



О. Й. Пісун, В. В. Рудик, Ю. М. Батько

Західноукраїнський національний університет, м. Тернопіль, Україна

КЛАСИФІКАЦІЯ ВЕБСАЙТІВ ЗА КРИТЕРІЄМ ЇХ ПРОДУКТИВНОСТІ НА ПІДСТАВІ АНСАМБЛІВ

Розглянуто підходи до визначення продуктивності вебсайтів та їхню класифікацію з подальшим розподілом на категорії. Встановлено, що серед сучасних алгоритмів до класифікації даних виділяють ансамблеві методи, зокрема soft та hard голосування для отримання найкращого результату. Основними критеріями оцінювання є: Page Load Time, Time To First Byte, Server Response Time, First Contentful Paint, Speed Index. Пошукові алгоритми, зокрема Google, віддають перевагу сайтам з високою швидкістю і плавною взаємодією, що підвищує їх видимість у результатах пошуку. Окрім цього, допустима продуктивність знижує показник відмов і покращує конверсію, що є критично важливим для успішного просування в онлайн. Оцінено вплив показників основних параметрів вебсайтів на швидкість їх завантаження та відповідно класифікації на три категорії. Оцінювання продуктивності вебсайтів є актуальним завданням, де швидкість завантаження та стабільність відображення істотно впливають на користувацький досвід та конверсію. Використання ансамблевих методів у завданні класифікації даних базується на принципі поєднання кількох моделей або джерел даних для підвищення загальної точності і стійкості прогнозів. Ансамблі дадуть змогу зменшити вплив випадкових помилок окремих моделей, усунути зміщення та знизити дисперсію результатів. За рахунок інтеграції різнорідних характеристик або ознак, ансамблеві дані забезпечують більш комплексне уявлення про об'єкт класифікації, що особливо важливо під час роботи з неоднорідними або складними за структурою наборами даних. Ефективне оцінювання та аналіз продуктивності дає змогу виявити проблемні місця в роботі сайту, раціонально використати ресурси та забезпечити безперебійний доступ для користувачів. На підставі експериментального підходу виділено кращі алгоритми для навчання без вчителя, що дало можливість розробити ансамблевий метод для класифікації вебсайтів на підставі показників їх продуктивності. Точність становила 98 % на тестовій вибірці. Наявність такого рішення дає змогу підбирати допустимі версії вебсторінок для конкретного типу вебсторінок.

Ключові слова: аналіз даних; логістична регресія; метод найближчих сусідів; швидкість завантаження сторінки; швидкість завантаження сайту.

Вступ / Introduction

Підбір адаптивних і легких сторінок вебсайтів є надзвичайно актуальним для забезпечення якісної роботи сайту, оскільки такі сторінки швидко завантажуються, коректно відображаються на різних пристроях і покращують користувацький досвід. Легкі сторінки зменшують тривалість завантаження та знижують кількість HTTP-запитів, що позитивно впливає на підставні параметри продуктивності. Адаптивність гарантує допустиме відображення контенту на мобільних телефонах, планшетах і десктопах, що підвищує час до інтерактивності і знижує показник відмов. Як наслідок, сайт стає більш доступним, швидким і привабливим для користувачів і пошукових систем, що є важливою умовою ефективного SEO-удосконалення та конверсії.

Ансамблеві методи важливо розглядати, бо вони дадуть змогу поєднувати переваги кількох моделей для підвищення точності та надійності прогнозів. Завдяки

агрегації результатів різних алгоритмів ансамблі знижують ймовірність помилок, зменшують вплив шуму та уникнення перенавчання.

Актуальність запропонованої роботи полягає у розробленні адаптивного алгоритму підбору версій вебсторінок на підставі попередньої їх класифікації за критеріями показників продуктивності, що дає змогу забезпечити максимально швидке завантаження вебсторінок незалежно від типу пристрою чи швидкості інтернет-з'єднання.

Об'єкт дослідження – класифікація вебсторінок на підставі показників їх продуктивності.

Предмет дослідження – методи та засоби класифікації числових даних на підставі алгоритмів машинного навчання моделей без вчителя з елементами ансамблів, що дасть змогу класифікувати кількісні показники вебсторінок для подальшого формування правил підбору вебсторінок.

