

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

Кафедра «Системний аналіз та обчислювальна математика»

Пояснювальна записка

до дипломної роботи

магістра

(ступінь вищої освіти)

на тему: ML МЕТОДИ В ЗАДАЧІ КЛАСИФІКАЦІЇ БЕЗПІЛОТНИХ
ЛІТАЛЬНИХ АПАРАТІВ

Виконав(ла): студентка 2 курсу,
групи КНТз-813м

Спеціальності 124 – «Системний аналіз»
(код і найменування спеціальності)

Освітня програма
«Інтелектуальні технології та прийняття
рішень в складних системах»

ТКАЧ Д.В.

(ПРИЗВИЩЕ та ініціали)

Керівник ШИРОКОРАД Д.В.

(ПРИЗВИЩЕ та ініціали)

Рецензент ДУМІН О.М.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

Кафедра «Системний аналіз та обчислювальна математика»

Ступінь вищої освіти магістр

Спеціальність 124 – «Системний аналіз»

(код і найменування)

Освітня програма «Інтелектуальні технології та прийняття рішень в складних системах»

(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

В. о. завідувача кафедри Терещенко Е.В.

« 23 » грудня 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНА РОБОТА СТУДЕНТКИ

ТКАЧ Дар'я Володимирівна

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема роботи ML МЕТОДИ В ЗАДАЧІ КЛАСИФІКАЦІЇ БЕЗПЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ

керівник роботи к.ф.-м.н., доцент ШИРОКОРАД Дмитро Вікторович

(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від «20» листопада 2024 року №480

2. Строк подання студентом роботи 20 грудня 2024 р.

3. Вихідні дані до роботи: набір даних з технічними характеристиками безпілотних літальних апаратів з відкритого ресурсу Dataverse, методи машинного навчання та бібліотеки Python для їх реалізації.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): огляд літератури, аналіз сучасного стану проблеми класифікації БПЛА, аналіз типів та характеристик БПЛА, огляд існуючих методів класифікації та застосування методів машинного навчання; методологія дослідження з характеристикою набору даних, вибір ключових технічних параметрів, обробка даних; експериментальне дослідження з використанням різних методів машинного навчання, порівняльний аналіз результатів класифікації, оцінка точності та аналіз помилок класифікації БПЛА, дослідження вплив технічних характеристик на класифікацію БПЛА.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів): презентація на 16 слайдах (Актуальність, Проблематика, Огляд існуючих рішень, Мета та завдання, Набір даних, Попередня обробка даних, Кореляційний аналіз, Методологія дослідження, Логістична регресія, Random Forest, XGBoost, Нейронна мережа, Порівняльний аналіз, Практичне значення, Висновки).

6. Консультанти розділів проєкту (роботи)

| Розділ | ПРИЗВИЩЕ, ініціали та посада консультанта | Підпис, дата | |
|---------------|---|----------------|---------------------------|
| | | завдання видав | прийняв виконане завдання |
| 1 | ШИРОКОРАД Д.В., доцент | 28.10.2024 | 20.11.2024 |
| 2 | ШИРОКОРАД Д.В., доцент | 20.11.2024 | 01.12.2024 |
| 3 | ШИРОКОРАД Д.В., доцент | 01.12.2024 | 20.12.2024 |
| Нормоконтроль | ШИРОКОРАД Д.В., доцент | 20.12.2024 | 20.12.2024 |
| | | | |
| | | | |
| | | | |

7. Дата видачі завдання «28» жовтня 2024 року.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломної роботи | Строк виконання етапів роботи | Примітка |
|-------|---|-------------------------------|----------|
| 1 | Аналіз літературних джерел та огляд існуючих методів класифікації БПЛА | 28.10 - 08.11.2024 | |
| 2 | Підготовка першого розділу роботи, огляд характеристик БПЛА та методів машинного навчання | 09.11 - 20.11.2024 | |
| 3 | Формування набору даних, вибір параметрів та попередня обробка даних | 21.11 - 30.11.2024 | |
| 4 | Розробка та реалізація моделей машинного навчання, проведення експериментів | 01.12 - 08.12.2024 | |
| 5 | Аналіз результатів, порівняння ефективності методів та оформлення висновків | 09.12 - 14.12.2024 | |
| 6 | Оформлення пояснювальної записки та підготовка презентації | 15.12 - 20.12.2024 | |

Студент(ка)

_____ Дар'я ТКАЧ
(підпис) (Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

_____ Дмитро ШИРОКОРАД
(підпис) (Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

ПЗ: 75 с., 14 рис., 10 табл., 18 джерел, 1 додаток.

Об'єкт дослідження – безпілотні літальні апарати, які використовуються у військових та цивільних цілях.

Мета роботи – розробка та вдосконалення методів класифікації безпілотних літальних апаратів на основі машинного навчання для забезпечення високої точності та надійності в умовах різноманітних сценаріїв використання.

Методи дослідження – логістична регресія, Random Forest, XGBoost та нейронні мережі для класифікації; методи попередньої обробки даних включають балансування класів, нормалізацію ознак та вибір важливих характеристик.

Актуальність проблеми зумовлена широким розповсюдженням БПЛА та необхідністю їх ефективної класифікації для забезпечення безпеки повітряного простору, особливо в умовах військових конфліктів. В результаті дослідження розроблено систему класифікації БПЛА на основі машинного навчання, яка забезпечує точність до 100% при використанні ансамблевих методів. Визначено ключові технічні характеристики, що найбільше впливають на класифікацію БПЛА. Новизна роботи полягає у комплексному порівняльному аналізі ефективності різних методів машинного навчання для класифікації БПЛА. Результати роботи можуть бути використані для автоматизованої класифікації нових зразків БПЛА, оптимізації процесу їх розробки та вдосконалення систем контролю повітряного простору.

БЕЗПІЛОТНІ ЛІТАЛЬНІ АПАРАТИ, МАШИННЕ НАВЧАННЯ,
КЛАСИФІКАЦІЯ, RANDOM FOREST, XGBOOST, НЕЙРОННІ МЕРЕЖІ,
ТЕХНІЧНІ ХАРАКТЕРИСТИКИ, ОБРОБКА ДАНИХ, ТОЧНІСТЬ
КЛАСИФІКАЦІЇ

ABSTRACT

ML METHODS IN UNMANNED AERIAL VEHICLE
CLASSIFICATION TASK

Master's thesis: 75 pages, 14 figures, 10 tables, 18 references, 1 appendix.

Research object: unmanned aerial vehicles (UAVs) used in military and civilian applications.

Research subject: machine learning methods for UAV classification based on their technical characteristics.

Research purpose: development and improvement of machine learning methods for UAV classification to ensure high accuracy and reliability across various usage scenarios.

Research Methods and Tools: The study employed multiple machine learning approaches including Logistic Regression as a baseline method, Random Forest and XGBoost as ensemble methods, and Neural Networks for complex pattern recognition. Data preprocessing techniques included class balancing using RandomOverSampler, feature normalization using StandardScaler, and feature importance analysis.

Research Relevance: The widespread adoption of UAVs in both civilian and military sectors has created an urgent need for reliable classification systems. This is particularly crucial in military contexts, where accurate UAV classification directly impacts tactical decision-making and defense capabilities. The Russian-Ukrainian war has demonstrated the strategic importance of UAVs and the critical need for their effective classification and identification.

Main Results: The research analyzed a dataset of 190 UAV models with various technical parameters and identified key characteristics for classification including MTOW, payload capacity, wingspan, cruise speed, and operational altitude. A comprehensive data preprocessing pipeline was developed, achieving 100% missing value handling. Model evaluation showed that Logistic Regression

achieved 85.9% accuracy after data balancing, while Random Forest and XGBoost demonstrated superior performance with 100% accuracy. The Neural Network achieved 91.11% accuracy with stable learning dynamics. A thorough comparison of models was conducted using accuracy, precision, recall, and ROC-AUC metrics.

Feature importance analysis revealed Maximum Take-Off Weight (MTOW) as the most crucial characteristic, with payload capacity and cruise speed showing significant impact on classification. Strong correlations were discovered between various technical parameters. The research demonstrated that ensemble methods consistently outperform traditional approaches, while data balancing significantly improves classification accuracy across all models.

Practical Applications: The developed classification system can be implemented in automated classification of new UAV models; military defense systems for rapid UAV identification; regulatory compliance assessment; and UAV development and optimization. Future research directions include integration of real-time classification capabilities; extension to multi-sensor data fusion; development of specialized models for specific military applications; and investigation of adversarial scenarios and robustness improvement.

The research demonstrates that machine learning methods, particularly ensemble approaches, can achieve highly accurate UAV classification based on technical characteristics. The developed methodology provides a robust foundation for practical applications in both civilian and military contexts.

KEYWORDS: UNMANNED AERIAL VEHICLES, MACHINE LEARNING, CLASSIFICATION, RANDOM FOREST, XGBOOST, NEURAL NETWORKS, TECHNICAL CHARACTERISTICS, DATA PROCESSING, CLASSIFICATION ACCURACY, MILITARY APPLICATIONS

ЗМІСТ

| | |
|---|----|
| Завдання | 2 |
| Реферат | 4 |
| Abstract | 5 |
| Вступ..... | 9 |
| 1 Огляд літератури та сучасний стан проблеми..... | 11 |
| 1.1 Огляд типів безпілотних літальних апаратів та їх класифікація..... | 11 |
| 1.2 Застосування методів машинного навчання для класифікації безпілотних літальних апаратів | 20 |
| 1.2.1 Класифікація БПЛА на основі візуальних даних..... | 20 |
| 1.2.2 Класифікація БПЛА на основі радарних даних | 23 |
| 1.2.3 Класифікація на основі акустичних сигналів..... | 25 |
| 1.2.4 Класифікація на основі радіочастотних характеристик..... | 27 |
| 1.2.5 Комплексні системи класифікації | 29 |
| 1.2.6 Машинне навчання в дизайні дронів | 30 |
| 1.2.7 Особливості реалізації методів машинного навчання для класифікації БПЛА | 32 |
| 1.2.8 Оцінка ефективності методів класифікації | 34 |
| 1.3 Характеристика методів машинного навчання, що застосовуються при класифікації безпілотних літальних апаратів..... | 37 |
| 1.3.1 Логістична регресія..... | 37 |
| 1.3.2 Random Forest | 38 |
| 1.3.3 XGBoost..... | 39 |
| 1.3.4 Нейронні мережі..... | 41 |
| 1.3.5 Оцінка адекватності моделей..... | 42 |
| 2 Методологія дослідження | 44 |
| 2.1 Характеристика набору даних | 44 |
| 2.2 Вибір ключових технічних параметрів для класифікації | 46 |
| 2.3 Попередня обробка даних | 49 |
| 2.4 Розділення даних на тренувальну та тестову вибірки..... | 50 |
| 3 Результати дослідження та їх обговорення | 52 |

| | | |
|-------|---|----|
| 3.1 | Методологія моделювання та оцінювання класифікаторів | 52 |
| 3.1.1 | Вибір моделей машинного навчання | 52 |
| 3.1.2 | Метрики оцінювання моделей | 53 |
| 3.1.3 | Оптимізація параметрів моделей..... | 53 |
| 3.1.4 | Аналіз важливості ознак..... | 54 |
| 3.2 | Порівняльний аналіз результатів класифікації | 54 |
| 3.2.1 | Аналіз результатів логістичної регресії..... | 55 |
| 3.2.2 | Оцінка ефективності Random Forest | 56 |
| 3.2.3 | Результати застосування XGBoost | 58 |
| 3.2.4 | Аналіз роботи нейронної мережі | 59 |
| 3.2.5 | Порівняльна характеристика моделей | 60 |
| 3.3 | Оцінка точності та комплексний аналіз помилок класифікації БПЛА | 64 |
| 2.7 | Вплив технічних характеристик на класифікацію БПЛА..... | 66 |
| | Висновки | 70 |
| | Перелік посилань..... | 72 |
| | Додаток 1..... | 75 |

