінійні взаємозв'язки між ознаками, а також забезпечити нечітку інтерпретацію рішень у випадках, коли дані мають невизначеність або накладання класів.

Етап 1. Формування базових моделей

Першим рівнем ансамблю є набір базових моделей M_i, $i = \overline{1, L}$, серед яких:

1. Класичні дерева рішень M_{DT};
2. Дерева з багатовимірними вузлами M_{MDT};

Кожна з моделей навчається на однаковій навчальній вибірці $TS = \{(x_i, y_i)\}$, де $x_i \in \mathbb{R}^m$ – вектор ознак, а $y_i \in \{\overline{1, K}\}$ – мітка класу.

Результатом навчання базових моделей є набір прогнозів:

$$Z_i = [M_1(x_i), M_2(x_i), \dots, M_L(x_i)]. \quad (10)$$

Цей набір утворює матрицю вихідних оцінок базових моделей $Z \in \mathbb{R}^{N \times L}$.

Етап 2. Побудова мета-рівня нечіткого узагальнення

Отримані вихідні значення Z подаються на вхід нечіткому дереву рішень (Fuzzy Decision Tree, FDT), яке виконує роль мета-класифікатора.

Для кожної з базових моделей формується відповідна лінгвістична змінна $\tilde{Z}_j$ з функціями належності $\mu_{jk}(z_j)$, які визначають ступінь впевненості моделі у своєму прогнозі («низька», «середня», «висока»).

Фазифікація векторів Z_i здійснюється аналогічно до попереднього алгоритму нечіткого дерева, що дозволяє побудувати тривимірну таблицю нечітких оцінок:

$$F_{ijk} = \mu_{jk}(Z_{ij}), i = \overline{1, N},\ j = \overline{1, L},\ k = \overline{1, S_j}. \quad (11)$$

На основі фазифікованих значень виконується побудова нечіткого дерева, яке визначає узагальнене рішення ансамблю:

$$\hat{y}_i = FDT(Z_i). \quad (12)$$

Таким чином, кожен об'єкт оцінюється через систему правил виду:

$$IF\ (Z_1 \text{is "High"}) \cap \\ \cap\ (Z_2 \text{ is "Middle"}) then y_i is \text{Anomaly}. \quad (13)$$

Етап 3. Агрегація рішень та адаптація ваг

Після первинного навчання ансамблю виконується оцінка впливу кожної базової моделі на кінцеве рішення. Для цього вводиться коефіцієнт ваги w_j, який визначає значущість моделі M_j у загальній структурі ансамблю:

$$w_i = 1/\big(E_j + \delta\big). \quad (14)$$

де E_j – середнє значення функції помилки для моделі M_j, δ – мала константа для уникнення ділення на нуль.

Підсумкове значення агрегованого рішення до фазифікації розраховується за зваженою сумою:

$$S_i = \frac{\sum_{j=1}^{L} w_j \cdot M_j(x_i)}{\sum_{j=1}^{L} w_j}. \quad (15)$$

Ці результати використовуються для побудови нечітких правил мета-рівня, що дозволяє FDT враховувати як індивідуальну впевненість моделей, так і їх взаємні суперечності.

Етап 4. Остаточне прийняття рішення

Після фазифікації агрегованих значень S_i нечітке дерево виконує дефазифікацію результату –визначає кінцевий клас:

$$\hat{y}_i = \text{Defuzzify}(S_i), \quad (16)$$

де може бути використано, наприклад, метод центру ваги:

$$\hat{y}_i = \frac{\int_\Omega y \cdot \mu(y)\, dy}{\int_\Omega \mu(y)\, dy}. \quad (17)$$

Отримане значення $\hat{y}_i$ є остаточним рішенням нечіткого ансамблю, що поєднує узагальнену інформацію від усіх базових класифікаторів.

Формулювання теоретичних очікувань

Очікується, що використання дерев рішень із багатовимірними вузлами дозволить покращити якість класифікації завдяки здатності таких моделей описувати складні, нелінійні кордони між класами у багатовимірному просторі ознак. На відміну від класичних дерев, які розділяють простір за одновимірними порогами, багатовимірні вузли здатні формувати області будь-якої форми (зокрема гіперсферичні), що підвищує здатність моделі до узагальнення, особливо при обробці даних з корельованими або взаємозалежними ознаками.