Мета роботи – розробити алгоритм підбору допус-

Інформація про авторів:

Пісун Олег Йосипович, канд. техн. наук, доцент, кафедра комп'ютерної інженерії. Email: o.pitsun@wunu.edu.ua;

<https://orcid.org/0000-0003-0280-8786>

Рудик Владислава Всеволодівна, магістрант, кафедра комп'ютерної інженерії. Email: vlada.rudyk.2015@gmail.com;

<https://orcid.org/0009-0007-9171-8008>

Батько Юрій Мирославович, канд. техн. наук, доцент, кафедра комп'ютерної інженерії. Email: bum@wunu.edu.ua;

<https://orcid.org/0000-0002-6732-4865>

Цитування за ДСТУ: Пісун О. Й., Рудик В. В., Батько Ю. М. Класифікація вебсайтів за критерієм їх продуктивності на підставі ансамблів. Науковий вісник НЛТУ України. 2025, т. 35, № 4. С. 180–185.

Citation APA: Pitsun, O. Yo., Rudyk, V. V., & Batko, Yu. M. (2025). Classification of websites by the criterion of productivity based on ensembles. *Scientific Bulletin of UNFU*, 35(4), 180–185. <https://doi.org/10.36930/40350421>

тих версій вебсторінок на підставі показників їхньої продуктивності та попередньо навчених даних, що дасть змогу реалізувати швидке відображення вмісту вебсторінки з урахуванням зовнішніх чинників.

Для досягнення зазначеної мети визначено такі основні завдання дослідження:

- проаналізувати алгоритми класифікації даних, зокрема – ансамблеві підходи до голосування, для отримання точнішого результату, що дасть змогу реалізувати класифікацію кількісних показників метрик оцінювання продуктивності вебсайтів та подальшого розроблення алгоритму вибору вебсторінок для пришвидшення їх завантаження;
- розробити узагальнений алгоритм вибору типу вебсторінки залежно від вхідних параметрів, що дасть змогу реалізувати швидке завантаження вебсайтів залежно від типу пристрою чи інтернет-з'єднання на підставі попередньої їх класифікації;
- провести порівняльний аналіз отриманих результатів для вибору найточнішого алгоритму та способу голосування в ансамблевих методах, які уможливають подання переліку кращих алгоритмів машинного навчання моделей для завдання класифікації кількісних показників метрик вимірювання продуктивності вебсайтів.

Аналіз останніх досліджень та публікацій. Дослідження [3] пропонує метод відбору ознак на підставі ансамблю фільтрів, який поєднує результати кількох методів для підвищення точності класифікації показників продуктивності роботи вебсайтів та зменшення розмірності даних. Ансамблеві методи є популярними для класифікації потоків даних завдяки високій продуктивності та простоті впровадження. Вони ефективні у поєднанні з алгоритмами виявлення дрейфу та динамічними оновленнями моделей. У роботі [21] автори пропонують таксономію понад 60 алгоритмів, описують ключові аспекти та окреслюють актуальні виклики у дослідженнях потоків даних.

У роботі [13] розроблено стійку процедуру класифікації вебсайтів на підставі ансамблів класифікаторів, кожен з яких будується з окремого підпростору предикторів, сформованого випадковим поділом повного набору ознак. Метод забезпечує покращену точність порівняно з відповідним поодиноким класифікатором. У дослідженні [16] проаналізовано 9 методів розширення даних і 9 ансамблевих алгоритмів, щоб виявити найкращі комбінації для покращення їх класифікації на незбалансованих наборах. Результати показали, що класичні методи, як SMOTE і випадкове повторне вибірконе збільшення, ефективніші та менш ресурсомісткі, ніж GAN.

У дослідженні [14] оцінено ефективність різних ансамблевих алгоритмів, зокрема AdaBoost, GBM, XGBoost, LightGBM та Random Forest, для класифікації супутникових даних дистанційного зондування: мультиспектральних, гіперспектральних та поляриметричних SAR-зображень. Узагальнений аналіз підходів до побудови ансамблевих методів розглянуто у роботах [18, 19, 23], які демонструють поширеність та актуальність застосування ансамблевого підходу для класифікації неструктурованих даних. Також даний підхід характеризується швидкістю роботи.

У дослідженні [17] вивчено вплив 11 параметрів на продуктивність e-commerce додатків. Результати показали, що деякі застосунки потребують доопрацювання, а їх можна класифікувати на три групи за часом реакції: низький, середній та високий. Дослідження [26] рекомендує виконувати кілька послідовних запусків (допус-

тимо 5) і використовувати медіану результатів для зменшення цієї варіативності. Це допомагає отримати надійніші показники продуктивності та уникнути помилок висновків за одноразового вимірювання. Дослідження [15] вивчає вплив комп'ютерних застосунків на ефективність кроскордонної електронної комерції, аналізуючи їх використання на різних етапах – від дослідження ринку до підтримки. Важливо приділити увагу інтелектуальним методам оброблення даних та алгоритмів машинного навчання моделей, які наведено у роботах [2, 24, 25]. Аналіз публікації в галузі продуктивності вебсайтів демонструє важливість проектування програмної системи з урахуванням вимог до часу завантаження та інших метрик. Також у роботах висвітлено підходи до класифікації сайтів за метриками оцінювання часу завантаження сторінок.

