

УДК 004.852:004.8

DOI 10.31651/2076-5886-2024-1-68-79

PACS 02.60.-x, 07.05.Mh

**ПІСКУН Олександр Варфоломійович**

кандидат технічних наук, доцент,  
завідувач кафедри прикладної математики  
та інформатики, Черкаський національний  
університет ім. Б. Хмельницького  
e-mail: piskun@ukr.net  
ORCID 0000-0001-5334-6337

**КРАСНОШЛИК Наталія Олександрівна**

кандидат технічних наук, доцент, доцент  
кафедри прикладної математики та  
інформатики Черкаського національного  
університету імені Богдана  
Хмельницького  
e-mail: krasnoshlyk@vu.cdu.edu.ua  
ORCID 0000-0003-4661-6997

**СВІРЕНКО Данило Євгенійович**

розробник ПЗ, м. Черкаси

## ЗАСТОСУВАННЯ ДЕРЕВ РІШЕНЬ З НЕЙРОННОЮ ПІДТРИМКОЮ У ЗАДАЧАХ ШТУЧНОГО ІНТЕЛЕКТУ

У статті досліджено гібридні моделі машинного навчання, що поєднують дерева рішень із нейронними мережами з метою одночасного досягнення високої точності прогнозування та інтерпретованості рішень. Проаналізовано сучасні архітектури таких моделей: *Neural Decision Trees (NDT)*, *Differentiable Decision Trees*, *Neural Oblivious Decision Trees (NODE)*, *TabNet* та *Neural-Backed Decision Trees (NBDT)*. Практична реалізація включає побудову гібридної моделі на датасетах *Iris* та нобелівських лауреатів з використанням SHAP-аналізу для інтерпретації отриманих результатів, що підтверджує ефективність та практичну застосовність описаного підходу.

**Ключові слова:** дерева рішень, нейронні мережі, гібридні моделі, м'які дерева рішень, NBDT, SHAP, машинне навчання, інтерпретованість.

### Вступ

Задачі класифікації та регресії є центральними у сучасному машинному навчанні. Два найпоширеніших підходи до їх розв'язання – дерева рішень і нейронні мережі – мають принципово різні властивості та взаємодоповнювальні обмеження.

Дерева рішень є прозорими та легко інтерпретованими моделями: кожен крок логічно обґрунтований і наочно представлений, а структуру дерева можна буквально «прочитати» як схему [1]. Однак у реальних задачах, де ознаки взаємопов'язані, дані містять шум або складні залежності, дерево рішень може бути недостатньо потужним [2]. Нейронні мережі, навпаки, демонструють широкі апроксимаційні можливості та здатні виявляти приховані взаємозв'язки, які людина не може легко помітити або описати [3, 4]. Однак низька інтерпретованість нейронних мереж суттєво обмежує їх застосування у критично важливих областях.

Ця фундаментальна суперечність стимулює розвиток гібридних підходів, що поєднують переваги обох методів. Ансамблі дерев (Random Forest, Gradient Boosting), нейросимвольні методи, м'які диференційовані дерева рішень та бібліотека NBDT – все

це різні варіанти вирішення однієї проблеми балансу між точністю та пояснюваністю [5, 6].

**Метою** даного дослідження є дослідження теоретичних основ і сучасних архітектур гібридних моделей, що поєднують дерева рішень із нейронними мережами, практична реалізація таких моделей на задачах класифікації та оцінка інтерпретованості отриманих рішень засобами SHAP-аналізу.

## **Виклад основного матеріалу**

### **1. Дерева рішень і нейронні мережі у машинному навчанні**

#### **1.1. Класичні дерева рішень**

Дерево рішень – це модель машинного навчання, що використовується для розв’язання задач класифікації або регресії і працює за принципом послідовного прийняття рішень. На кожному етапі модель ставить запитання про певну ознаку об’єкта: «Чи значення ознаки більше порогу?», «Чи виконується умова?». Залежно від відповіді модель переходить до наступного запитання, доки не дійде до остаточного рішення – прогнозу [2].

Класичні алгоритми побудови дерев становлять основу багатьох алгоритмів машинного навчання. Алгоритм ID3, запропонований Квінланом у 1986 році, використовує інформаційний приріст для вибору атрибутів розгалуження. Модифікація C4.5 вирішує проблему схильності до атрибутів з великою кількістю значень шляхом використання коефіцієнта інформаційного приросту. Алгоритм CART (Classification and Regression Trees) застосовує індекс Джині для класифікації та середньоквадратичну помилку для регресії, підтримуючи як категоріальні, так і числові атрибути [1].

Головною перевагою класичних дерев є їх інтерпретованість: кожне рішення може бути простежене через послідовність логічних умов. Проблема вибору оптимальної структури дерева залишається однією з ключових проблем у традиційних підходах. Методи прунінгу – pre-pruning та post-pruning – спрямовані на мінімізацію складності дерева при збереженні точності прогнозування. Разом з тим, у задачах з нелінійними залежностями, зашумленими даними або складними взаємодіями між ознаками дерева рішень поступаються більш виразним моделям [1].