Впровадження ансамблю на основі стекінгу (stacking) дозволяє комбінувати результати кількох різнорідних базових моделей, отриманих за допомогою різних параметрів або початкових умов побудови дерев. Такий підхід має забезпечити підвищення стійкості моделі до випадкових коливань у навчальних даних, а також зменшення варіації результатів.

Згідно з теоретичними очікуваннями, стекінг ансамбль з багатовимірних дерев повинен забезпечити:

1) підвищення показників Recall та Precision за рахунок глибшого аналізу локальних областей простору ознак;

2) покращення F1-score та MCC як результат балансу між повнотою та точністю;

3) зниження рівня overfitting у порівнянні з одиночними моделями завдяки узгодженому рішенню мета-рівня;

4) стабільність класифікації при зміні розподілу навчальних даних.

Таким чином, очікується, що інтеграція дерев з багатовимірними вузлами у структуру стекінг ансамблю забезпечить не лише локальне покращення точності на окремих класах, але й глобальну узгодженість результатів класифікації, підвищуючи ефективність моделі у задачах з високою розмірністю та складними взаємозалежностями між ознаками.

Експериментальні дослідження

Комп'ютерна система характеризується великою кількістю показників функціонування, серед яких параметри обчислювальних ресурсів, системних процесів, мережевої взаємодії, продуктивності, безпеки та зберігання даних. Однак використання всіх можливих характеристик є неможливим через обмеження обчислювальних ресурсів, пам'яті, складність оптимізації моделей та ризик перенавчання. Тому для побудови моделі ідентифікації станів КС було відібрано ключові групи показників:

- обчислювальні ресурси (завантаження процесора, пам'яті, носіїв);
- системні характеристики (операції введення/виведення, робота з носіями);
- показники мережевої взаємодії (обсяг переданих/отриманих даних);
- характеристики ОС та процесів;
- показники енергоспоживання і тепловиділення.

Збір даних здійснювався за допомогою технології Windows Management Instrumentation (WMI), яка забезпечує централізований моніторинг системних параметрів. Моніторинг проводився як у нормальному, так і аномальному стані системи. Нормальні дані збирались під час стандартної роботи навчальних комп'ютерів кафедри «Комп'ютерної інженерії та програмування» НТУ «ХПІ», а аномальні – під час виконання віртуальних середовищ, інфікованих різними типами шкідливого ПЗ (шифрувальники, віруси, хробаки, спуфінг-атаки тощо).

Отримано близько 690 тисяч зразків нормальної поведінки та 7,5 тисячі зразків аномальної, що дозволяє сформувати збалансовану навчальну вибірку для задачі класифікації. Відібрані ознаки включають показники навантаження дискових підсистем, процесора, оперативної пам'яті та мережевих інтерфейсів.

Для оцінки ефективності розроблених методів ідентифікації стану комп'ютерних систем (КС) та порівняння їх із класичними алгоритмами машинного навчання (Fine Tree, Weighted KNN, Cubic SVM) використовувались стандартні метрики точності, повноти, F1-score, коефіцієнт кореляції Метьюза (MCC), а також показники зміщення (bias) та розкиду (variance). Як показано у табл. 1, розроблені методи на основі дерев з багатовимірними вузлами рішень, нечіткого дерева та стекінгу демонструють нульове або мінімальне зміщення на етапі навчання, водночас класичні алгоритми, особливо Fine Tree, мають значну помилку розкиду через ризик перенавчання.

Таблиця 1 – **Помилки зміщення та розкиду моделей ідентифікації стану КС**

№	Метод ідентифікації стану КС	Bias, %	Variance, %
1	Fine Tree	0,13	31,97
2	Weighted KNN	0,03	10,97
3	Cubic SVM	0,07	26,07
4	Multi Decision Tree	0	9,10
5	Fuzzy Decision Tree	0,03	6,80
6	*Fuzzy Stacking with MDT*	0	0,1

Аналіз значень Accuracy та MCC (рис. 3) підтверджує здатність стекінгу до узагальнення та стабільної класифікації як позитивних, так і негативних випадків.

Високі значення F1-score, TS та DOR (рис. 4-5) демонструють збалансованість моделі та її здатність мінімізувати помилки першого і другого роду. Thread Score враховує одночасно вірно-позитивні, хибно-позитивні та хибно-негативні передбачення, що робить його корисним показником ефективності в умовах високої критичності хибних спрацьовувань.