У роботі [22] висвітлено важливість правильного налаштування серверної частини та віртуальної машини Java для підвищення продуктивності вебсайтів.

У роботі [11] запропоновано високопродуктивний метод захисту вебзастосунків, однак не враховано впливу на швидкодію сайту з погляду користувача. Окремий акцент варто зробити на роботі [6, 7], які аналізують продуктивність функціонування сайтів, однак, орієнтуються на тільки серверній частині та швидкості синхронних та асинхронних запитів від клієнта до сервера.

Отже, останні дослідження демонструють актуальність застосування засобів штучного інтелекту для вирішення завдань підбору адаптивних версій вебсторінок на підставі показників спеціалізованих метрик. Проте наведені вище дослідження дотично акцентують свою увагу на комплексному використанні алгоритмів навчання без вчителя, зосереджуючись на аналізі статистичних показників продуктивності вебсайтів або використанні конкретного алгоритму без порівняння з іншими.

Матеріали та методи дослідження. Розроблений алгоритм підбору вебсторінок на підставі результатів класифікації показників продуктивності передбачає використання алгоритмів навчання моделі без вчителя, об'єднаних ансамблевим методом класифікації вебсайтів для виділення окремих класів вебсторінок, використання метрик для формування набору даних та подальшої класифікації. На підставі функціонального підходу до розроблення програмного забезпечення реалізовано програмний модуль для класифікації показників продуктивності вебсайтів.

Результати дослідження та їх обговорення / Research results and their discussion

Формування даних. GTmetrix та Google PageSpeed Insights є інструментами для комплексного оцінювання продуктивності вебсайтів, які аналізують швидкість завантаження сторінок та оптимізацію ресурсів. Приклад оцінювання показників вебсайту засобами Pagespeed наведено на рис. 1. На рисунку наведено тільки частину метрик оцінювання продуктивності сайтів, однак вони є найбільш ключовими і важливими. Основними прикладами вебсайтів є [8]: сайт університетів; сайти-блоги; інтернет-магазини;

До основних метрик, які використовувались для оцінювання сайтів, належать [9]:

- Cumulative Layout Shift – показник, який вимірює сумарний зсув елементів інтерфейсу під час завантаження;
- First Contentful Paint – показник, що відображає час від початку завантаження сторінки до етапу, коли браузер вперше відображає текст;

- Total Blocking Time – сумарний час, коли основний потік браузера був заблокований довгими завданнями, через що сторінка не могла швидко реагувати на взаємодії користувача;

- Speed Index – оцінювання швидкості відображення вмісту сторінки, що призводить до певних очікувань користувача, а також часто негативно впливає на покращення його досвіду.

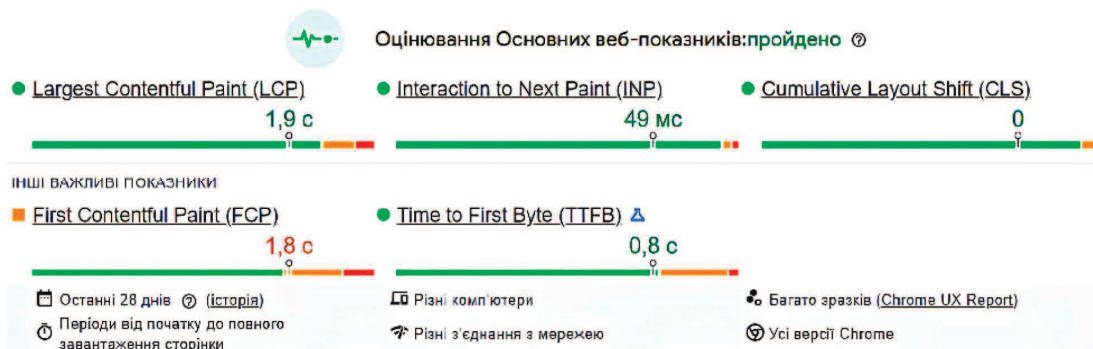


Рис. 1. Приклад оцінювання показників вебсайту засобами Pagespeed / Example of evaluating website performance using Pagespeed tools

Приклад сформованого набору даних наведено у табл. 1. Набір даних сформований автором шляхом тестування вебсайтів засобами Pagespeed.

Табл. 1. Приклад сформованого набору даних / Example of a generated dataset [1]

First Contentful Paint	Total Blocking Time	Largest Contentful Paint	Cumulative Layout Shift	Speed Index
1735	1070	4662	0	2573
1684	979	4650	0	2311
1850	2050	4057	0	2505
2096	780	2070	0	2773
2083	858	3001	0	2518

Після додаткового опрацювання, кожному запису додається окреме поле "status", який знаходиться в межах від 0...2, що характеризує ступінь завантаженості вебсайту. Візуалізацію параметра "Total Blocking Time" наведено на рис. 2.