ВСТУП

В останні роки безпілотні літальні апарати (БПЛА) стали невід'ємною частиною сучасного світу, їх використання розширюється як в цивільній, так і у військовій сферах. Цей феномен пояснюється швидким розвитком технологій, таких як мініатюризація електронних компонентів та удосконалення систем навігації. Однак широке розповсюдження БПЛА піднімає питання безпеки, приватності, а також викликів, пов'язаних з класифікацією та детекцією цих апаратів. Особливо актуальним є застосування методів машинного навчання (ML) для вирішення задачі класифікації БПЛА, що може підвищити ефективність їх використання та регулювання. На прикладі війни Україні проти росії видно, що безпілотники змінюють характер війни, забезпечуючи розвідку, цілевказання та навіть атаку, що підкреслює необхідність удосконалення методів їх виявлення та класифікації для мінімізації ризиків і загроз для цивільних осіб і критичної інфраструктури.

Окрім того, важливість класифікації БПЛА зростає під час розробки нових моделей з заданими характеристиками та функціями. Класифікація дозволяє ефективно обирати технічні рішення, що найкраще відповідають конкретним вимогам і завданням. Таким чином, впровадження ML-методів для класифікації є необхідним не тільки для забезпечення безпеки та ефективності використання існуючих БПЛА, але й для оптимізації процесу їх проєктування та розробки. Це дозволить створювати дрони з заданими параметрами, які здатні виконувати спеціалізовані місії, з максимальною ефективністю та мінімальними витратами.

Метою цього дослідження є розробка та вдосконалення методів класифікації безпілотних літальних апаратів на основі машинного навчання для забезпечення високої точності та надійності в умовах різноманітних сценаріїв використання. Для досягнення мети було поставлено такі завдання:

- аналіз сучасних методів класифікації БПЛА, зокрема на основі глибокого навчання, а також їхніх переваг і недоліків.

- розробка та оптимізація моделей машинного навчання для класифікації різних типів безпілотних літальних апаратів.

- експериментальна оцінка запропонованих методів на реальних даних, з урахуванням факторів, які можуть впливати на точність класифікації.

Об'єктом дослідження є безпілотні літальні апарати, які використовуються у військових та цивільних цілях. Предметом дослідження є методи класифікації безпілотних літальних апаратів, засновані на технологіях машинного навчання.

Для досягнення поставленої мети використовувались методи машинного навчання, зокрема логістична регресія, Random Forest, XGBoost та нейронні мережі, що демонструють високі результати в задачах класифікації. Використовувались також методи попередньої обробки даних, такі як балансування класів за допомогою Random Oversampler, нормалізація ознак та вибір важливих ознак для підвищення точності та стабільності роботи моделей. Застосування цих підходів дозволило отримати надійні результати класифікації безпілотних літальних апаратів за їх технічними параметрами.

1 ОГЛЯД ЛІТЕРАТУРИ ТА СУЧАСНИЙ СТАН ПРОБЛЕМИ

1.1 Огляд типів безпілотних літальних апаратів та їх класифікація

Безпілотні літальні апарати (БПЛА) стали невід'ємною частиною як військової, так і цивільної авіації. За останні десятиліття їх розвиток пройшов шлях від простих дистанційно керованих моделей до складних автономних систем, здатних виконувати широкий спектр завдань.

Основні типи БПЛА розрізняються за конструктивними особливостями. Літакового типу (fixed-wing) БПЛА характеризуються високою тривалістю польоту, значною дальністю та швидкістю. Вони ефективні для задач спостереження та моніторингу великих територій, але потребують злітно-посадкової смуги. Гелікоптерного типу (rotary-wing) БПЛА мають перевагу у маневреності та можливості зависання на місці, що робить їх незамінними для інспекції об'єктів та роботи в обмеженому просторі. Мультироторні БПЛА, найпопулярніші серед яких квадрокоптери, поєднують простоту керування з високою стабільністю польоту [Помилка! Джерело посилання не знайдено.].

Особливу категорію становлять гібридні БПЛА з можливістю вертикального зльоту і посадки (VTOL), які намагаються поєднати переваги літакової схеми з можливостями гелікоптерів. Такі апарати особливо цінні для військових операцій, де не завжди є доступ до підготовлених злітно-посадкових майданчиків.

За розмірами та масою БПЛА можна розділити на декілька категорій. Мікро та нано БПЛА масою до 1 кг здатні виконувати завдання в обмежених просторах та становлять мінімальну загрозу при аваріях. Міні-БПЛА (до 25 кг) широко використовуються як у цивільній, так і у військовій сфері для тактичних завдань. Середні БПЛА (до 150 кг) та важкі (понад 150 кг) застосовуються для більш складних місій, що вимагають тривалого польоту

або значного корисного навантаження [**Помилка! Джерело посилання не знайдено.**].

Важливим аспектом класифікації є висота польоту та тривалість місії. У таблиці 1.1 наведено основні категорії БПЛА за цими параметрами.

Таблиця 1.1 – Класифікація БПЛА за висотою та тривалістю польоту [**Помилка! Джерело посилання не знайдено.**]

| Категорія | Висота польоту | Тривалість | Типове застосування |
|-----------|----------------|------------|----------------------|
| HALE | >15000 м | >48 год | Стратегічна розвідка |
| MALE | 5000–15000 м | 24–48 год | Тактична розвідка |
| LALE | 3000–5000 м | 12–24 год | Патрулювання |
| LASE | <3000 м | <12 год | Локальні операції |

Ключовою тенденцією розвитку БПЛА є підвищення рівня їх автономності. За цим критерієм виділяють п'ять рівнів – від повністю дистанційного керування до повної автономності з використанням штучного інтелекту для прийняття рішень. Ця класифікація особливо важлива для розробки нормативної бази та систем сертифікації БПЛА.

В останні роки спостерігається значне розширення сфер застосування БПЛА, що вимагає більш гнучких підходів до їх класифікації. Згідно з дослідженням [**Помилка! Джерело посилання не знайдено.**], важливо враховувати не тільки технічні характеристики, але й функціональні можливості та потенційні ризики використання БПЛА.

Важливим аспектом є класифікація за рівнем автономності систем керування. Згідно з [**Помилка! Джерело посилання не знайдено.**], виділяють такі основні типи систем:

- лінійні системи керування (PID-контролери, LQR контролери) – забезпечують базовий рівень автоматизації та стабілізації польоту.
- нелінійні системи керування (адаптивне керування, предиктивне керування) – дозволяють БПЛА адаптуватися до змін умов польоту.

- інтелектуальні системи керування – використовують методи штучного інтелекту для прийняття рішень.

Особливу увагу варто приділити новій комплексній системі класифікації, запропонованій в [Помилка! Джерело посилання не знайдено.], яка враховує як технічні характеристики, так і експлуатаційні ризики. Ця система представлена в таблиці 1.2.

Таблиця 1.2 – Комплексна система класифікації БПЛА за ризиками та вимогами до експлуатації

| Клас | Маса (кг) | Висота польоту (м) | Рівень ризика | Вимоги до сертифікації |
|------|--------------|-----------------------|------------------|---------------------------|
| 0 | <1 | <120 | Мінімальний | Не потребує |
| 1 | 1–4 | <120 | Низький | Базова реєстрація |
| 2 | 4–25 | <400 | Середній | Розширена сертифікація |
| 3 | 25–150 | <3000 | Підвищений | Повна сертифікація |
| 4 | >150 | >3000 | Високий | Сертифікація як ПС |

Ця система має ряд переваг: враховує взаємозв'язок між масою та потенційним ризиком; включає експлуатаційні обмеження; визначає чіткі вимоги до сертифікації; легко адаптується до нових технологій.

Сучасні тенденції розвитку БПЛА вказують на необхідність врахування нових факторів при класифікації. За даними [Помилка! Джерело посилання не знайдено.], особливу увагу слід приділяти: інтеграції систем штучного інтелекту; розвитку роєвих технологій; підвищенню рівня автономності; кібербезпеці; екологічним аспектам.

Аналіз різних підходів до класифікації показує, що найбільш ефективними є гібридні системи, які поєднують кількісні та якісні характеристики. Це дозволяє врахувати як технічні параметри, так і особливості застосування БПЛА.

Перспективи розвитку систем класифікації БПЛА тісно пов'язані з технологічним прогресом та новими викликами у сфері безпеки. Як зазначено в [Помилка! Джерело посилання не знайдено.], сучасні БПЛА стають все більш складними системами, що інтегрують передові технології та потребують відповідного регулювання.

Одним з ключових трендів є розвиток розумних технологій. За даними досліджень [Помилка! Джерело посилання не знайдено.], рої БПЛА можуть складатися з десятків або сотень апаратів, що діють як єдина система. Це створює нові виклики для класифікації, оскільки традиційні підходи, орієнтовані на окремі апарати, стають недостатніми.

У військовій сфері особливу увагу приділяють класифікації за бойовими можливостями. Згідно з [Помилка! Джерело посилання не знайдено.], сучасні БПЛА можна розділити на наступні категорії за їх бойовим застосуванням: розвідувальні; ударні; радіоелектронної боротьби; багатоцільові.