Таким чином, стекінг дерев з багатовимірними та нечіткими вузлами рішень забезпечує найвищу точність, узагальнюючу здатність та ефективність ідентифікації станів КС, перевершуючи класичні методи та демонструючи практичну придатність для виявлення складних закономірностей у даних

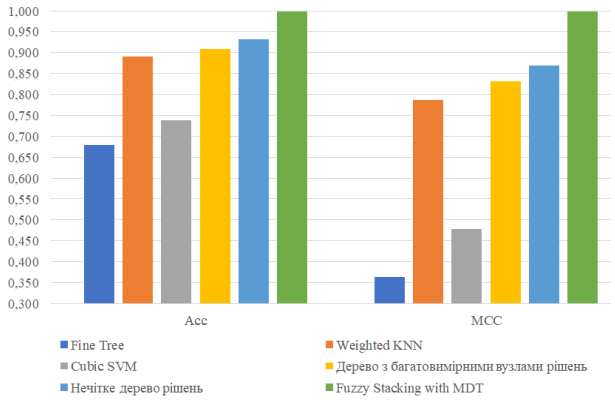

Рис. 3. Діаграми метрик ACC та MCC

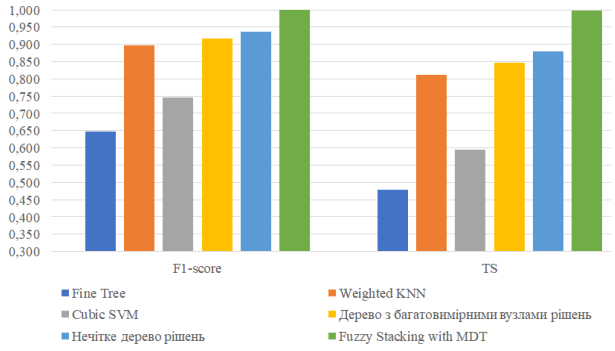

Рис. 4. Діаграми метрик F1-міри та TS

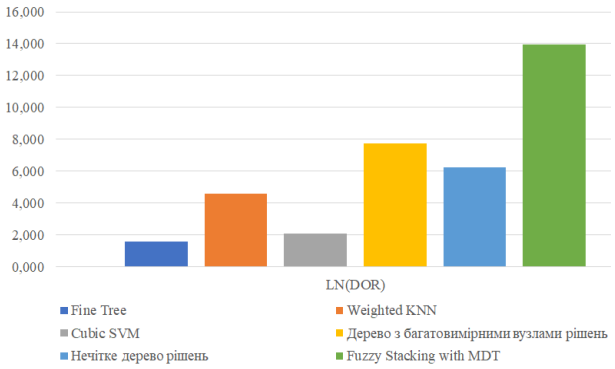

Рис. 5. Діаграма метрики Ln(DOR)

Оцінка результатів та формування рекомендацій для подальших досліджень

Проведений аналіз показав, що традиційні методи машинного навчання (Fine Tree, Weighted KNN, Cubic SVM) демонструють обмежену здатність до ефективної класифікації станів комп'ютерних систем (КС). Метод Fine Tree, зокрема, має високий рівень розкиду (Variance ≈ 31,97%) та низький коефіцієнт кореляції Метьюза (MCC ≈ 0,364), що свідчить про перенавчання та нестійкість моделі. Weighted KNN та Cubic SVM показують покращення, але все ще поступаються за ефективністю деревним методам із модифікованими вузлами.

Розроблені методи – дерево з багатовимірними вузлами рішень, нечітке дерево рішень та Fuzzy Stacking з MDT – демонструють суттєво вищу ефективність. Зокрема, стекінг на основі Fuzzy Stacking з MDT досяг майже ідеальних показників: Accuracy ≈ 0,999, MCC ≈ 0,998, F1-score ≈ 0,999, TS ≈ 0,998 та LN(DOR) ≈ 13,93, що підтверджує його високу узагальнюючу здатність та здатність виявляти складні взаємозв'язки в даних. Високе значення LN(DOR) та TS свідчить про мінімальну кількість помилок першого та другого роду, що критично для практичного застосування в інформаційних системах.