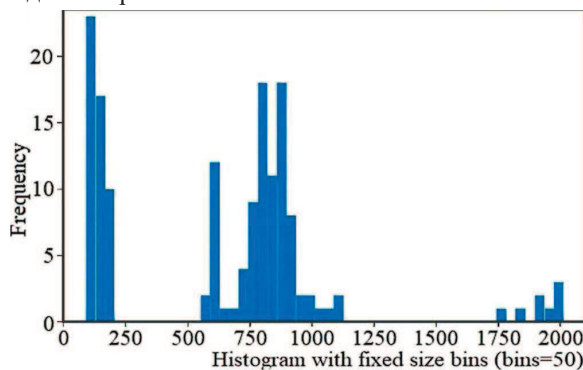


Рис. 2. Візуалізація параметра "Total Blocking Time" / "Total Blocking Time" Visualization

Візуалізацію параметра "Largest Contentful Paint" наведено на рис. 3.

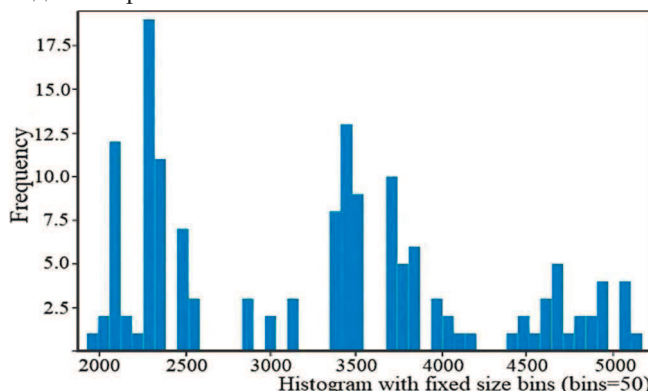


Рис. 3. Візуалізація параметра "Largest Contentful Paint" / "Largest Contentful Paint" Visualization

Кореляційну матрицю наведено на рис. 4. Кореляційну матрицю за класифікації показників продуктивності вебсайтів використовують для виявлення взаємозв'язків між ознаками та їх впливу на цільову змінну.

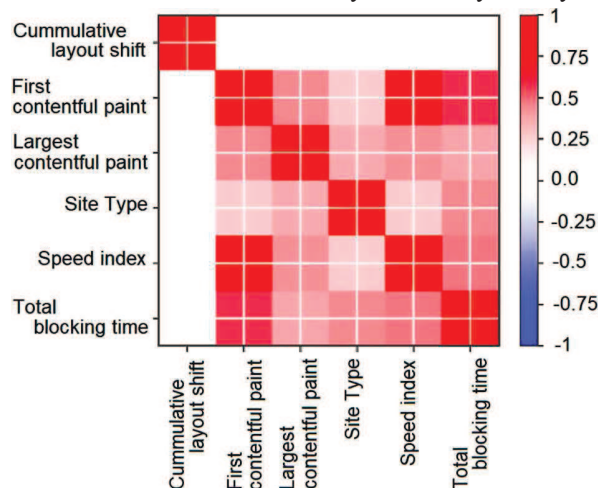


Рис. 4. Кореляційна матриця / Correlation matrix

Узагальнений алгоритм вибору типу сторінки залежно від вхідних параметрів наведено на рис. 5.

Алгоритм складається з таких кроків:

- 1) Визначення параметрів інтернет-з'єднання, типу пристрою.
- 2) Визначення роздільної здатності екрана.
- 3) Встановлення початкових значень у разі неможливості визначити зовнішні параметри.
- 4) Визначення типу вебсторінки залежно від зовнішніх параметрів на підставі попередньої класифікації показників продуктивності вебсайтів методом ансамблів. Як правило підбору вебсторінок, використано результати класифікації кількісних показників метрик продуктивності на підставі показників pagespeed, GTMetrix. Після оцінювання характеристик кожного нового сайту відбувається віднесення до одного з трьох класів "легка сторінка", "середня сторінка", "важка сторінка" та відповідно підвантаження необхідної версії сторінки. Результати класифікації вебсайтів продемонстровано у табл. 1. Приклад поділу на три класи на підставі попередньої класифікації показників вебсайтів можна сформулювати у вигляді таких правил:
 First Contentful Paint (FCP);
 Largest Contentful Paint (LCP);
 IF FCP > 2 c та LCP > 2.5 c THEN class 1 (Light version);
 IF (FCP > 0.7c & FCP < 2c) & (LCP > 1.0c & LCP < 2c) THEN class 2 (Medium version);
 IF FCP < 0.7c & LCP < 1.0c THEN class 3 (Full version).
- 5) Завантаження версії вебсторінки на підставі попередніх кроків.