#### **1.2. Нейронні мережі та їх роль у машинному навчанні**

На відміну від дерев рішень, нейронні мережі будують свої передбачення, імітуючи принципи роботи мозку. Вони складаються з великої кількості взаємопов’язаних нейронів – математичних функцій, що отримують вхідні дані, перетворюють їх і передають далі через мережу. Кожен нейрон вчиться виявляти певні шаблони або закономірності в даних, і чим більше таких рівнів (шарів), тим складніші залежності здатна розпізнати модель [3].

Нейронні мережі та глибоке навчання революціонізували машинне навчання завдяки можливості автоматичного вивчення представлень даних. Багатошарові перцептрони використовують зворотне поширення похибки для навчання ваг синапсів. Сучасні архітектури, такі як згорткові та рекурентні мережі, демонструють гарні результати у комп’ютерному зорі, обробці природної мови та інших областях [3, 4].

Ключовою перевагою нейронних мереж є їх універсальна апроксимаційна здатність: теоретично, достатньо широка мережа може наблизити будь-яку неперервну функцію. Практично це означає можливість моделювання складних залежностей у даних без необхідності ручного конструювання ознак. Функція активації відіграє критичну роль у здатності мережі до навчання нелінійних залежностей; ReLU, сигмоїд, tanh та їх варіації забезпечують різні властивості градієнтного потоку.

Процес навчання нейронних мереж базується на мінімізації функції втрат через градієнтний спуск. Регуляризаційні техніки, такі як L1/L2-регуляризація чи dropout, запобігають перенавчанню та покращують узагальнюючу здатність моделі.

### 1.3. Гібридні моделі та їх класифікація

Гібридні моделі в машинному навчанні представляють спробу об'єднати переваги різних підходів. Ансамблі методів, такі як Random Forest та Gradient Boosting, поєднують множину простих моделей для досягнення кращої точності. Нейросимвольні підходи інтегрують логічне мислення з нейронними обчисленнями [6].

Специфічні гібриди дерев рішень та нейронних мереж можна класифікувати за способом інтеграції:

- послідовна інтеграція: нейронна мережа обробляє вихід дерева або навпаки;
- паралельна інтеграція: результати об'єднуються зваженою сумою;
- вбудована інтеграція: нейронні компоненти інтегровані безпосередньо у структуру дерева.

Кожен підхід має свої переваги та обмеження в контексті конкретних застосувань.

## 2. Сучасні моделі дерев рішень з нейронною підтримкою

### 2.1. Neural Decision Trees (NDT)

Neural Decision Trees (NDT) представляють піонерський підхід до створення диференційованих дерев рішень [7]. Основна ідея полягає у заміні дискретних розгалужень «м'якими» функціями, що дозволяє використовувати градієнтні методи оптимізації. Кожен внутрішній вузол дерева містить нейронну мережу, що обчислює ймовірність переходу до кожної з дочірніх гілок.

Архітектура NDT дозволяє кінцевому прогнозу бути зваженою комбінацією всіх листових вузлів, де ваги визначаються добутком ймовірностей проходження шляху від кореня до листа. Такий підхід забезпечує гладкість функції втрат та можливість ефективного навчання методом зворотного поширення.

Математично ймовірність досягнення листа  $l$  через шлях від кореня обчислюється як добуток ймовірностей у всіх проміжних вузлах:

$$P(l | x) = \prod P(\text{decision}_i | x),$$

де  $\text{decision}_i$  – рішення в  $i$ -му вузлі шляху.

Ключовою інновацією NDT є використання диференційованих функцій розщеплення. Замість дискретних порогових функцій кожен вузол використовує сигмоїдну функцію:

$$P(\text{right} | x) = \sigma(f(x)),$$

де  $f(x)$  – лінійна або нелінійна функція вхідного вектора.

Така архітектура дозволяє навчати всі параметри дерева одночасно через стандартні методи оптимізації [7].

### 2.2. Differentiable Decision Trees

Differentiable Decision Trees розширюють концепцію NDT шляхом використання більш складних функцій розгалуження. Замість простих лінійних перетворень у вузлах застосовуються повноцінні нейронні мережі, здатні виявляти нелінійні залежності в даних. Особливу увагу приділено методам регуляризації, що запобігають деградації структури дерева до суцільної нейронної мережі.

Регуляризація структури дерева включає штрафи за складність окремих функцій розщеплення та загальну глибину дерева. Функція втрат має такий вигляд:

Під час виведення зразок проходить через дерево, починаючи з кореня, і на кожному кроці вибирається дочірній вузол з найбільшою косинусною подібністю до поточного зразка.

Ключовою інновацією NBDT є Tree Supervision Loss – спеціалізована функція втрат, що поєднує стандартну втрату класифікації з додатковим членом, який заохочує правильне прийняття рішень на всіх рівнях дерева. Це дозволяє моделі навчатися не лише фінальній класифікації, але й проміжним рішенням на шляху до неї.

Бібліотека підтримує два основні режими роботи:

- м'який вивід (soft-режим) – використовує ймовірнісну маршрутизацію, де зразок може «протікати» через множину шляхів з різними вагами; забезпечує кращу точність;
- жорсткий вивід (hard-режим) – застосовує детерміністичну маршрутизацію, вибираючи єдиний шлях на кожному кроці; надає більш чітку інтерпретацію.