Висновки

В рамках дослідження було перевірено ефективність класичних та розроблених методів ідентифікації стану комп'ютерних систем (КС), включаючи Fine Tree, Weighted KNN, Cubic SVM, дерево з багатовимірними вузлами рішень, нечітке дерево рішень та Fuzzy Stacking з MDT. Результати показали, що класичні методи демонструють обмежену здатність до точного визначення станів КС. Метод Fine Tree має високий рівень розкиду та низьку узагальнюючу здатність, що свідчить про перенавчання та нестійкість моделі.

Найвищу ефективність продемонстрував Fuzzy Stacking з MDT, який забезпечує майже ідеальні показники Accuracy (0,999), MCC (0,998), F1-score (0,999), TS (0,998) та LN(DOR) (13,93), підтверджуючи його здатність точно класифікувати стани КС та мінімізувати помилки першого та другого роду.

Використання ансамблевих та стекінгових підходів значно підвищує точність і стабільність класифікації порівняно з окремими класичними методами.

Аналіз узагальнюючої здатності (MCC, F1-score, TS, LN(DOR)) є важливим для оцінки моделей у задачах з дисбалансом даних та аномальних станів, а також для перевірки стійкості моделей при введенні нових типів аномалій.

Подальше дослідження методів вибору та комбінування ознак КС дозволить зменшити обчислювальні витрати та підвищити інтерпретованість моделей без зниження точності.

Отже, розроблені методи ідентифікації стану КС є перспективними для практичного використання, оскільки забезпечують високу точність, збалансованість та стабільність класифікації навіть у складних умовах нерівномірного розподілу даних.

Список літератури

1. K. Shanthi and R. Maruthi, "A Comparative Study of Intrusion Detection and Prevention Systems for Cloud Environment," 2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2023, pp. 493-496, doi: https://doi.org/10.1109/ICESC57686.2023.10193694
2. Z. Chiba, N. Abghour, K. Moussaid, O. Lifandali and R. Kinta, "Review of Recent Intrusion Detection Systems and Intrusion Prevention Systems in IoT Networks," 2022 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 2022, pp. 1-6, doi: https://doi.org/10.23919/SoftCOM55329.2022.9911401
3. W. Villegas-Ch, J. Govea, R. Gutierrez, A. Maldonado Navarro and A. Mera-Navarrete, "Effectiveness of an Adaptive Deep Learning-Based Intrusion Detection System," in IEEE Access, vol. 12, pp. 184010-184027, 2024, doi: https://doi.org/10.1109/ACCESS.2024.3512363

4. R. Latha and S. J. J. Thangaraj, “Securing the Digital Perimeter: A Comprehensive Intrusion Detection System with Ensemble Learning”, 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI), Chennai, India, 2023, pp. 1-5, doi: https://doi.org/10.1109/ICDSAAI59313.2023.10452636
5. О. Горносталь, В. Челак, Класифікація мережевих атак методами машинного навчання в умовах дисбалансу тренувальних даних / Системи управління, навігації та зв’язку, Полтава, 2025, Том 3 (81), С. 64-71, doi: https://doi.org/10.26906/SUNZ.2025.3.064
6. V. Chelak, S. Gavrylenko Method of Computer System State Identification based on Boosting Ensemble with Special Preprocessing Procedure / Advanced Information Systems, Kharkiv, 2022, Vol. 6 No. 1, P. 12-18, 2022, doi: https://doi.org/10.20998/2522-9052.2022.1.02
7. O.Hornostal , S. Gavrylenko. Development of a method for identification of the state of computer systems based on bagging classifiers. Advanced Information Systems. Kharkiv, 2021, vol. 5, no. 4, pp. 5–9, doi: https://doi.org/10.20998/2522-9052.2021.4.01
8. D. Wang, S. Ji, H. Shi and J. Liu, "Power Encoding Transmission Intrusion Detection Method Based on Convolutional Autoencoders-XGBoost," 2025 2nd International Symposium on New Energy Technologies and Power Systems (NETPS), Hangzhou, China, 2025, pp. 470-474, doi: https://doi.org/10.1109/NETPS65392.2025.11102095
9. Гавриленко С. Ю., Челак В. В. Розробка методу ідентифікації стану комп'ютерної системи на основі нечітких дерев рішень / Системи управління, навігації та зв’язку, Полтава, 2023, Випуск 1 (71), С. 78-83, doi: https://doi.org/100.26906/SUNZ.2023.1.078
10. Q. -V. Dang, "Studying the Fuzzy clustering algorithm for intrusion detection on the attacks to the Domain Name System," 2021 Fifth World Conference on Smart Trends in Systems Security and Sustainability (WorldS4), London, United Kingdom, 2021, pp. 271-274, doi: https://doi.org/10.1109/WorldS451998.2021.9514038
11. Das, Abhishek, et al. "Design of deep ensemble classifier with fuzzy decision method for biomedical image classification." Applied Soft Computing, vol. 115, 2022, p. 108178. ScienceDirect, doi: https://doi.org/10.1016/j.asoc.2021.108178
12. Chatterjee, S., Das, A. An ensemble algorithm integrating consensus-clustering with feature weighting based ranking and probabilistic fuzzy logic-multilayer perceptron classifier for diagnosis and staging of breast cancer using heterogeneous datasets. Appl Intell 53, 13882–13923 (2023), doi: https://doi.org/10.1007/s10489-022-04157-0
13. Shaikh, T.A., Rasool, T., Verma, P. et al. A fundamental overview of ensemble deep learning models and applications: systematic literature and state of the art. Ann Oper Res (2024), doi: https://doi.org/10.1007/s10479-024-06444-0
14. S. Y. Gavrylenko, V. V. Chelak, S. G. Semenov Development of Method for Identification the Computer System State based on the Decision Tree with Multi-Dimensional Nodes / Radio Electronics, Computer Science, Control (RECSC), Zaporizhzhia, No. 2 (2022), P. 113-122, doi: https://doi.org/10.15588/1607-3274-2022-2-11