Рис. 5. Узагальнений алгоритм вибору типу сторінки залежно від вхідних параметрів / Generalized algorithm for selecting page type depending on input parameters

Random forest – це ансамблевий метод машинного навчання множини моделей, який використовується для вирішення завдань класифікації даних, зокрема показників продуктивності вебсайтів, регресії та інших типів прогнозування. Його принцип роботи полягає у створенні множини дерев рішень під час тренування моделі, після чого для класифікації вебсайтів обирається клас на підставі більшості голосів (мода), а для регресії – розраховується середнє значення прогнозів усіх дерев.

Hard voting – це простий алгоритм, який можна використовувати для об'єднання прогнозів кількох класифікаторів. Простота жорсткого голосування робить його популярним вибором для побудови ансамблевих моделей машинного навчання. Жорстке голосування базується на більшості голосів окремих класифікаторів. Якщо дані шумні, ймовірніше, що більшість голосів буде неправильною.

Soft Voting – це алгоритм, який можна використовувати для об'єднання прогнозів кількох класифікаторів за допомогою ймовірності прогнозів. Алгоритм працює

так, що спочатку кожен класифікатор призначає ймовірність для кожного класу. Тоді прогноз ансамблю є просто класом із найвищою загальною ймовірністю.

Результати класифікації показників продуктивності вебсайтів, поділений за рівнем складності на три частини, наведено у табл. 2. Результати отримано експериментально.

Для виконання експериментів використано таке програмне забезпечення:

- мова програмування: Python;
- бібліотеки: numpy, pandas, keras;
- дві відеокарти NVIDIA Tesla T4. кожна з яких має 16 ГБ відеопам'яті.

Табл. 2. Порівняльний аналіз алгоритмів класифікації показників продуктивності вебсайтів / Comparative analysis of classification algorithms

Алгоритм	Навчальна точність	Тестова точність
RidgeClassifier	86.33	83.0
Logistic Regression	99.33	97.0
k-Nearest Neighbors	59.67	32.0
Stochastic Gradient	38.67	25.5
VotingClassifier-hard voting	99.33	98.0
VotingClassifier-soft voting	100.00	83.5

Аналізуючи дані у наведеній вище табл. 2, можна зробити висновок, що hard та soft VotingClassifier показують кращі результати порівняно з одиничним використанням загальновідомих алгоритмів та підходів. Такий підхід дає змогу якісніше класифікувати показники продуктивності вебсайтів та відносити їх до певної категорії для подальшого завантаження відповідно до типу пристрою, інтернет-з'єднання тощо.

Унаслідок тестування алгоритмів класифікації даних на підставі методу ансамблів, встановлено що VotingClassifier-hard voting показав результат 98 % на тестовій вибірці. Своєю чергою, VotingClassifier-soft voting показав кращий результат на навчальній вибірці, однак гірший на тестовій.

Обговорення результатів дослідження. Дедалі частіше під час розроблення вебсистем використовують елементи сторінок, які характеризуються зручним дизайном та високим навантаженням контенту одночасно. Розробникам необхідно розробляти різні версії одних і тих самих вебсторінок для різних пристроїв. Наявність адаптивного інтерфейсу не завжди забезпечує швидкість завантаження сторінок, тому щоразу частіше використовують підхід, де розробляють різні версії вебсторінок для завантаження на різних пристроях з різними розширеннями екрана. Важливо також висвітлити актуальність розроблення "легких" сторінок, які швидко завантажуються для забезпечення кращих показників SEO-удосконалення.

У роботі [22] описано програмне забезпечення, яке дає змогу аналізувати вплив різних конфігурацій JVM на багатофункціональний вебзастосунок. У цьому рішенні відбувається аналіз вебсайтів, що базуються на мові програмування Java. Недоліком є те, що не наведено аналізу швидкодії сайту у кінцевого користувача на різних пристроях.

У роботі [11] запропоновано високопродуктивний метод і фреймворк захисту вебзастосунків із мінімальним впливом на продуктивність системи. До недоліків можна віднести акцент на серверній частині, без особливого врахування клієнтської складової.

У дослідженні [7] запропоновано механізм false front для вебзастосунків, що відображає інтерактивний макет сторінки до повного завантаження. Експерименти показали, що цей підхід підвищує сприйняття швидкодії, скорочує час виконання завдань і покращує користувацький досвід порівняно з традиційними індикаторами завантаження. Однак у цій роботі неповною мірою проаналізовано показники метрик, зокрема щодо завантаження окремих блоків вебсторінок.