NBDT демонструє такі результати на стандартних датасетах комп'ютерного зору: на CIFAR-10 досягається точність 97.55%, що на 3.23% краще за найкращі попередні методи поєднання дерев рішень та глибокого навчання; на CIFAR-100 покращення становить 6.73% з досягненням 82.97% точності; на ImageNet NBDT досягає 76.60% точності, що на 15.31% краще за попередні підходи [9]. Особливо важливою є здатність NBDT до узагальнення на незнайомі класи: експерименти показують покращення до 16% при класифікації зразків з класів, що не використовувалися під час навчання.

### **3. Компоненти гібридних дерев рішень та алгоритми навчання**

#### **3.1. М'які дерева рішень з диференційованими розгалуженнями**

Традиційні дерева рішень працюють як жорстка система «так – ні»: дані у кожному вузлі направляються лише в одну гілку. Гібридні дерева рішень працюють інакше – вони дозволяють даним проходити через усі можливі шляхи одночасно, але з різними ступенями ймовірності.

Така архітектура має важливу перевагу: усі операції стають гладкими і диференційованими, що дозволяє використовувати стандартні методи машинного навчання для оптимізації моделі.

Нейронні мережі у вузлах дерева замінюють просту логіку порівняння повноцінними малими нейронними мережами (зазвичай 1-2 приховані шари з 32-128 нейронами), які здатні виявляти складні залежності між різними ознаками. Для запобігання перенавчанню використовуються: dropout у функціях розщеплення, обмеження складності параметрів та нормалізація даних для стабільності навчання.

Механізми уваги дозволяють дереву динамічно вибирати, на які ознаки звертати увагу для кожного конкретного зразка – аналогічно до того, як людина залежно від ситуації звертає увагу на різні аспекти проблеми. Механізм самоуваги дозволяє моделі аналізувати взаємодії між різними ознаками, що є особливо корисним для табличних даних з можливими складними залежностями між стовпцями. Багатополюсна увага дає моделі можливість одночасно фокусуватися на різних аспектах даних.

Ансамблі нейронних дерев замість використання одного дерева об'єднують кілька різних гібридних моделей для отримання більш точних та стабільних результатів. На відміну від традиційних лісів, що просто навчають багато схожих дерев на різних підвбірках даних, ансамблі нейронних дерев можуть використовувати різні архітектури, різні способи навчання чи різні критерії оптимізації. Методи поєднання прогнозів включають: просте усереднення, зважене усереднення та навчання спеціальної мета-моделі [6].

### 3.2. Алгоритми навчання гібридних моделей

*Градiєнтний спуск* є основним інструментом навчання гібридних дерев рішень. Традиційні дерева рішень складно оптимізувати за допомогою градiєнтних методів через їх дискретну природу: неможливо обчислити, як саме малі зміни параметрів вплинуть на результат. Гібридні моделі вирішують цю проблему, роблячи всі операції гладкими та диференційованими. Процес навчання полягає у мінімізації функції втрат, що вимірює різницю між прогнозами моделі та правильними відповідями, доповненої регуляризаційними членами. Схема процесу:

початкові параметри → прогноз → обчислення втрат → обчислення градiєнтів → оновлення параметрів → повторення.

Обчислення градiєнтів для гібридних дерев складніше, ніж для звичайних нейронних мереж, оскільки потрібно враховувати вплив змін в одному вузлі на всю структуру дерева. Стабільність градiєнтів критично важлива – використовується обрізання градiєнтів для запобігання різким стрибкам параметрів; адаптивні методи навчання, що автоматично підстроюють швидкість навчання, зазвичай працюють краще за стандартні підходи.

*Регуляризація* є необхідним компонентом навчання гібридних моделей, оскільки висока виразна потужність робить їх схильними до перенавчання. Основні типи регуляризації представлено у таблиці 1.

Таблиця 1

Основні типи регуляризації гібридних дерев рішень

Тип регуляризації	Опис	Застосування
Параметрична	Обмеження складності параметрів мереж	Штрафи за великі ваги
Структурна	Контроль архітектури дерева	Обмеження глибини, кількості вузлів
Стохастична	Випадкове відключення компонентів	Dropout у функціях розщеплення
Ентропійна	Заохочення визначених рішень	Штраф за невизначеність у вузлах

Рання зупинка навчання реалізується через відстеження якості моделі на валідаційних даних: якщо протягом певного часу покращення не спостерігається, навчання припиняється. Dropout у деревах рішень може застосовуватися по-різному: відключення цілих вузлів, окремих нейронів у функціях розщеплення чи випадкове блокування деяких шляхів проходження даних через дерево.

*Прунінг* – процес спрощення дерева через видалення найменш важливих частин. У гібридних моделях це складніше, ніж у традиційних деревах, оскільки всі вузли впливають на фінальний результат. Етапи прунінгу:

навчання повної моделі → оцінка важливості вузлів → поступове видалення → донавчання.