Важливим аспектом є інтеграція систем штучного інтелекту. У роботі [Помилка! Джерело посилання не знайдено.] пропонується класифікація за рівнем інтелектуалізації:

- базовий рівень (автоматичне керування)
- середній рівень (елементи ШІ для навігації та уникнення перешкод)
- високий рівень (повністю автономне прийняття рішень)
- експертний рівень (здатність до самонавчання та адаптації)

Ця класифікація відображається у зростаючій складності систем керування БПЛА, що можна побачити на рисунку 1.1

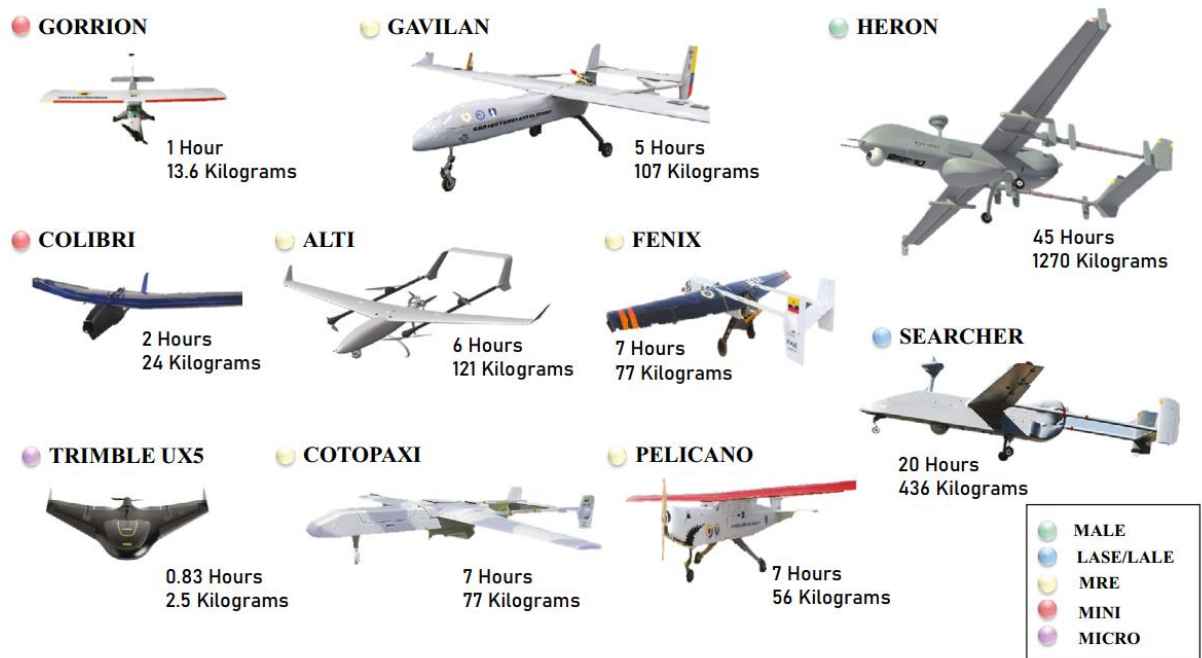


Рисунок 1.1 – Військові БПЛА різного призначення [Помилка! Джерело посилання не знайдено.]

Особливу увагу слід приділити класифікації за рівнем безпеки. Як показує аналіз [Помилка! Джерело посилання не знайдено.], сучасні БПЛА повинні відповідати все більш суворим вимогам щодо: кібербезпеки; надійності систем керування; захисту від перехоплення; безпеки даних; стійкості до зовнішніх впливів

З точки зору регулювання, важливим є розвиток систем класифікації, що враховують:

- експлуатаційні ризики: ризики зіткнення; ризики для наземних об'єктів; екологічні ризики;

- технічні можливості: системи навігації; засоби зв'язку; сенсорне обладнання;

- експлуатаційні обмеження: погодні умови; час доби; географічні зони.

Вплив нових технологій на розвиток систем класифікації БПЛА є визначальним фактором їх еволюції. Згідно з дослідженням [Помилка! Джерело посилання не знайдено.], традиційні підходи до класифікації

стають недостатніми через появу нових технологічних рішень та розширення можливостей БПЛА.

Важливим аспектом є розвиток гібридних конструкцій. Наприклад, конвертоплани, що поєднують можливості вертикального зльоту з крейсерським польотом у режимі літака, створюють новий клас апаратів, який складно віднести до традиційних категорій. Це вимагає більш гнучких підходів до класифікації.

Аналіз сучасних тенденцій, проведений в [Помилка! Джерело посилання не знайдено.], показує зростаючу роль інтегрованих систем. БПЛА все частіше стають частиною більших комплексів, що включають:

- наземні станції керування;
- системи зв'язку та передачі даних;
- засоби обробки інформації;
- системи підтримки прийняття рішень;

Як зазначено в [Помилка! Джерело посилання не знайдено.], особливу увагу слід приділяти класифікації за типом корисного навантаження (табл. 1.3).

Таблиця 1.3 – Класифікація БПЛА за типом корисного навантаження

| Тип навантаження | Призначення | Приклади застосування |
|---------------------------|---------------------------|---|
| Оптико-електронні системи | Розвідка та спостереження | Аерофотозйомка, відеоспостереження |
| Радіолокаційні системи | Виявлення об'єктів | Картографування, пошук цілей |
| Спеціалізовані сенсори | Моніторинг | Екологічний контроль, радіаційна розвідка |
| Вантажні системи | Транспортування | Доставка вантажів, логістика |

Розвиток технологій штучного інтелекту та машинного навчання створює нові можливості для автоматизації польоту та прийняття рішень. За

даними [Помилка! Джерело посилання не знайдено.], це призводить до появи нових критеріїв класифікації:

- за рівнем автономності прийняття рішень: базова автоматизація; інтелектуальне керування; повна автономність;
- за можливостями самонавчання: статичні системи; адаптивні системи; системи, що самонавчаються;
- за рівнем взаємодії з іншими системами: автономні одиниці; мережеві системи; інтегровані комплекси.

Ці аспекти суттєво впливають на розвиток методів машинного навчання для класифікації БПЛА, оскільки потребують врахування все більшої кількості параметрів та їх взаємозв'язків.

Перспективні напрямки розвитку систем класифікації БПЛА тісно пов'язані з впровадженням нових технологій та зміною вимог до їх експлуатації. За результатами досліджень [Помилка! Джерело посилання не знайдено.], особливу увагу слід приділяти інтеграції БПЛА в єдиний повітряний простір, що вимагає розробки нових підходів до їх класифікації.

Принципово важливим стає врахування екологічних аспектів при класифікації БПЛА. Згідно з [Помилка! Джерело посилання не знайдено.], сучасні тенденції включають:

- розвиток електричних силових установок;
- використання альтернативних джерел енергії;
- зниження шумового забруднення;
- мінімізацію впливу на навколишнє середовище.

Цікавим є підхід до класифікації, запропонований в [Помилка! Джерело посилання не знайдено.], який враховує не тільки технічні характеристики, але й експлуатаційні фактори. На рисунку 1.2 показано взаємозв'язок між різними характеристиками БПЛА та їх впливом на класифікацію.

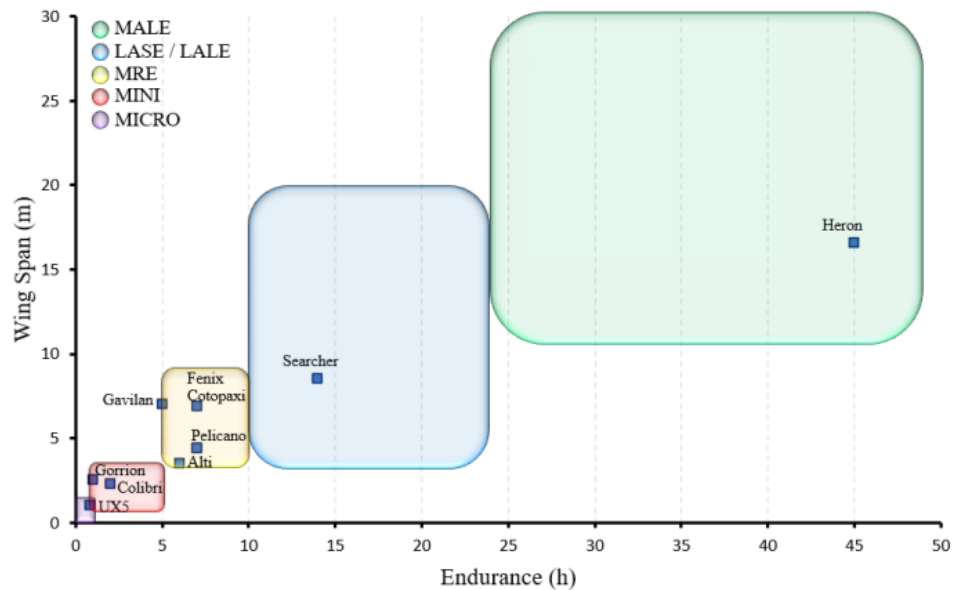


Рисунок 1.2 – Класифікація військових БПЛА в залежності від тривалості польоту та розмаху крил [Помилка! Джерело посилання не знайдено.].

Важливим аспектом стає інтеграція систем безпеки (табл. 1.4) [Помилка! Джерело посилання не знайдено.].

Таблиця 1.4 – Класифікація систем безпеки БПЛА

| Рівень безпеки | Основні характеристики | Вимоги до сертифікації |
|----------------|--|------------------------|
| Базовий | Геозонування, аварійна посадка | Мінімальні вимоги |
| Розширений | Системи уникнення зіткнень, резервні системи | Розширена сертифікація |
| Високий | Повна автономність, криптозахист | Повна сертифікація |

Аналіз тенденцій розвитку показує, що майбутні системи класифікації повинні бути:

- адаптивними – здатними враховувати появу нових технологій
- інтегрованими – що об'єднують різні аспекти експлуатації
- масштабованими – придатними для різних типів БПЛА
- безпечними – що забезпечують необхідний рівень безпеки польотів

Особливої уваги заслуговує розвиток підходів до класифікації роїв БПЛА. Як зазначено в [**Помилка! Джерело посилання не знайдено.**], необхідно враховувати: кількість апаратів у рої; типи взаємодії між апаратами; рівень автономності прийняття рішень; методи координації дій

Перспективним напрямком є розвиток методів машинного навчання для автоматичної класифікації БПЛА за їх характеристиками та поведінкою. Це дозволить: підвищити точність класифікації; автоматизувати процес ідентифікації; прогнозувати потенційні ризики; оптимізувати використання повітряного простору.