У роботі [6] розроблено високопродуктивну платформу на підставі Django для вебавтоматизації, що підтримує асинхронне оброблення 180 мс середнього часу відгуку. У цьому дослідженні акцент зроблено на удосконалення запитів до сервера, а не до швидкодії завантаження статичного контенту. У роботі [12] досліджують продуктивність та розробляють набір тестів для оцінювання продуктивності у хмарних сервісах. У роботі оцінюють завантаженість середовища у різних часових проміжках, не аналізуючи фронтенд-частину.

У дослідженні [10] оцінено вплив використання п'яти шаблонів проектування React та Vue.js на споживання пам'яті, використання CPU та часу рендерингу. Акцент у цьому дослідженні зроблено на удосконалення фронтенд-складової для підвищення продуктивності, що вказує актуальність дослідження. У роботі [4] запропоновано автоматизовану стратегію удосконалення продуктивності фронтенд-фреймворків на підставі байєсової оптимізації, що підтверджує актуальність застосування алгоритмів для покращення та удосконалення вебсайтів. У дослідженні [20] показано, що використання алгоритмів підкріплювального навчання істотно підвищує продуктивність вебзастосунків, зменшуючи затримку, споживання CPU та пам'яті.

У цій роботі запропоновано рішення для можливості підбору різних версій вебсторінок залежно від типу пристрою на підставі використання ансамблевого підходу для класифікації вебсайтів.

За результатами наведеного вище аналізу, можна зробити висновок, що завдання удосконалення навантаження контенту та підвищення продуктивності вебсторінок є актуальним завданням і викликом, який постає перед розробниками програмного забезпечення. У попередніх роботах автори приділяють більше уваги серверному удосконаленню, однак досліджують можливість удосконалення фронтенду частини вебсайту. Зокрема, основна увага спрямована на удосконалення фронтенд-фреймворків. Щодо оцінювання метрик навантаження контенту, то у роботах трапляється здебільшого статистичний аналіз показників без чіткого переліку пропозицій із покращення швидкодії. Також останнім часом автори використовують алгоритми машинного навчання моделей з та без вчителя для аналізу кількісних показників метрик продуктивності. У цій роботі запропоновано алгоритм, який використовує декілька алгоритмів машинного навчання без вчителя і на підставі ансамблевого підходу дає змогу забезпечити найкращий результат. Класифікація кількісних показників продуктивності вебсайтів необхідна для подальшої можливості підбору різних версій вебсторінок за ступенем їхньої складності залежно від зовнішніх чинників, що дасть змогу забезпечити швидке завантаження вебсторінок.

Отже, внаслідок виконаної роботи можна сформулювати такі наукову новизну та практичну значущість результатів дослідження.

Наукова новизна отриманих результатів дослідження – удосконалено методику підбору версій вебсторінок на підставі попередньо класифікованих даних, у якій, на відміну від наявної, використано ансамблевий підхід до підвищення точності їх класифікації, що дає змогу відображати максимально швидку версію вебсторінки залежно від зовнішніх факторів.

Практична значущість результатів дослідження – запропоновану методику можна використовувати як складову частину високонавантажених вебсайтів для забезпечення можливості відображення швидкої та адаптивної версії вебсторінки.

Висновки / Conclusions

Розроблено алгоритм підбору допустимих версій вебсторінок на підставі показників їхньої продуктивності та попередньо навчених даних, що дає змогу реалізувати швидке відображення вмісту вебсторінки з урахуванням зовнішніх чинників. За результатами проведеного дослідження можна зробити такі основні висновки.

1. Проаналізовано наявні підходи до вимірювання продуктивності вебсайтів для можливості подальшого формування набору даних показників сайтів, що дало можливість виділити перелік ключових параметрів оцінювання продуктивності сайтів для подальшої їх класифікації.
2. Проаналізовано алгоритми класифікації даних, зокрема ансамблеві підходи до голосування для отримання якіснішого результату. Класифікація проводилась на вибірці показників метрик продуктивності вебсайтів.
3. Розроблено узагальнений алгоритм вибору типу сторінки залежно від вхідних параметрів, що дає змогу автоматично підбирати версії вебсторінок відповідно до умов. Порівняльний аналіз результатів показав точність у 98 % на тестовій вибірці для алгоритму hard voting. Використання цього алгоритму у вебсайтах дає змогу підбирати адаптивніші версії сайтів на підставі показників продуктивності.