Методи оцінки важливості включають аналіз розмірів параметрів (менші параметри – менш важливі вузли), аналіз градієнтів (вузли з малими градієнтами менш важливі) та аналіз впливу на валідаційну втрату.

*Оптимізація гіперпараметрів* гібридних моделей передбачає налаштування таких параметрів, як глибина дерева, розмір нейронних мереж у вузлах, швидкість навчання, коефіцієнти регуляризації. Прості методи (пошук по сітці, випадковий пошук) прості у реалізації, але неефективні для складних просторів параметрів. Інтелектуальні методи (Байєсова оптимізація, еволюційні алгоритми) потребують меншої кількості обчислювальних ресурсів. Комбінований метод ВОНВ поєднує переваги різних підходів: ефективність Байєсової оптимізації зі швидкістю випадкового пошуку та раннього зупинення неперспективних конфігурацій.

## 4. Практична реалізація та результати

### 4.1. Середовище реалізації та інструментарій

У роботі розроблено серію демонстраційних програм мовою Python з використанням бібліотек scikit-learn (класифікатори MLP та DecisionTree), NumPy, matplotlib, shar та pandas. Для демонстрації бібліотеки NBDT використовується PyTorch та torchvision. Програми реалізовані з підвищенням складності від базової гібридної моделі до повноцінного аналізу з поясненням рішень.

### 4.2. Гібридна модель на датасеті Iris

Перший програмний приклад (програма snt1.py) демонструє підхід до створення гібридної моделі, що поєднує нейронну мережу та дерево рішень на датасеті Iris (150 зразків, 4 ознаки, 3 класи квіток).

Логіка побудови моделі:

1. Навчання нейронної мережі MLP (1 прихований шар з 10 нейронами, 1000 ітерацій).
2. Отримання вектора ймовірностей як нейронної підтримки: `train_proba = mlp.predict_proba(X_train)`.
3. Розширення простору ознак: оригінальні 4 ознаки доповнюються 3 новими – ймовірностями для кожного класу, таким чином простір ознак розширюється з 4 до 7 вимірів: `X_train_augmented = np.hstack([X_train, train_proba])`.
4. Навчання дерева рішень на розширеному наборі ознак (`max_depth=4`).

Дерево рішень отримує доступ до ймовірностей від нейронної мережі як до додаткових ознак, що дозволяє йому використовувати узагальнену «думку» нейронної мережі при прийнятті рішень. Результатом є точність гібридної моделі 1.0 (100%) на тестовій вибірці. Висока точність пояснюється кількома чинниками. По-перше, датасет Iris є добре структурованим: вид Iris setosa практично лінійно відокремлений від двох інших видів. По-друге, гібридний підхід дозволяє об'єднати переваги двох принципово різних типів моделей: нейронна мережа відмінно справляється з виявленням складних нелінійних закономірностей, тоді як дерево рішень пропонує структурований, логічний підхід до класифікації. Розширення простору ознак створює багатший контекст – додаткові ознаки з нейронної мережі особливо корисні у випадках, коли зразок знаходиться на межі між двома класами за фізичними характеристиками.

### 4.3. Візуалізація гібридного дерева рішень

Другий програмний приклад (програма snt2.py) є розширенням попереднього та додає повноцінну візуалізацію структури дерева рішень засобами функції `plot_tree` бібліотеки matplotlib. Для цього формується список назв розширеного простору ознак: оригінальні 4 назви (sepal length, sepal width, petal length, petal width) та три нових (NN\_proba\_setosa, NN\_proba\_versicolor, NN\_proba\_virginica).

Таблиця 2

Порівняння точності базових методів та гібридної моделі на датасеті Iris

Модель	Опис	Точність на тестовій вибірці
MLP	1 прихований шар, 10 нейронів	~97%
Дерево рішень	max_depth=4, 4 ознаки	~97%
Гібридна модель	Дерево з нейронною підтримкою, 7 ознак	100%

Графічна репрезентація дерева (рис. 1) розкриває внутрішню логіку гібридної моделі. Кольорове кодування вузлів надає миттєву візуальну інформацію про чистоту класифікації в кожному розгалуженні. У вузлах дерева присутні як оригінальні ознаки (sepal width (cm)), так і нейронні ймовірності (NN\_proba\_virginica  $\leq$  0.016, NN\_proba\_versicolor  $\leq$  0.504), що підтверджує активне використання нейронної підтримки при прийнятті рішень. Умови типу NN\_proba\_setosa  $\leq$  0.001 демонструють, що дерево активно використовує висновки нейронної мережі як повноцінні ознаки для класифікації.