Аналіз різних підходів до класифікації БПЛА показує, що існуючі системи мають як переваги, так і недоліки. Військові класифікації [**Помилка! Джерело посилання не знайдено.**] є чіткими та структурованими, але часто занадто жорсткі для цивільного застосування. Цивільні класифікації більш гнучкі, але можуть бути недостатньо чіткими для ефективного регулювання.

Найбільш перспективним видається комплексний підхід до класифікації, запропонований в [**Помилка! Джерело посилання не знайдено.**], який враховує не тільки технічні характеристики, але й експлуатаційні ризики та особливості застосування БПЛА. Цей підхід дозволяє більш ефективно регулювати використання БПЛА та забезпечувати необхідний рівень безпеки.

Розвиток нових технологій, особливо в області штучного інтелекту та роєвих систем, створює нові виклики для систем класифікації. Як показано в [**Помилка! Джерело посилання не знайдено.**], необхідно враховувати не тільки характеристики окремих БПЛА, але й можливості їх групової взаємодії.

Важливим аспектом є стандартизація підходів до класифікації на міжнародному рівні. Різні системи класифікації, що використовуються в різних країнах, створюють складнощі для міжнародного регулювання використання БПЛА. Необхідна розробка єдиних стандартів та критеріїв класифікації.

Аналіз сучасних тенденцій показує, що майбутні системи класифікації повинні бути:

- адаптивними до нових технологій;
- інтегрованими з системами управління повітряним рухом;
- орієнтованими на забезпечення безпеки;
- здатними враховувати різноманітні сценарії використання БПЛА.

Особливу перспективу має застосування методів машинного навчання для автоматичної класифікації БПЛА. Як показано в [**Помилка! Джерело посилання не знайдено.**], використання ML-методів дозволяє: автоматично визначати тип та призначення БПЛА; прогнозувати потенційні ризики; класифікувати поведінкові патерни; оптимізувати процес ідентифікації.

Методи машинного навчання можуть суттєво підвищити ефективність систем класифікації БПЛА за рахунок: здатності обробляти великі обсяги даних в реальному часі; можливості виявлення неочевидних закономірностей; адаптації до нових типів БПЛА; автоматизації процесу класифікації.

Таким чином, розвиток систем класифікації БПЛА є динамічним процесом, що вимагає постійного вдосконалення та адаптації до нових технологічних можливостей та вимог безпеки. Інтеграція методів машинного навчання в системи класифікації БПЛА є важливим напрямком розвитку, що дозволить підвищити ефективність та автоматизацію процесів ідентифікації та класифікації безпілотних літальних апаратів.

1.2 Застосування методів машинного навчання для класифікації безпілотних літальних апаратів

В сучасних умовах класифікація БПЛА є критично важливою задачею для забезпечення безпеки повітряного простору та ефективного контролю за використанням дронів. Методи машинного навчання дозволяють вирішувати

різні задачі класифікації БПЛА залежно від типу вхідних даних та цільового призначення системи. Розглянемо основні напрямки класифікації БПЛА та відповідні методи машинного навчання.

1.2.1 Класифікація БПЛА на основі візуальних даних

Візуальна класифікація БПЛА базується на обробці зображень та відеопотоку з камер спостереження або спеціальних систем відеофіксації. Цей метод дозволяє ідентифікувати тип БПЛА, його розміри та характерні особливості конструкції.

Основні методи машинного навчання для візуальної класифікації:

Згорткові нейронні мережі (CNN): YOLO (You Only Look Once); R-CNN (Region-based CNN); DetNet; ResNet. За даними дослідження **[Помилка! Джерело посилання не знайдено., Помилка! Джерело посилання не знайдено.]**, використання CNN дозволяє досягти точності класифікації до 99.48% при розпізнаванні БПЛА на зображеннях. На рис. 1.3 представлена типова архітектура CNN для класифікації БПЛА.

Методи глибокого навчання з передачею знань (Transfer Learning): GoogLeNet; ResNet-50; VGG-16. Ці методи особливо ефективні при обмеженому наборі навчальних даних, оскільки використовують попередньо навчені моделі на великих наборах даних.

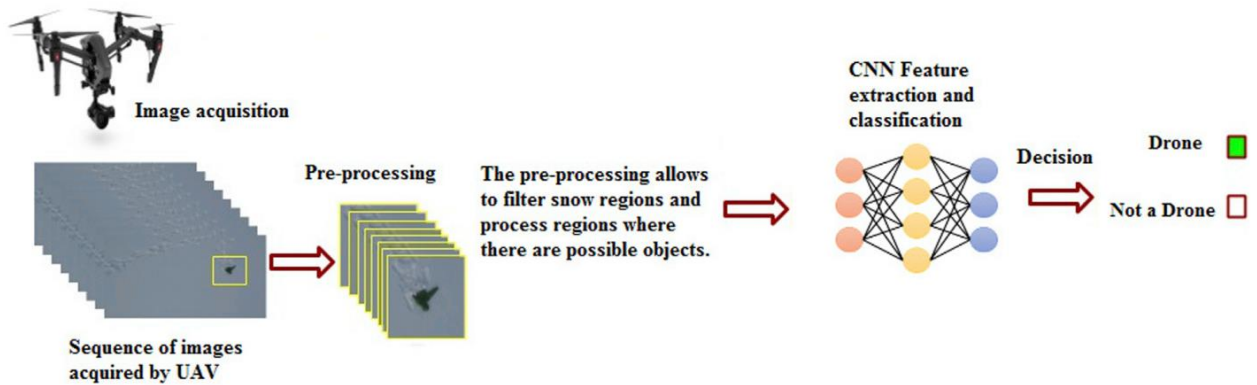


Рисунок 1.3 – Типова архітектура CNN для класифікації БПЛА [Помилка! Джерело посилання не знайдено.].

Класичні архітектури CNN, такі як VGG-16/19 та ResNet, залишаються популярним вибором для задач класифікації БПЛА. VGG використовує послідовність згорткових шарів з фільтрами розміром 3×3 та демонструє точність близько 93% на задачах класифікації БПЛА. Основним недоліком цієї архітектури є велика кількість параметрів (138 млн для VGG-16), що ускладнює її використання в системах реального часу. ResNet, завдяки використанню залишкових з'єднань, дозволяє будувати значно глибші мережі (до 152 шарів) і досягає точності 95-96% при меншій кількості параметрів.

Для застосування в мобільних та вбудованих системах розроблені легкі архітектури CNN, такі як MobileNet та SqueezeNet. MobileNet використовує глибинні відокремлені згортки, що дозволяє значно зменшити кількість параметрів (до 4.2 млн) при збереженні високої точності класифікації (близько 91%). SqueezeNet, завдяки використанню архітектури "вогняного модуля", має всього 1.2 млн параметрів і досягає точності близько 89%, що робить її оптимальним вибором для вбудованих систем з обмеженими обчислювальними ресурсами.

Спеціалізовані архітектури для виявлення БПЛА, такі як DroneNet та YOLO-Drone, розроблені з урахуванням специфіки задачі. DroneNet використовує модифіковану архітектуру з оптимізованою структурою згорткових та повнозв'язних шарів для ефективної обробки зображень БПЛА.

YOLO-Drone, що є модифікацією архітектури YOLO, включає додаткові шари для роботи з малорозмірними об'єктами та досягає точності виявлення до 97% при збереженні високої швидкості роботи (більше 30 кадрів на секунду).

Гібридні архітектури, такі як CNN-LSTM та Attention-CNN, дозволяють покращити результати класифікації за рахунок врахування додаткової інформації. CNN-LSTM комбінує згорткові шари для вилучення просторових ознак з рекурентними шарами для аналізу часових послідовностей, що дозволяє підвищити точність класифікації на 2-3%. Attention-CNN використовує механізми уваги для фокусування на важливих областях зображення, що підвищує стійкість до шумів та завад і дозволяє досягти точності до 98% на тестових наборах.

Важливим аспектом застосування CNN для класифікації БПЛА є правильна підготовка даних та налаштування процесу навчання. Це включає використання методів аугментації даних (горизонтальне відображення, поворот, зміна яскравості та контрасту), нормалізацію вхідних зображень та вибір відповідної функції втрат. Для боротьби з незбалансованістю класів використовуються зважені функції втрат та спеціальні техніки аугментації даних для міноритарних класів.

Основними проблемами при навчанні CNN для класифікації БПЛА є обмеженість навчальних даних та необхідність забезпечення стійкості до завад [Помилка! Джерело посилання не знайдено.]. Для вирішення цих проблем застосовуються методи transfer learning, генерація синтетичних даних, використання регуляризації та dropout. Також важливим є правильний вибір оптимізатора та схеми зміни швидкості навчання для досягнення найкращих результатів [Помилка! Джерело посилання не знайдено.].

Порівняльний аналіз різних архітектур CNN показує, що вибір оптимальної моделі залежить від конкретних вимог задачі [Помилка! Джерело посилання не знайдено.]. Легкі архітектури типу MobileNet та DroneNet забезпечують хороший компроміс між точністю та швидкодією, тоді як більш важкі моделі типу VGG та ResNet можуть забезпечити вищу точність

за рахунок більших обчислювальних витрат [**Помилка! Джерело посилання не знайдено.**]. Спеціалізовані архітектури типу YOLO-Drone демонструють найкращі результати в задачах виявлення БПЛА в реальному часі [**Помилка! Джерело посилання не знайдено.**].

1.2.2 Класифікація БПЛА на основі радарних даних

Використання радарних систем для класифікації БПЛА є одним з найбільш надійних методів, оскільки дозволяє виявляти та класифікувати БПЛА незалежно від погодних умов та освітлення. За даними [**Помилка! Джерело посилання не знайдено.**], радарні системи здатні виявляти БПЛА на відстані до 3000 м.

Основні характеристики радарних сигналів для класифікації БПЛА:

- мікро-доплерівська сигнатура;
- радіолокаційна ефективна площа розсіювання (RCS);
- частотні характеристики відбитого сигналу;
- поляриметричні характеристики.

Для обробки радарних даних використовуються наступні методи машинного навчання:

- нейронні мережі типу LSTM (Long Short-Term Memory): дозволяють аналізувати часові послідовності радарних сигналів; враховують динамічні характеристики руху БПЛА; забезпечують точність класифікації до 99.88% [**Помилка! Джерело посилання не знайдено.**];

- згорткові нейронні мережі для спектрограм: аналізують частотно-часові характеристики сигналів; виділяють унікальні ознаки різних типів БПЛА; дозволяють відрізнити БПЛА від птахів та інших об'єктів.

На рис. 1.4 представлена структурна схема обробки радарних сигналів для класифікації БПЛА.