Received (Надійшла) 24.08.2025
Accepted for publication (Прийнята до друку) 22.10.2025

ВІДОМОСТІ ПРО АВТОРІВ / ABOUT THE AUTHORS

Челак Віктор Володимирович – PhD, доцент кафедри "Комп'ютерна інженерія та програмування", Національний технічний університет “Харківський політехнічний інститут”, Харків, Україна;
Viktor Chelak – PhD, Associate Professor of Department of "Computer Engineering and Programming", National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine;
e-mail: victor.chelak@gmail.com; ORCID ID: https://orcid.org/0000-0001-8810-3394;
Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=57216331944&origin=resultslist.

Горносталь Олексій Андрійович – PhD, асистент кафедри "Комп'ютерна інженерія та програмування", Національний технічний університет “Харківський політехнічний інститут”, Харків, Україна;
Oleksii Hornostal – PhD, Assistant Professor of Department of "Computer Engineering and Programming", National Technical University «Kharkiv Polytechnic Institute», Kharkiv, Ukraine;
e-mail: gornostalaa@gmail.com; ORCID ID: https://orcid.org/0000-0001-5820-9999;
Scopus Author ID: https://www.scopus.com/authid/detail.uri?authorId=57189040595.

Fuzzy ensemble of decision trees for the computer systems state identification

Viktor Chelak, Oleksii Hornostal

Abstract. **The object of the research** is the process of identifying the state of computer systems. **The subject of the research** is the methods of fuzzy ensemble decision trees with multidimensional decision nodes for identifying the state of computer systems. **The goal of the research** is to develop and evaluate the effectiveness of a fuzzy ensemble of decision trees to improve the accuracy of identifying computer system states under conditions of uncertainty, noise, and incomplete data. **Methods used**: machine learning methods, data preprocessing techniques, ensemble classifiers, stacking approaches, methods for feature selection and combination of computer system attributes. **Results obtained**: the effectiveness of both classical and newly developed methods for identifying the state of computer systems under complex conditions, including data imbalance and the presence of anomalous states, was investigated. A comprehensive approach using Fuzzy Stacking with MDT was proposed, providing high accuracy and stability of classification. The best results were achieved with the stacking approach, which combines base classifiers and fuzzy decision trees, minimizing both type I and type II errors and achieving high generalization ability (MCC, F1-score, TS, LN(DOR)). **Conclusions**. Based on the results of the study, an improved approach for identifying the state of computer systems is proposed, which combines the stacking method with Fuzzy MDT and feature selection optimization. The integrated use of these methods significantly enhances classification accuracy, result stability, and model robustness to data imbalance, while ensuring high-quality classification even in the presence of new anomalous states.

Keywords: state identification, computer systems, machine learning, ensemble, stacking, decision trees, fuzzy logic.