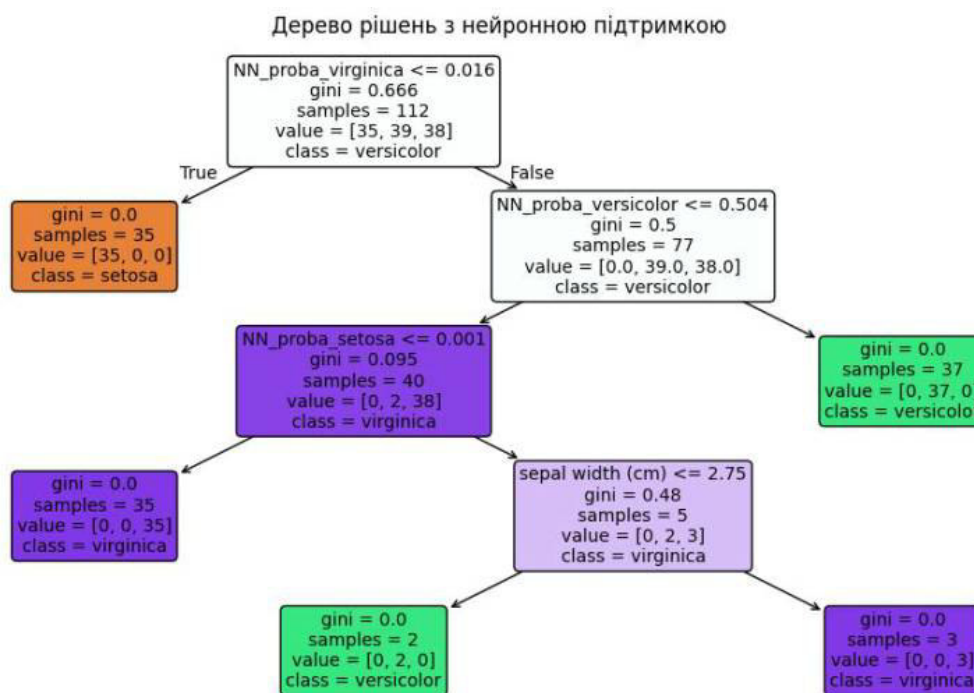


Рис. 1. Візуалізація дерева рішень з нейронною підтримкою

#### 4.4. SHAP-аналіз для інтерпретації рішень

Третій програмний приклад (програма snt3.py) додає до гібридної моделі можливості пояснюваного штучного інтелекту (Explainable AI) через інтеграцію бібліотеки SHAP (SHapley Additive exPlanations).

SHAP-аналіз виявляє нові можливості розуміння поведінки гібридної моделі, надаючи кількісні метрики впливу кожної ознаки на процес прийняття рішень. Особливо цінним є аналіз взаємодій між оригінальними фізичними ознаками та нейронними ймовірностями: у деяких випадках нейронні ймовірності мають вищий вплив на рішення, ніж оригінальні ознаки, що свідчить про ефективність гібридного

підходу; в інших – традиційні ознаки домінують, що вказує на ситуації, де пряме вимірювання є більш інформативним за складні нейронні обчислення.

Для аналізу дерев рішень застосовується `shap.TreeExplainer`, що використовує ефективні алгоритми для точного обчислення значень Шеплі.

Метод `shap_values` повертає матрицю, де кожен елемент показує, наскільки конкретна ознака збільшує або зменшує ймовірність віднесення зразка до конкретного класу. Функція `summary_plot` створює комплексну візуалізацію (рис. 2), що поєднує інформацію про важливість ознак з їх значеннями для всіх зразків тестової вибірки.