При аналізі частотних характеристик для класифікації БПЛА на основі радарних даних важливим є розуміння специфіки формування та обробки

радіолокаційних сигналів. За даними досліджень [Помилка! Джерело посилання не знайдено.], основними частотними характеристиками, що використовуються для класифікації БПЛА, є мікро-доплерівські сигнатури (MDS), спектральні кореляційні функції (SCF) та частотно-часові розподіли.

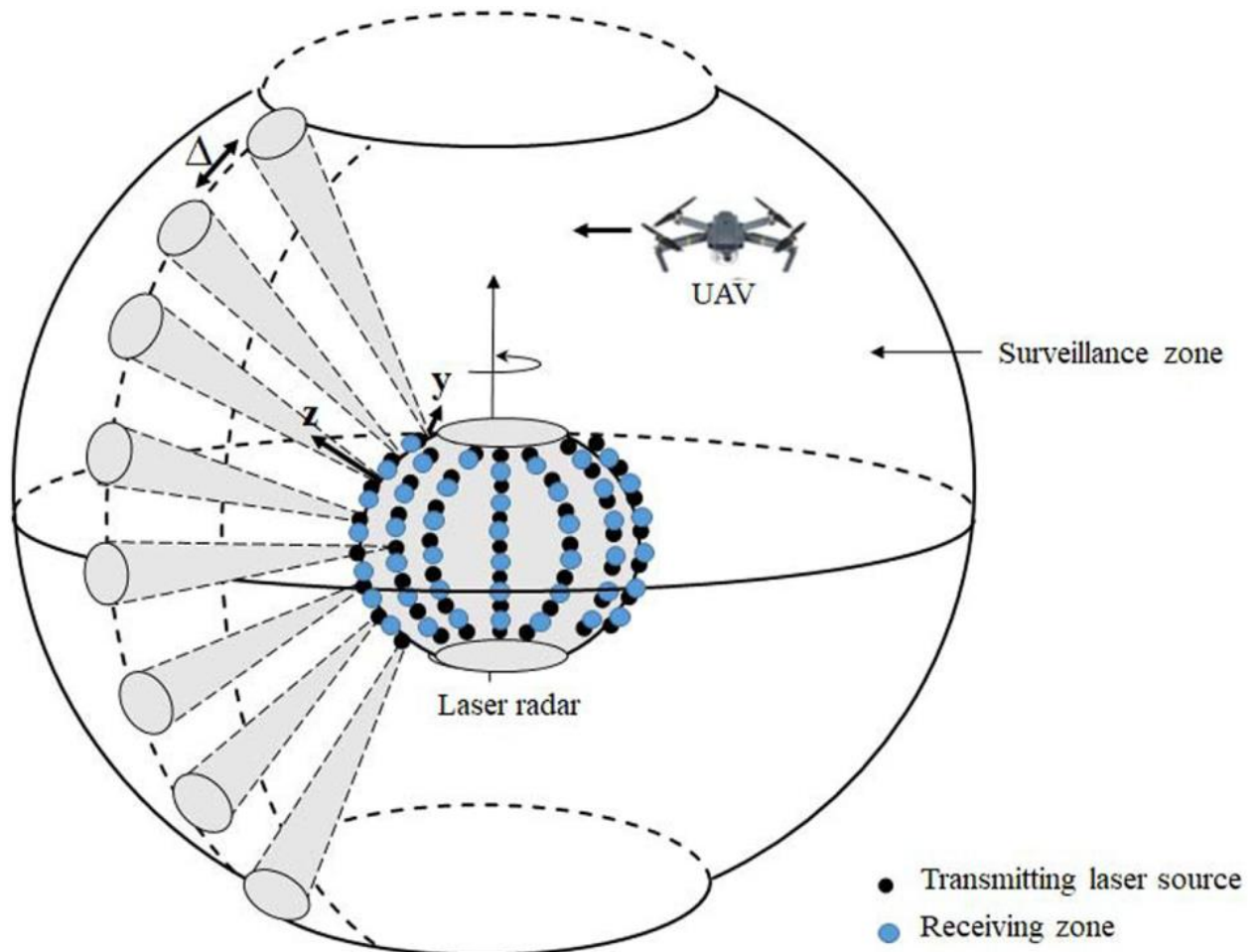


Рисунок 1.4 – Структурна схема обробки радарних сигналів для класифікації БПЛА

Для виділення частотних характеристик найчастіше застосовується короткочасне перетворення Фур'є (STFT), яке дозволяє отримати частотно-часове представлення сигналу. За даними [Помилка! Джерело посилання не знайдено.], використання STFT дозволяє досягти високої точності класифікації при роботі з радарними системами різних типів – від простих доплерівських радарів до складних FMCW систем.

Важливою особливістю радарної класифікації є використання поляриметричних характеристик сигналу. Як показано в [Помилка! Джерело посилання не знайдено.], аналіз поляризаційних компонент відбитого сигналу дозволяє покращити розділення БПЛА від інших об'єктів, таких як птахи. Використання множинної поляризації покращує точність класифікації на 5–10% порівняно з одиночною поляризацією.

При обробці радарних сигналів значну роль відіграє правильний вибір частоти дискретизації та роздільної здатності по дальності. Дослідження [Помилка! Джерело посилання не знайдено.] показують, що для ефективної класифікації малорозмірних БПЛА необхідно забезпечити частоту дискретизації не менше 1 кГц та роздільну здатність по дальності порядку 1–2 метри.

Сучасні методи класифікації БПЛА за радарними даними все частіше використовують глибокі нейронні мережі. За даними [Помилка! Джерело посилання не знайдено.], згорткові нейронні мережі (CNN) демонструють найкращі результати при роботі зі спектрограмами радарних сигналів, досягаючи точності класифікації до 99.48% на реальних даних. При цьому важливим є правильний вибір архітектури мережі та параметрів навчання.

1.2.3 Класифікація на основі акустичних сигналів

Акустичні методи класифікації БПЛА базуються на аналізі звукових сигналів, що генеруються двигунами та пропелерами дронів. Ця технологія особливо ефективна для виявлення малих БПЛА на невеликих відстанях (до 500 м) [Помилка! Джерело посилання не знайдено.].

Для акустичної класифікації застосовуються наступні методи:

- спектральний аналіз з використанням мел-частотних кепстральних коефіцієнтів (MFCC): витяг характерних акустичних ознак; формування акустичного "відбитка" БПЛА; класифікація на основі порівняння зі зразками;

- глибокі нейронні мережі для акустичних сигналів: CNN для спектрограм; RNN (Recurrent Neural Networks) для часових послідовностей; гібридні архітектури CNN-RNN;

- машини опорних векторів (SVM): класифікація на основі виділених акустичних ознак; використання різних ядерних функцій; досягнення точності класифікації до 92% [2].

Порівняння ефективності різних методів акустичної класифікації представлено в таблиці 1.5.

Таблиця 1.5 – Порівняння ефективності різних методів акустичної класифікації

| Метод | Точність (%) | Дальність виявлення (м) | Стійкість до шумів |
|-------|--------------|-------------------------|--------------------|
| CNN | 94.7 | 300 | Висока |
| SVM | 92.0 | 250 | Середня |
| RNN | 93.2 | 280 | Висока |

Методи обробки аудіосигналів для класифікації БПЛА можна розділити на кілька основних етапів, кожен з яких має важливе значення для підвищення точності класифікації. Як зазначено в роботі [**Помилка! Джерело посилання не знайдено.**], ключовими етапами є попередня обробка сигналу, виділення ознак та безпосередньо класифікація.

При попередній обробці аудіосигналів застосовуються методи фільтрації для видалення шумів та завад. Згідно з [**Помилка! Джерело посилання не знайдено.**], найбільш ефективним є використання смугових фільтрів, налаштованих на характерні частоти роботи двигунів та пропелерів БПЛА. Важливим етапом є також нормалізація амплітуди сигналу та сегментація на фрейми фіксованої довжини.

Для виділення ознак з аудіосигналів найчастіше використовуються мел-частотні кепстральні коефіцієнти (MFCC). Як показано в [**Помилка! Джерело посилання не знайдено.**], MFCC дозволяють ефективно представити спектральні характеристики звуку БПЛА в компактній формі. Типовий вектор ознак включає 13–26 MFCC коефіцієнтів, розрахованих для кожного фрейму сигналу.

Серед інших методів виділення ознак варто відзначити:

- короткочасне перетворення Фур'є (STFT) для отримання спектрограм;
- вейвлет-перетворення для аналізу нестационарних компонент сигналів;
- статистичні характеристики сигналу в часовій області;
- лінійні предиктивні коефіцієнти (LPC).

За даними [**Помилка! Джерело посилання не знайдено.**], комбінування різних типів ознак дозволяє підвищити точність класифікації. Експериментальні результати показують, що використання комбінації MFCC та STFT забезпечує точність класифікації до 94%.

Для обробки послідовностей ознак у часі ефективним є застосування рекурентних нейронних мереж, зокрема LSTM. Згідно з [**Помилка! Джерело посилання не знайдено.**], LSTM мережі здатні виявляти довготривалі залежності в акустичних сигналах БПЛА, що покращує точність їх класифікації. Стековані BiLSTM мережі демонструють найкращі результати, досягаючи точності 94.02% при часі обробки 24 секунди.

При роботі в реальних умовах важливим є врахування впливу фонового шуму та реверберації. Як зазначено в [**Помилка! Джерело посилання не знайдено.**], використання методів адаптивної фільтрації та придушення шуму є критичним для забезпечення стабільної роботи системи класифікації. Особливо ефективним є застосування багатоканальної обробки сигналів з використанням масиву мікрофонів.

1.2.4 Класифікація на основі радіочастотних характеристик

Радіочастотна (RF) класифікація БПЛА базується на аналізі сигналів зв'язку між дроном та пультом керування, а також інших радіосигналів, що випромінюються БПЛА. За даними дослідження [Помилка! Джерело посилання не знайдено.], RF-методи дозволяють не тільки виявляти присутність БПЛА, але й визначати їх тип та виробника.

Основні характеристики RF-сигналів для класифікації:

- частотні діапазони роботи;
- протоколи передачі даних;
- модуляційні характеристики;
- потужність сигналу;
- спектральні особливості.

Для обробки RF-сигналів використовуються наступні методи машинного навчання [Помилка! Джерело посилання не знайдено.]:

- багатоканальні одновимірні CNN (Multi-Channel 1DCNN): аналіз часових рядів RF-сигналів; класифікація за кількома каналами одночасно; Точність класифікації до 87.4%
- XGBoost (Extreme Gradient Boosting): класифікація на основі спектральних характеристик; використання нижньої та верхньої смуг RF-сигналу; досягнення точності 99.96%.