References

1. A dataset of website performance metrics. URL: <https://www.kaggle.com/datasets/olkoooleh/site-performance4>
2. Berezhsky, O. M., Pitsun, O. Yo., Melnyk, G. M., & Datsko, T. V. (2021). Application of linear regression method for analysis of cytological images quantitative characteristics. *Ukrainian Journal of Information Technology*, 3(1), 73–77. <https://doi.org/10.23939/ujit2021.03.073>
3. Bolón-Canedo, V., Sánchez-Marono, N., & Alonso-Betanzos, A. (2014). Data classification using an ensemble of filters. *Neurocomputing*, 135, 13–20. <https://doi.org/10.1016/j.neucom.2013.03.067>
4. Chen, C., Meng, Q., & Huang, J. (2024). Performance Optimization of Web Front-End Frameworks: Automatic Adjustment Strategies Based on Bayesian Optimization Algorithm. In *Proceedings of the 2024 International Conference on Machine Intelligence and Digital Applications*, 523–528. <https://doi.org/10.1145/3662739.3672180>
5. Dasari, A. K., Biswas, S. K., Thounaojam, D. M., Devi, D., & Purkayastha, B. (2023). Ensemble Learning Techniques and Their Applications: Advances in Cognitive Science and Communications. ICCCE 2023. *Cognitive Science and Technology*. Springer, Singapore, pp. 897–912. https://doi.org/10.1007/978-981-19-8086-2_85
6. Du, J. (2024). Programming and Development of Web Automation Application Platform Based on Django. In *Proceedings of the 2024 2nd International Conference on Artificial Intelligence, Systems and Network Security*, pp. 197–200. <https://doi.org/10.1145/3714334.3714368>
7. Dufresne, T., Gutwin, C., & Graham, T. N. (2025). Fake it till you load it: User Perceptions and Performance with Fast-Loading "False Front" Web Pages. *Proceedings of the ACM on Human-*

- Computer Interaction*, 9(4), 1–41. <https://doi.org/10.1145/3735593>
8. Espinosa-Leal, L., Akusok, A., Lendasse, A., & Björk, K. M. (2019). Website classification from webpage renders. In *International Conference on Extreme Learning Machine*, pp. 41–50. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-58989-9_5
 9. Ghaffar, A., Abid, F., Ashraf, M., Jamil, A., Abbas, A., & Malik, F. R. (2023). Identification of an Optimized Google PageSpeed Audit-Rule-Sequence to Optimize Page Speed. *VAKKUM Transactions on Computer Sciences*, 11(1), 123–137. <https://doi.org/10.21015/vtcs.v11i1.1329>
 10. Gomes, F., Soares, P., & Araújo, A. A. (2023). Examining the Performance Implications of Design Patterns in Front-end Web Development: A Preliminary Comparative Study with React and Vue.js. In *Proceedings of the 29th Brazilian Symposium on Multimedia and the Web*, 255–259. <https://doi.org/10.1145/3617023.3617057>
 11. He, C., & Ding, C. H. (2025). SecRASP: Next generation web application security protection methodology and framework. *Computers & Security*, Vol. 154, article ID 104445. <https://doi.org/10.1016/j.cose.2025.104445>
 12. Henning, S., Vogel, A., Perez-Wohlfeil, E., Ertl, O., & Rabiser, R. (2025). When Should I Run My Application Benchmark? Studying Cloud Performance Variability for the Case of Stream Processing Applications. In *Proceedings of the 33rd ACM International Conference on the Foundations of Software Engineering*, 400–410. <https://doi.org/10.1145/3696630.372856>
 13. Hongshik, A., Fazzari, M. J., Lim, N., Chen, J. J., & Kodell, R. L. (2007). Classification by ensembles from random partitions of high-dimensional data. *Computational Statistics & Data Analysis*, 51(12), 6166–6179. <https://doi.org/10.1016/j.csda.2006.12.043>
 14. Jafarzadeh, H., Mahdianpari, M., Gill, E., Mohammadimanesh, F., & Homayouni, S. (2021). Bagging and Boosting Ensemble Classifiers for Classification of Multispectral, Hyperspectral and PolSAR Data: A Comparative Evaluation. *Remote Sensing*, 13(21), article ID 4405. <https://doi.org/10.3390/rs13214405>
 15. Jin, L., & Chen, L. (2024). Exploring the Impact of Computer Applications on Cross-Border E-Commerce Performance. In *IEEE Access*, 12, 74861–74871. <https://doi.org/10.1109/ACCESS.2024.3385017>
 16. Khan, A. A., Chaudhari, O., & Chandra, R. (2024). A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation. *Expert Systems with Applications*, 244, article ID 122778. <https://doi.org/10.1016/j.eswa.2023.122778>
 17. Mahfida, A., Hossain, T., Hassan, R., & Rahman, A. (2021). Web application performance analysis of E-commerce sites in Bangladesh: an empirical study. *International Journal of Information Engineering and Electronic Business*, 13(2), 47–54. <https://doi.org/10.5815/ijieeb.2021.02.04>
 18. Majid, T., Udzir, N. I., Abdullah, M. T., & Razali, Y. (2021). International Journal of Advanced Computer Science and Applications. *West Yorkshire*, 12(5), article ID 2021. <https://doi.org/10.14569/IJACSA.2021.0120566>
 19. Mienye, I. D., & Sun, Y. (2022). A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects". In *IEEE Access*, 10, pp. 99129–99149. <https://doi.org/10.1109/ACCESS.2022.3207287>
 20. Moghaddasi, K., & Rajabi, S. (2023). Learning at the Edge: Mobile Edge Computing and Reinforcement Learning for Enhanced Web Application Performance. 9th *International Conference on Web Research (ICWR)*, pp. 300–304. <https://doi.org/10.1109/ICWR57742.2023.10138952>
 21. Murilo, G., Barddal, L. P., Enembreck, F., & Bifet, A. (2017). A survey on ensemble learning for data stream classification. *ACM Computing Surveys (CSUR)*, 50(2), 1–36. <https://doi.org/10.1145/305492>
 22. Noetzold, D., de Moraes Rossetto, A. G., Silva, L. A., Crocker, P., & Leithardt, V. R. Q. (2024). JVM optimization: An empirical analysis of JVM configurations for enhanced web application performance. *SoftwareX*, 28, article ID 101933. <https://doi.org/10.1016/j.softx.2024.101933>
 23. Palak, M., Uddin, S., Hajati, F., & Moni, M. A. (2023). Ensemble Learning for Disease Prediction: A Review. *Healthcare*, 11(12), article ID 1808. <https://doi.org/10.3390/healthcare11121808>
 24. Pitsun, O. (2024). Comparative analysis of stress factors of humanities and technical specialties students. *CEUR Workshop Proceedings Conference*, 46–55. URL: <https://ceur-ws.org/Vol-3716/paper6.pdf>
 25. Pitsun, O. (2025). Comparative analysis of the value-semantic sphere of Ukrainian volunteers using AI. *CEUR Workshop Proceedings Conference*, 225–234. URL: <https://ceur-ws.org/Vol-3974/short06.pdf>
 26. Tjaša, H., Šumak, B., & Brdnik S. (2021). Towards representative web performance measurements with google lighthouse". In *Proceedings of the 2021 7th student computer science research conference (StuCoSReC)*, pp. 39–42. <https://doi.org/10.18690/978-961-286-516-0-9>
 27. Yuzhen, Z., Liu, J., & Shen, W. (2022). A Review of Ensemble Learning Algorithms Used in Remote Sensing Applications. *Applied Sciences*, 12(17), article ID 8654. <https://doi.org/10.3390/app12178654>