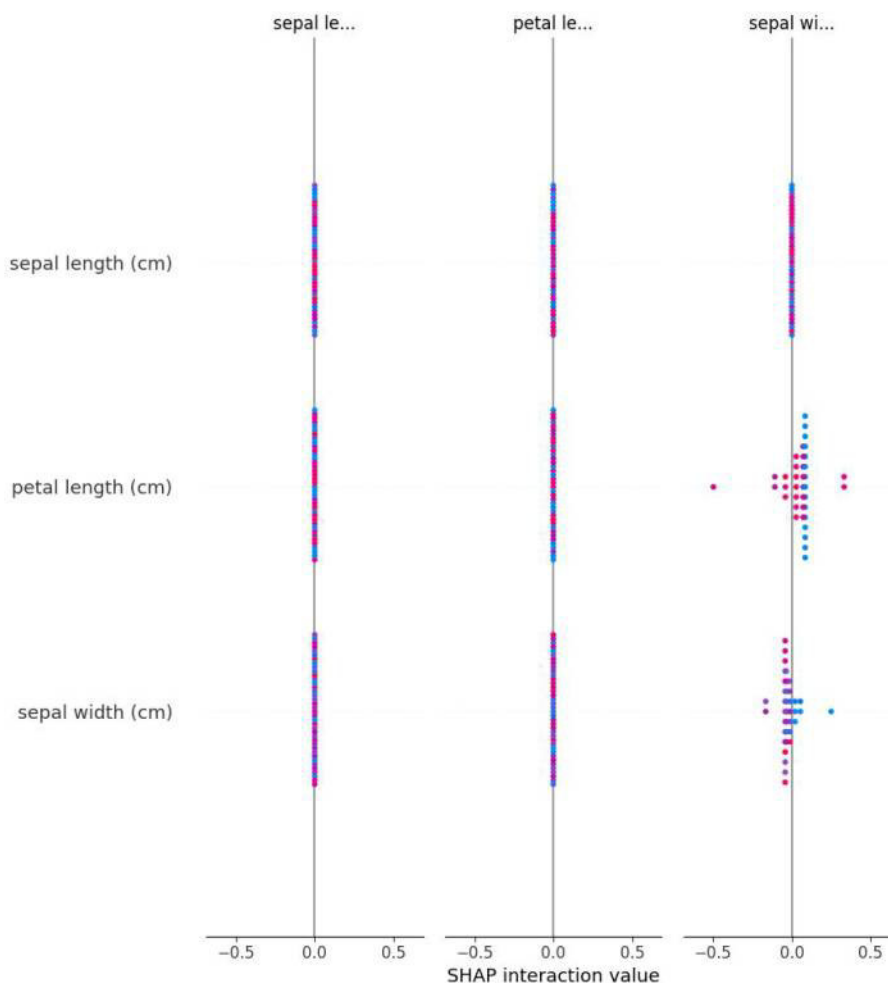


Рис.2. Візуалізація результатів SHAP-аналізу

Кольорове кодування на Summary Plot розкриває нелінійні залежності між значеннями ознак та їх впливом на класифікацію: висока довжина пелюстки може мати позитивний SHAP-вплив для класу *virginica*, але негативний для *setosa*. Загальна картина SHAP-аналізу створює основу для подальшого вдосконалення гібридної моделі, надаючи конкретні, кількісно обґрунтовані рекомендації щодо того, які аспекти моделі працюють найефективніше.

#### 4.5. Застосування до реальних даних: датасет нобелівських лауреатів

Четвертий програмний приклад (програма `snt4.py`) демонструє перехід від класичного датасету *Iris* до практичного застосування гібридної моделі на реальних даних про нобелівських лауреатів. Для роботи з CSV-файлом використовується бібліотека `pandas`.

Особливістю реалізації є створення нових ознак зі стовпця дати народження: обчислюється рік народження та похідна ознака «вік лауреата на момент отримання премії». Категоріальні ознаки (стать, країна народження) кодуються за допомогою LabelEncoder. Числові ознаки масштабуються за допомогою StandardScaler, що є критично важливим для нейронної мережі при роботі з ознаками різного масштабу (рік може бути 1900–2020, тоді як стать кодується як 0 або 1). Цільова змінна – категорія нобелівської премії, закодована також через LabelEncoder.

Простір ознак гібридної моделі формується динамічно залежно від кількості категорій премій: ["sex", "country", "year", "age"] + [f'NN\_proba\_{i}' for i in range(train\_proba.shape)] [8].

Точність моделі на цьому датасеті є нижчою порівняно з Iris через значно більшу складність та суб'єктивність процесу присудження нобелівських премій: рішення комітетів залежать від багатьох факторів, які не відображені в базових демографічних даних, що робить задачу класифікації значно складнішою. Тим не менш гібридна модель стабільно перевершує базові методи, а дерево рішень виявляє цікаві закономірності – наприклад, щодо ролі нейронних ймовірностей у першому вузлі дерева та взаємодії демографічних ознак із категоріями премій.

## 5. Аналіз результатів

Результати порівняльного аналізу розглянутих гібридних архітектур наведено в таблиці 3.

Таблиця 3

Порівняльний аналіз розглянутих гібридних архітектур

Архітектура	Основна ідея	Тип розщеплення	Інтерпретованість	Ефективність на табл. даних
NDT [4]	М'які розгалуження	Лінійна сигмоїдна	Середня	Середня
Differentiable DT	Нейромережі у вузлах	Нелінійна нейромережа	Середня	Висока
NODE [11]	Oblivious-структура	Entmax-зважена	Низька	Висока
TabNet [10]	Sequential attention	Attention-маски	Висока	Висока
NBDT [9]	Ієрархія з ваг FC-шару	Косинусна подібність	Висока	Середня
Нейронна підтримка	Ймовірності MLP як ознаки	Стандартна (CART)	Висока	Середня

Порівняння продуктивності показує, що гібридні моделі особливо ефективні на датасетах середнього розміру, де ансамблеві методи можуть страждати від перенавчання, а прості моделі – від недостатньої виразності. Ансамблі дерев мають переваги у стійкості через агрегацію множини моделей, однак це призводить до значно більшої обчислювальної складності як під час навчання, так і під час виведення.

Гібридні моделі пропонують компромісне рішення, поєднуючи виразність складних моделей з відносною простотою одиначної архітектури [11].

### Висновки

У роботі проведено комплексне дослідження гібридних моделей машинного навчання, що поєднують дерева рішень із нейронними мережами.

Встановлено, що традиційні дерева рішень і нейронні мережі мають взаємодоповнювальні властивості, а гібридні архітектури ефективно поєднують переваги обох підходів. Розглянуто сучасні гібридні архітектури (NDT, Differentiable Decision Trees, NODE, TabNet, NBDT) та виділено їх ключові особливості з точки зору типу розщеплення, навчальності та інтерпретованості.

Практично реалізовано та верифіковано метод нейронної підтримки: на датасеті Iris гібридна модель досягла точності 100%, перевершивши обидва базових методи. Засобами SHAP-аналізу підтверджено, що нейронні ймовірності є значущими предикторами для дерева рішень і відіграють вирішальну роль у вузлах з високою невизначеністю. Застосування методу до датасету нобелівських лауреатів демонструє його практичну застосовність на реальних соціальних даних.