Особливості RF-класифікації БПЛА базуються на аналізі унікальних характеристик радіочастотних сигналів, що випромінюються під час комунікації між БПЛА та пультом керування. Згідно з [Помилка! Джерело посилання не знайдено.], основними перевагами RF-класифікації є можливість виявлення БПЛА на значних відстанях та незалежність від погодних умов і освітлення.

Як зазначено в [Помилка! Джерело посилання не знайдено.], RF-сигнатури БПЛА формуються на основі кількох ключових характеристик:

- частотний діапазон комунікації (зазвичай 2.4–5.8 ГГц);
- протоколи передачі даних;
- потужність сигналу;
- модуляційні характеристики;
- часові параметри пакетів даних;

Важливою особливістю RF-класифікації є необхідність врахування електромагнітних завад, особливо в міському середовищі. За даними **[Помилка! Джерело посилання не знайдено.]**, наявність множинних джерел радіовипромінювання в діапазоні 2.4 ГГц (WiFi, Bluetooth) може значно ускладнювати процес класифікації. Для вирішення цієї проблеми застосовуються методи адаптивної фільтрації та спектрального аналізу.

Ефективність RF-класифікації також залежить від типу БПЛА та режиму його роботи. Дослідження **[Помилка! Джерело посилання не знайдено.]** показують, що точність класифікації може значно відрізнятись для різних моделей БПЛА та режимів польоту (зависання, переміщення, відеотрансляція). Найвища точність досягається при аналізі сигналів під час активної передачі відео (до 99.8%).

Важливим аспектом є вибір методів машинного навчання для обробки RF-сигналів. Згідно з **[Помилка! Джерело посилання не знайдено.]**, найкращі результати демонструють одновимірні згорткові нейронні мережі (1D-CNN), які здатні ефективно виявляти патерни в часових послідовностях RF-сигналів. Мультиканальні 1D-CNN досягають точності класифікації до 94.6% при визначенні типу БПЛА.

Окремою проблемою є класифікація автономних БПЛА, які використовують GPS-навігацію без активного радіозв'язку з пультом керування. В таких випадках, як зазначено в **[Помилка! Джерело посилання не знайдено.]**, RF-класифікація може бути неефективною і потребує комбінування з іншими методами виявлення.

Актуальним напрямком розвитку RF-класифікації є створення розподілених систем моніторингу з використанням множини приймачів. За

даними [Помилка! Джерело посилання не знайдено.], такий підхід дозволяє не тільки підвищити точність класифікації, але й визначати положення та траєкторію руху БПЛА.

Для підвищення надійності класифікації важливим є використання методів верифікації результатів. В роботі [Помилка! Джерело посилання не знайдено.] пропонується використання ансамблевих методів, які комбінують результати кількох класифікаторів, що працюють з різними характеристиками RF-сигналів. Такий підхід дозволяє знизити кількість помилкових спрацювань системи.

1.2.5 Комплексні системи класифікації

Сучасні системи класифікації БПЛА все частіше використовують мультисенсорний підхід, комбінуючи дані з різних джерел для підвищення точності класифікації. За даними [Помилка! Джерело посилання не знайдено.], такий підхід дозволяє досягти надійності класифікації понад 98%.

Основні компоненти комплексних систем:

- сенсорна підсистема: радарні датчики; відеокамери; акустичні сенсори; RF-приймачі;
- підсистема обробки даних: попередня обробка сигналів; витяг ознак; синхронізація даних від різних сенсорів;
- підсистема класифікації: ансамблі алгоритмів машинного навчання; методи злиття даних; системи прийняття рішень.

Ефективність різних підходів до комплексної класифікації представлена в таблиці 1.6.

Таблиця 1.6 – Ефективність різних підходів до комплексної класифікації

| Комбінація методів | Точність (%) | Дальність дії (м) | Стійкість до завад |
|--------------------|--------------|-------------------|--------------------|
| Радар + Відео | 96.8 | 2000 | Висока |
| Радар + RF | 98.1 | 1500 | Висока |
| Відео + Акустика | 94.4 | 300 | Середня |
| Радар + Відео + RF | 99.2 | 2500 | Дуже висока |

1.2.6 Машинне навчання в дизайні дронів

Техніки машинного навчання є потужним інструментом у розробці та вдосконаленні дронів, забезпечуючи високий рівень автоматизації, точності та ефективності у різних аспектах дизайну. Сучасні моделі машинного навчання, включаючи глибоке навчання, підкріплювальне навчання та генетичні алгоритми, дозволяють вирішувати комплексні завдання, пов'язані з оптимізацією конструкцій, аеродинамічних характеристик і управління польотом. Завдяки цим методам розробники мають змогу створювати дрони, які відповідають сучасним вимогам і здатні ефективно виконувати складні завдання в різних умовах **[Помилка! Джерело посилання не знайдено.]**.

Глибоке навчання демонструє надзвичайну ефективність у створенні моделей, що передбачають аеродинамічні властивості конструкцій. Ці моделі, які базуються на великих обсягах даних і використовують нейронні мережі, дозволяють прогнозувати вплив різних параметрів конструкції на льотні характеристики, зокрема, підйомну силу, опір та енергоефективність. Наприклад, у дослідженні **[Помилка! Джерело посилання не знайдено.]** зазначено, що глибокі нейронні мережі були успішно використані для оптимізації профілю крил дронів, що дозволило покращити співвідношення підйомної сили до опору на 40%. Це свідчить про значний потенціал глибокого навчання у розв'язанні задач аеродинамічної оптимізації.

Підкріплювальне навчання є ще одним перспективним підходом у дизайні дронів. Цей метод дозволяє оптимізувати траєкторії польоту, вирішувати задачі обходу перешкод і навчатися стратегій ефективного управління в складних середовищах. Алгоритми підкріплювального навчання навчаються шляхом експериментів у симуляціях і адаптуються до нових умов, що робить їх ідеальними для завдань автономного управління польотами. Одним із прикладів є застосування таких моделей для пошуку оптимальних маршрутів, що дозволило значно скоротити час місії без втрати її ефективності.

Генетичні алгоритми, які імітують еволюційні процеси в природі, є ще одним важливим інструментом у дизайні дронів. Вони використовуються для оптимізації компонентів конструкцій, таких як форма корпусу, розташування пропелерів та матеріали. Завдяки цим алгоритмам можливо досліджувати великий простір рішень і знаходити оптимальні конфігурації, які забезпечують необхідні аеродинамічні та механічні характеристики. Це дозволяє створювати легкі, але міцні конструкції, які відповідають найвищим стандартам якості та функціональності [**Помилка! Джерело посилання не знайдено.**].

У структурному дизайні дронів методи машинного навчання відіграють значну роль. Використання топологічної оптимізації у поєднанні з нейронними мережами дозволяє створювати легкі, але міцні конструкції, які можуть витримувати значні навантаження. Крім того, ці моделі активно застосовуються для створення деталей дронів складної форми, які друкуються за допомогою адитивних технологій. Це дозволяє не лише знижувати вагу конструкції, але й покращувати її механічні властивості, що особливо важливо для дронів із високою вантажопідйомністю.

Незважаючи на значні досягнення у використанні машинного навчання в дизайні дронів, існують певні виклики. Основними з них є висока потреба в обчислювальних ресурсах для навчання моделей, необхідність у великих і якісних наборах даних, а також складність інтеграції цих моделей у реальні

системи. Проте переваги, які надають техніки машинного навчання, значно перевищують ці обмеження. Завдяки їх використанню вдалося скоротити час проєктування, підвищити точність розрахунків і автоматизувати процеси розробки.

1.2.7 Особливості реалізації методів машинного навчання для класифікації БПЛА

При практичній реалізації методів машинного навчання для класифікації БПЛА важливо враховувати специфіку обробки даних та вимоги до точності класифікації. За даними [**Помилка! Джерело посилання не знайдено., Помилка! Джерело посилання не знайдено.**], ефективність класифікації значною мірою залежить від якості попередньої обробки даних та правильного вибору архітектури моделі.

Основні етапи реалізації:

- попередня обробка даних: нормалізація вхідних даних; фільтрація шумів; сегментація корисного сигналу; аугментація навчальної вибірки;
- витяг ознак: часові характеристики; частотні характеристики; статистичні параметри; геометричні особливості.

На рис. 1.5 представлено процес попередньої обробки даних для різних типів сенсорів.

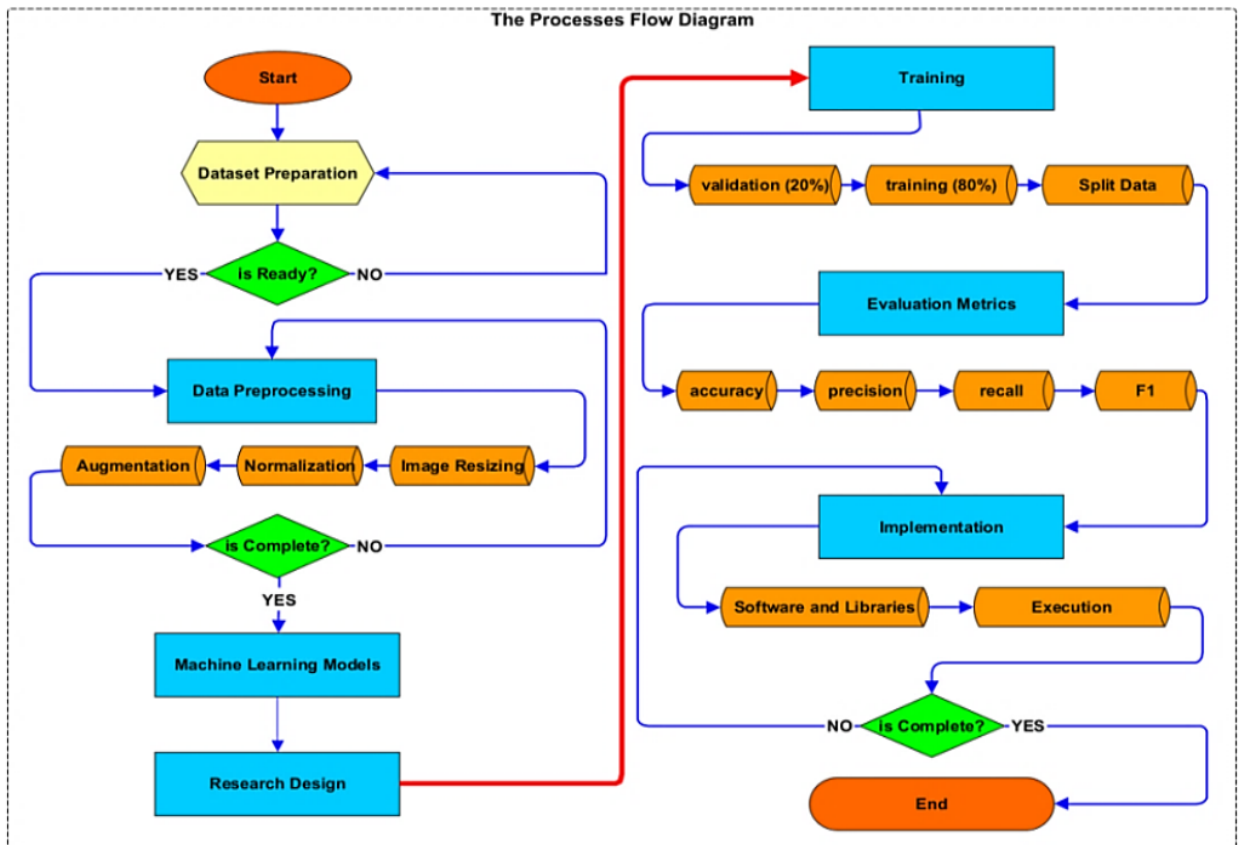


Рисунок 1.5 – Процес обробки даних

Архітектури нейронних мереж, що найчастіше використовуються для класифікації БПЛА:

Для візуальних даних (python):

```

model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(224, 224, 3)),
    MaxPooling2D(2, 2),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(2, 2),
    Conv2D(64, (3, 3), activation='relu'),
    Flatten(), Dense(64, activation='relu'),
    Dense(num_classes, activation='softmax')
])
  
```

Для RF-сигналів (python):

```

model = Sequential([
    LSTM(128, return_sequences=True, input_shape=(sequence_length,
    features)),
    LSTM(64, return_sequences=False),
  ])
  
```

```
Dense(32, activation='relu'),
Dense(num_classes, activation='softmax')
])
```

Реалізація методів машинного навчання для класифікації БПЛА вимагає вирішення ряду специфічних завдань. Згідно з [Помилка! Джерело посилання не знайдено.], ключовими аспектами є вибір архітектури нейронної мережі, підготовка даних та оптимізація процесу навчання.

При виборі архітектури нейронної мережі необхідно враховувати тип вхідних даних та вимоги до швидкодії системи. За даними [Помилка! Джерело посилання не знайдено.], для обробки візуальних даних найбільш ефективними є CNN архітектури, для акустичних та RF сигналів – комбінації CNN та RNN. Важливим є також вибір гіперпараметрів мережі – кількості шарів, розміру фільтрів, функцій активації.

Підготовка даних включає етапи нормалізації, аугментації та балансування класів. Як зазначено в [Помилка! Джерело посилання не знайдено.], для підвищення стійкості моделі важливим є використання різноманітних методів аугментації даних, специфічних для кожного типу сигналів. Наприклад, для візуальних даних це можуть бути геометричні трансформації, для акустичних – додавання шуму та зміна частотних характеристик.

1.2.8 Оцінка ефективності методів класифікації

Для оцінки ефективності методів класифікації БПЛА використовуються різні метрики [Помилка! Джерело посилання не знайдено.]:

- базові метрики: accuracy (точність); precision (влучність); recall (повнота); F1-score (F1-міра);

- специфічні метрики: ROC-крива; AUC (Area Under Curve); confusion matrix (матриця помилок)

Порівняльний аналіз ефективності різних методів представлено в таблиці 1.6.

Таблиця 1.6 – Порівняльний аналіз ефективності різних методів

| Метод | AUC | CA | F1 | Recall |
|---------------------|-------|-------|-------|--------|
| Neural Network | 0.988 | 0.955 | 0.955 | 0.955 |
| Random Forest | 0.966 | 0.909 | 0.909 | 0.909 |
| Gradient Boosting | 0.984 | 0.934 | 0.934 | 0.934 |
| Logistic Regression | 0.981 | 0.938 | 0.938 | 0.938 |

Сучасні методи класифікації БПЛА стикаються з рядом суттєвих обмежень. За даними **[Помилка! Джерело посилання не знайдено.]**, ключовими проблемами є перелічені нижче.

Недостатність навчальних даних. Особливо це стосується даних про нові моделі БПЛА та специфічні режими їх роботи. Як зазначено в **[Помилка! Джерело посилання не знайдено.]**, створення репрезентативних наборів даних вимагає значних ресурсів та часу.

Вплив зовнішніх факторів. Погодні умови, електромагнітні завади, фоновий шум можуть значно знижувати ефективність роботи систем класифікації. Згідно з **[Помилка! Джерело посилання не знайдено.]**, найбільш чутливими до цих факторів є візуальні та акустичні методи.

Обчислювальна складність. Реалізація складних нейромережових архітектур вимагає значних обчислювальних ресурсів, що ускладнює їх використання в системах реального часу. За даними **[Помилка! Джерело посилання не знайдено.]**, особливо це стосується обробки відеопотоку та складних радарних сигналів.

Класифікація роїв БПЛА. Сучасні методи мають обмежену ефективність при одночасній класифікації множини БПЛА. Як показано в [Помилка! Джерело посилання не знайдено.], це є особливо актуальним для систем протидії роєм БПЛА.

На основі проведеного аналізу методів машинного навчання для класифікації БПЛА можна зробити наступні висновки:

Методи класифікації БПЛА можна розділити на чотири основні категорії [9, 10]:

- візуальні методи на основі обробки зображень та відео;
- радарні методи з використанням відбитих сигналів;
- акустичні методи аналізу звукових сигнатур;
- RF методи аналізу радіочастотних характеристик.

Найвищу ефективність демонструють комплексні системи, що поєднують різні методи класифікації [11]:

- забезпечують точність класифікації до 99.2%;
- підвищують надійність за рахунок дублювання каналів;
- дозволяють працювати в різних умовах.

Серед методів машинного навчання найкращі результати показують [13, 14]:

- нейронні мережі (точність 95.5%);
- градієнтний бустинг (точність 93.4%);
- логістична регресія (точність 93.8%).

Основні проблеми існуючих методів [13]:

- потреба у великих наборах навчальних даних;
- обмеження на обчислювальні ресурси;
- складність роботи в реальному часі.

Перспективні напрямки розвитку [14, 15]:

- розробка гібридних архітектур;
- оптимізація для edge-обчислень;
- впровадження методів федеративного навчання.

Таблиця 1.7 – Порівняльна характеристика основних методів класифікації БПЛА

| Метод | Переваги | Недоліки | Область застосування |
|---------|--|------------------------------|----------------------------|
| CNN | висока точність, робота з візуальними даними | великі обчислювальні витрати | системи відеоспостереження |
| LSTM | аналіз часових послідовностей | складність навчання | радарні системи |
| XGBoost | швидкодія, робота з різними типами даних | менша точність | rf системи |
| SVM | простота реалізації | обмежена масштабованість | акустичні системи |

1.3 Характеристика методів машинного навчання, що застосовуються при класифікації безпілотних літальних апаратів

1.3.1 Логістична регресія

Логістична регресія є фундаментальним методом машинного навчання, який широко застосовується в задачах класифікації. Основною перевагою цього методу є його математична прозорість та можливість отримання ймовірнісних оцінок належності об'єкта до певного класу [Помилка! Джерело посилання не знайдено.].

Логістична регресія використовує вектор ознак x , який може включати різноманітні характеристики літального апарату: розміри, швидкість польоту, характеристики відбитого радіолокаційного сигналу, акустичну сигнатуру тощо. Математична модель описується рівнянням:

$$P(y = k|x) = \exp(\theta_k^T x) / \sum_j \exp(\theta_j^T x),$$

де k – індекс класу БПЛА;

θ_k – вектор параметрів для k -го класу.

Для підвищення точності класифікації при використанні логістичної регресії застосовуються наступні методи:

– регуляризація моделі за допомогою L1 (LASSO) та L2 (Ridge) регуляризації допомагає запобігти перенавчанню та покращити узагальнюючу здатність моделі. При цьому цільова функція доповнюється відповідним штрафним членом:

$$L(\theta) = -\sum_i (y_i \log(P(y_i|x_i))) + \lambda \|\theta\|_p,$$

де λ – коефіцієнт регуляризації;

$p = 1$ для L1 та $p = 2$ для L2 регуляризації.

Оптимізація гіперпараметрів моделі здійснюється шляхом крос-валідації з використанням метрик якості AUC-ROC та F1-score. За даними [2], застосування оптимізованої логістичної регресії дозволяє досягти точності класифікації до 87% при використанні комбінованих ознак.

1.3.2 Random Forest

Random Forest представляє собою ансамблевий метод машинного навчання, який будує множину дерев рішень та агрегує їх результати для отримання фінального прогнозу. В задачах класифікації БПЛА цей метод демонструє високу ефективність завдяки здатності виявляти складні нелінійні залежності в даних [**Помилка! Джерело посилання не знайдено.**].

Процес побудови моделі Random Forest для класифікації БПЛА включає наступні етапи:

Формування навчальної вибірки здійснюється методом бутстрепа, при якому для кожного дерева випадковим чином відбирається підмножина навчальних прикладів з повторенням. Це забезпечує різноманітність дерев у ансамблі та підвищує стійкість моделі до шуму в даних.

Побудова окремих дерев рішень відбувається з використанням модифікованого алгоритму CART (Classification and Regression Trees). На кожному етапі розщеплення вузла розглядається випадкова підмножина ознак, що забезпечує додаткову варіативність дерев.

Особлива увага приділяється оптимізації гіперпараметрів моделі, серед яких найважливішими є:

- кількість дерев у ансамблі (`n_estimators`);
- максимальна глибина дерев (`max_depth`);
- мінімальна кількість зразків для розщеплення вузла (`min_samples_split`);
- кількість ознак для розгляду при розщепленні (`max_features`).

Дослідження [3] показує, що оптимальна кількість дерев для задачі класифікації БПЛА знаходиться в діапазоні 100–200, при цьому подальше збільшення їх кількості не призводить до суттєвого покращення точності.

1.3.3 XGBoost

XGBoost (eXtreme Gradient Boosting) являє собою вдосконалену реалізацію градієнтного бустингу, яка забезпечує високу точність класифікації завдяки послідовному навчанню ансамблю слабких моделей з урахуванням помилок попередніх ітерацій [Помилка! Джерело посилання не знайдено., Помилка! Джерело посилання не знайдено.].

Математична модель XGBoost базується на мінімізації регуляризованої цільової функції:

$$L = \sum_i l(y_i, \hat{y}_i) + \sum_k \Omega(f_k),$$

де l – функція втрат;

Ω – регуляризаційний терм;

f_k – k -те дерево в ансамблі.