O. Yo. Pitsun, V. V. Rudyk, Yu. M. Batko

Western Ukrainian National University, Ternopil, Ukraine

CLASSIFICATION OF WEBSITES BY THE CRITERION OF PRODUCTIVITY BASED ON ENSEMBLES

Approaches to measuring the productivity of websites and their classification with subsequent division into categories are considered. It was established that among modern algorithms for data classification, ensemble methods are distinguished, in particular soft and hard voting to obtain the best result. The main evaluation criteria are: Page Load Time, Time To First Byte, Server Response Time, First Contentful Paint, Speed Index. Search algorithms, in particular Google, prefer sites with high speed and smooth interaction, which increases their visibility in search results. In addition, optimal performance reduces the bounce rate and improves conversion, which is critical for successful online promotion. The impact of indicators of the main parameters of websites on their loading speed and, accordingly, classification into 3 categories was assessed. Evaluating the performance of websites is a relevant task, where loading speed and display stability significantly affect the user experience and conversion. The use of ensemble methods in classification is based on the principle of combining multiple models or data sources to increase the overall accuracy and stability of predictions. Ensembles allow you to reduce the impact of random errors of individual models, eliminate bias and reduce the dispersion of results. By integrating heterogeneous characteristics or features, ensemble data provides a more comprehensive view of the classification object, which is especially important when working with heterogeneous or structurally complex data sets. Effective performance assessment and analysis allows you to identify problem areas in the sites operation, optimize resources and ensure uninterrupted access for users. Based on the experimental approach, the best algorithms for unsupervised learning were identified, which allowed developing an ensemble method for classifying websites and based on their performance indicators. The accuracy was 98 % on the test sample. The availability of such a solution allows selecting the optimal versions of web pages for a specific type of web pages.

Keywords: data analysis; logistic regression; nearest neighbor method; pagespeed; GTmetrics.