Розглянута бібліотека NBDT підтверджує, що описаний принцип масштабується до задач комп'ютерного зору (97.55% на CIFAR-10, 82.97% на CIFAR-100, 76.60% на ImageNet), що свідчить про широкий потенціал гібридних нейронно-символьних підходів.

Подальші дослідження доцільно спрямувати на аналіз поведінки запропонованих моделей при підвищенні розмірності та зашумленості даних, а також на дослідження методів автоматичного визначення оптимального способу інтеграції нейронної та символічної компонент.

### Список використаної літератури:

1. Breiman L., Friedman J., Stone C. J., Olshen R. A. Classification and regression trees. – CRC press, 1984.
2. Quinlan J. R. Induction of decision trees // Machine learning. – 1986. – Vol. 1 (1). – P. 81–106.
3. Goodfellow I., Bengio Y., Courville A. Deep learning. – MIT press, 2016. – 800 p.
4. LeCun Y., Bengio Y., Hinton G. Deep learning // Nature. – 2015. – Vol. 521 (7553). – P. 436–444.
5. Kotschieder P., Fiterau M., Criminisi A., Rota Bulò S. Deep neural decision forests // Proceedings of the IEEE International Conference on Computer Vision (ICCV). – 2015. – P. 1467–1475.
6. Zhou Z. H. Ensemble methods: foundations and algorithms. – CRC press, 2012.
7. Irsoy O., Yıldız O. T., Alpaydm E. Soft decision trees // In 21st International Conference on Pattern Recognition (ICPR). – 2012. – P. 1819–1822.
8. Kaddour J., Lynch A., Liu Q., Kusner M. J., Silva R. When do neural nets outperform boosted trees on tabular data? // arXiv preprint arXiv:2305.02997. – 2022.
9. Neural-Backed Decision Trees [Електронний ресурс]. – Режим доступу: <https://research.alvinwan.com/neural-backed-decision-trees>. – Назва з екрана.
10. Arik S. Ö., Pfister T. TabNet: Attentive interpretable tabular learning // Proceedings of the AAAI Conference on Artificial Intelligence. – 2021. – Vol. 35, No. 8. – P. 6679–6687.
11. Popov S., Morozov S., Babenko A. Neural oblivious decision trees for deep learning on tabular data // arXiv preprint arXiv:1909.06312. – 2019.

### References:

1. Breiman L., Friedman J., Stone C. J., Olshen R. A. (1984) Classification and regression trees. CRC press.
2. Quinlan J. R. (1986) Induction of decision trees. Machine learning, 1 (1), pp. 81–106.
3. Goodfellow I., Bengio Y., Courville A. (2016) Deep learning. MIT press.
4. LeCun Y., Bengio Y., Hinton G. (2015) Deep learning. Nature, 521 (7553), pp. 436–444.
5. Kotschieder P., Fiterau M., Criminisi A., Rota Bulò S. (2015) Deep neural decision forests. Proceedings of the IEEE ICCV, pp. 1467–1475.
6. Zhou Z. H. (2012) Ensemble methods: foundations and algorithms. CRC press.
7. Irsoy O., Yıldız O. T., Alpaydm E. (2012) Soft decision trees. 21st International Conference on Pattern Recognition (ICPR), pp. 1819–1822.

8. Kaddour J., Lynch A., Liu Q., Kusner M. J., Silva R. (2022) When do neural nets outperform boosted trees on tabular data? arXiv:2305.02997.
9. Neural-Backed Decision Trees. Available at: <https://research.alvinwan.com/neural-backed-decision-trees/>
10. Arik S. Ö., Pfister T. (2021) TabNet: Attentive interpretable tabular learning. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 8, pp. 6679–6687.
11. Popov S., Morozov S., Babenko A. (2019) Neural oblivious decision trees for deep learning on tabular data. arXiv:1909.06312.

**PISKUN Oleksandr,**

Candidate of Technical Sciences, Associate Professor, Head of Department of Applied Mathematics and Informatics, Bohdan Khmelnytsky National University of Cherkasy

**KRASNOSHLYK Nataliya,**

Candidate of Technical Sciences, Associate Professor, Department of Applied Mathematics and Informatics, The Bohdan Khmelnytsky National University of Cherkasy, Ukraine

**SVIRENKO Danylo**

Software Developer, Cherkasy, Ukraine

**APPLICATION OF NEURAL-ASSISTED DECISION TREES IN ARTIFICIAL INTELLIGENCE PROBLEMS*****Summary. Introduction.***

The article investigates hybrid machine learning models that combine decision trees with neural networks to simultaneously achieve high prediction accuracy and interpretability of decisions. The modern architectures of the following models are analyzed: Neural Decision Trees (NDT), Differentiable Decision Trees, Neural Oblivious Decision Trees (NODE), TabNet, and Neural-Backed Decision Trees (NBDT). The practical implementation includes building a hybrid model on the Iris and Nobel laureate datasets using SHAP analysis to interpret the results, which confirms the effectiveness and practical applicability of the described approach.

Modern machine learning faces the fundamental challenge of balancing predictive accuracy with model interpretability. Classical decision trees offer transparent, rule-based reasoning but are limited in capturing complex nonlinear patterns. Neural networks achieve state-of-the-art accuracy but function as opaque "black boxes." Hybrid models that combine the strengths of both approaches represent a promising research direction, particularly for safety-critical applications where algorithmic decisions must be explainable.

***Purpose.*** The aim of this article is to investigate the theoretical foundations and modern architectures of hybrid models combining decision trees with neural networks, to practically implement such models on classification tasks, and to evaluate the interpretability of obtained decisions using SHAP analysis.

***Results.*** The paper systematizes and compares five classes of modern hybrid architectures: Neural Decision Trees (NDT), Differentiable Decision Trees, Neural Oblivious Decision Trees (NODE), TabNet, and Neural-Backed Decision Trees (NBDT). The components of hybrid decision trees are examined in detail: soft differentiable branching, neural networks as split functions, attention mechanisms, and ensemble approaches. Training algorithms – gradient descent, regularization types (parametric, structural, stochastic, and entropy-based), pruning, and hyperparameter optimization – are described. The NBDT library achieves 97.55% on CIFAR-10, 82.97% on CIFAR-100, and 76.60% on ImageNet, surpassing previous hybrid methods by 3.23%, 6.73%, and 15.31% respectively. A two-stage hybrid model is implemented, where the probabilistic outputs of an MLP are used as additional features for a decision tree. The hybrid model achieves 100% accuracy on the Iris dataset, outperforming both baseline models. SHAP analysis confirms that neural probability features play a decisive role in uncertain nodes, while original features dominate in straightforward cases. Application to the Nobel laureates dataset further validates the approach on real social data.

***Conclusion.*** Hybrid models combining decision trees with neural network support effectively resolve the accuracy–interpretability trade-off. The proposed neural support mechanism enriches the feature space of a decision tree without sacrificing its structural interpretability, as confirmed through experiments on two different datasets and quantitative SHAP-based explanations.

***Keywords:*** decision trees, neural networks, hybrid models, soft decision trees, NBDT, SHAP, machine learning, interpretability.

Одержано редакцією 05.09.2024 р.  
Прийнято до публікації 30.10.2024 р.

УДК 004.415.2:004.5

DOI 10.31651/2076-5886-2024-1-79-90

PACS 07.05.Tr, 89.20.Ff

**ВОЙЦІХОВСЬКА Лєна Іванівна**  
студентка спеціальності «Інформаційні системи та технології» Черкаського національного університету імені Богдана Хмельницького

**ДЗІЮБА Вікторія Анатоліївна**  
кандидат технічних наук, старший викладач кафедри прикладної математики та інформатики Черкаського національного університету імені Богдана Хмельницького  
e-mail: viktoriya.dzyuba15@vu.cdu.edu.ua  
ORCID 0000-0003-1655-0333

## РОЗРОБКА ІНФОРМАЦІЙНО-ТЕСТУВАЛЬНОГО ВЕБ-ДОДАТКУ ДЛЯ НАВЧАЛЬНОГО ЗАКЛАДУ

*У статті розглядається процес проєктування, розробки та розгортання сучасного веб-застосунку для навчальної платформи. Детально аналізується архітектура системи, включаючи бази даних PostgreSQL, використання ORM Prisma, а також побудова серверної частини на основі Node.js, TypeScript та фреймворку NestJS. Описано принципи організації REST API, механізми аутентифікації та авторизації користувачів із використанням JWT, а також підходи до безпечного зберігання даних і управління доступом.*

*У роботі також висвітлено розробку клієнтської частини застосунку із використанням React, Vite, Redux та бібліотеки MUI, що дозволило створити інтерактивний і масштабований інтерфейс користувача. Особливу увагу приділено процесу проєктування інтерфейсу у Figma та практичній реалізації системи розгортання за допомогою Docker, MinIO для зберігання файлів і MailHog для тестування електронної пошти. Результатом роботи є комплексне програмне рішення, яке поєднує сучасні технології веб-розробки та забезпечує стабільність, безпеку й зручність використання*

**Ключові слова:** веб-застосунок, інформаційна система, клієнт-серверна архітектура, база даних, PostgreSQL, Prisma, Node.js, TypeScript, NestJS, REST API, JWT, аутентифікація, авторизація, React, SPA, Redux, Vite, MUI, Figma, Docker, MinIO, MailHog, розгортання, веб-розробка.

### Вступ

Сучасний ритм життя вносить свої корективи в наші плани та буденність, змушуючи змінювати тимчасово чи на зовсім своє місце проживання, перебувати по декілька годин в укритті, правильно виставляти пріоритети та економити електроенергію. Увесь світ, зокрема Україна, проходить зараз етап цифровізації, щоб будь-яка людина змогла зробити базові речі, не виходячи з дому. Це покупки в магазині, оформлення документів, запис до певного спеціаліста і тому подібне. Таким чином, у деякій мірі, можна забезпечити вдале поєднання безпеки та комфорту людського життя.

Цифровізації зазнає і освіта, адже у такому потоці хаосу потрібно продовжувати навчатись та розвиватись, незалежно від розташування та речей поруч. Для цього