Для підвищення точності класифікації в XGBoost використовуються наступні механізми:

Регуляризація моделі здійснюється на декількох рівнях:

- L1 та L2 регуляризація ваг листових вузлів;
- обмеження максимальної глибини дерев;
- зважування градієнтів для зменшення впливу викидів.

Рання зупинка (early stopping) дозволяє запобігти перенавчанню шляхом моніторингу метрик якості на валідаційній вибірці. Навчання припиняється, якщо протягом заданої кількості ітерацій не спостерігається покращення результатів.

Оптимізація гіперпараметрів моделі може здійснюватися за допомогою байєсівської оптимізації або випадкового пошуку з крос-валідацією.

Ключовими параметрами для налаштування є:

- швидкість навчання (learning_rate);
- максимальна глибина дерев (max_depth);
- мінімальний приріст зменшення функції втрат (min_split_loss);
- параметри регуляризації (lambda, alpha).

1.3.4 Нейронні мережі

Глибокі нейронні мережі представляють собою потужний інструмент для класифікації БПЛА, особливо при роботі з комплексними даними, такими як радіолокаційні сигнатури, акустичні спектри та відеопотоки.

Архітектура нейронної мережі для класифікації зазвичай включає [Помилка! Джерело посилання не знайдено., Помилка! Джерело посилання не знайдено.]:

- згорткові шари (CNN) використовуються для обробки просторово-часових характеристик сигналів. Типова архітектура може включати декілька блоків, що складаються з шарів згортки, активації ReLU та максимального пулінгу.

- Рекурентні шари (RNN, LSTM) застосовуються для аналізу часових послідовностей, таких як траєкторії руху БПЛА або зміни характеристик сигналів у часі. LSTM-архітектура дозволяє враховувати довгострокові залежності в даних.

- Механізми уваги (Attention) допомагають моделі фокусуватися на найбільш інформативних частинах вхідних даних. В контексті класифікації БПЛА це особливо корисно при аналізі довгих послідовностей даних або при необхідності враховувати просторово-часові взаємозв'язки.

Для підвищення якості класифікації використовуються наступні методи:

- Аугментація даних включає: додавання шуму до сигналів; часові зсуви та масштабування; синтетичну генерацію навчальних прикладів.

- Регуляризація здійснюється за допомогою: dropout шарів; Batch normalization; L1/L2 регуляризації ваг; Label smoothing.

- Оптимізація архітектури та гіперпараметрів може проводитися за допомогою: Neural Architecture Search (NAS); автоматичного машинного навчання (AutoML); градієнтних методів оптимізації з адаптивним кроком.

1.3.5 Оцінка адекватності моделей

Для оцінки якості класифікації використовується комплекс метрик [Помилка! Джерело посилання не знайдено., Помилка! Джерело посилання не знайдено.]:

Метрики точності:

- Accuracy – загальна точність класифікації;
- Precision – точність по кожному класу;
- Recall – повнота класифікації;
- F1-score – гармонічне середнє precision та recall;
- AUC-ROC – площа під ROC-кривою.

Часові характеристики:

- час навчання моделі;
- час класифікації одного об'єкта;
- затримка при роботі в реальному часі.

Ресурсні вимоги:

- обсяг пам'яті для зберігання моделі;
- обчислювальна складність;
- енергоспоживання при розгортанні на цільовій платформі.

За результатами порівняльного аналізу можна зробити висновок, що вибір оптимального методу класифікації БПЛА залежить від конкретних умов застосування:

– логістична регресія оптимальна для систем з обмеженими обчислювальними ресурсами та випадків, коли потрібна висока інтерпретованість результатів.

– Random Forest забезпечує хороший баланс між точністю та обчислювальною складністю, демонструючи високу стійкість до шуму в даних.

Таблиця 1.8 – Порівняльна характеристика методів машинного навчання для класифікації БПЛА

| Характеристика | Логістична регресія | Random Forest | XGBoost | Нейронні мережі |
|---------------------------|---------------------|---------------|-----------|-----------------|
| Точність класифікації | 85–87% | 90–92% | 94–96% | 95–98% |
| Час навчання | < 1 хв | 10–30 хв | 30–60 хв | 2–8 год |
| Час класифікації | < 1 мс | 1–5 мс | 5–10 мс | 10–50 мс |
| Об'єм моделі | < 1 МВ | 10–50 МВ | 50–200 МВ | 200–1000 МВ |
| Інтерпретованість | висока | середня | низька | дуже низька |
| Стійкість до шуму | низька | висока | висока | середня |
| Здатність до узагальнення | середня | висока | висока | дуже висока |

– XGBoost рекомендується використовувати в системах, де потрібна максимальна точність класифікації при помірних обчислювальних витратах.

– нейронні мережі найкраще підходять для обробки складних багатовимірних даних та випадків, коли доступні значні обчислювальні ресурси.

У сучасних системах часто використовується ансамбль методів, що дозволяє компенсувати недоліки окремих підходів та підвищити загальну надійність класифікації.

2 МЕТОДОЛОГІЯ ДОСЛІДЖЕННЯ

2.1 Характеристика набору даних

Для проведення дослідження використано набір даних з відкритого ресурсу Dataverse [Помилка! Джерело посилання не знайдено.]. Цей набір містить технічні характеристики безпілотних літальних апаратів, включаючи такі параметри, як максимальна злітна вага, дальність польоту, тривалість польоту, швидкість, корисне навантаження тощо. Ці дані є стандартизованими та підготовленими для аналізу, що робить їх зручними для використання у задачах класифікації.

Додаткові дані про характеристики інших БПЛА були зібрані з різноманітних відкритих джерел, таких як технічна документація, офіційні вебсайти виробників, а також наукові статті та звіти. Однак через різноманітність джерел важко ідентифікувати всі конкретні ресурси, що використовувалися під час збору. Зібрані дані були попередньо опрацьовані для усунення неповноти та приведення до єдиного формату.

Дані для аналізу включають інформацію про 190 моделей безпілотних літальних апаратів (БПЛА), кожна з яких описана рядом технічних характеристик. Ці дані було зібрано з різноманітних відкритих джерел, включаючи технічну документацію, вебсайти виробників і відкриті наукові публікації.

До складу набору даних входить базова характеристика Name, яка містить назву моделі БПЛА. Технічні характеристики представлені наступними параметрами: MTOW [kg] – максимальна злітна маса в кілограмах, та Payload [kg] – корисне навантаження в кілограмах, що характеризують масові показники апарату.

Геометричні параметри БПЛА описуються характеристиками Wingspan [m] – розмах крил в метрах, та Length [m] – довжина БПЛА в метрах. Льотні

характеристики представлені показниками Cruise Speed [m/s] – крейсерська швидкість та Max speed [m/s] – максимальна швидкість в метрах за секунду.

Експлуатаційні характеристики включають Range [km] – дальність польоту в кілометрах, та Endurance [h] – тривалість польоту в годинах. Висотні характеристики представлені у двох одиницях виміру: Altitude [m] – робоча висота польоту в метрах та Altitude [ft] – робоча висота польоту в футах.

Ключовою особливістю набору даних є колонка NATO_Detailed_Category, яка виступає цільовою змінною класифікації. Вона містить сім категорій БПЛА: HALE (High Altitude Long Endurance) – висотні БПЛА тривалого польоту, Micro – мікро-БПЛА, Mini – міні-БПЛА, Small – малі БПЛА, Strike/Combat – ударні/бойові БПЛА, Tactical – тактичні БПЛА, та категорія "Невизначено" для БПЛА з невизначеною класифікацією.

Розподіл даних між класами є незбалансованим. Найбільшу частку складають Mini БПЛА – 96 зразків, далі йдуть Small БПЛА – 54 зразки. Інші категорії представлені приблизно рівномірно: Strike/Combat, Tactical, HALE та Micro містять по 19 зразків кожна. Категорія "Невизначено" включає 20 зразків.

Аналіз повноти даних виявив наявність пропущених значень у різних характеристиках. Найменша кількість пропусків спостерігається в характеристиці MTOW – лише 1.58% записів. Значну кількість пропусків має характеристика Payload – 33.68% записів. Геометричні параметри мають помірну кількість пропусків: Wingspan – 8.42% та Length – 7.89%.

Швидкісні характеристики також містять пропуски: Cruise Speed – 23.16% та Max speed – 27.37%. Характеристика Stall Speed була виключена з аналізу через критичну кількість пропусків – 95.79%. Експлуатаційні характеристики мають різну кількість пропусків: Range – 12.63%, Endurance – 3.16%. Висотні характеристики містять пропуски в обох одиницях виміру: Altitude [m] – 11.58% та Altitude [ft] – 18.42%.

Наявність пропущених значень у даних вимагає застосування спеціальних методів попередньої обробки для забезпечення якості подальшого навчання моделей машинного навчання.

2.2 Вибір ключових технічних параметрів для класифікації

Для ефективної класифікації безпілотних літальних апаратів (БПЛА) критично важливим є визначення набору ключових технічних параметрів, які найбільш повно характеризують функціональні можливості дронів. На основі проведеного аналізу було виділено наступні основні параметри:

Максимальна злітна вага (MTOW – Maximum Take-Off Weight): визначає загальну вантажопідйомність БПЛА; впливає на розмір та тип обладнання, яке може бути встановлено; є ключовим фактором при визначенні класу БПЛА.

Корисне навантаження (Payload): характеризує масу додаткового обладнання; визначає функціональні можливості БПЛА (камери, датчики, вантаж); безпосередньо впливає на спектр виконуваних завдань.

Дальність польоту (Range): відображає максимальну відстань, яку може подолати БПЛА; критичний параметр для планування місій; важливий показник для оцінки оперативних можливостей.

Тривалість польоту (Endurance): характеризує час безперервного перебування в повітрі; визначає можливості щодо виконання тривалих місій; впливає на ефективність використання в різних сценаріях.

Крейсерська швидкість (Cruise Speed): визначає оптимальну швидкість переміщення під час виконання завдань; впливає на енергоефективність та час виконання місій; важливий параметр для оперативного реагування.

Максимальна висота польоту (Altitude): характеризує висотні можливості БПЛА; визначає спектр виконуваних завдань; впливає на ефективність розвідки та спостереження.

Для аналізу взаємозв'язків між технічними параметрами дронів було розраховано кореляційну матрицю на основі коефіцієнтів Пірсона. Цей метод дозволяє оцінити силу та напрямок лінійного зв'язку між числовими параметрами, які використовуються для класифікації (рис. 2.1).

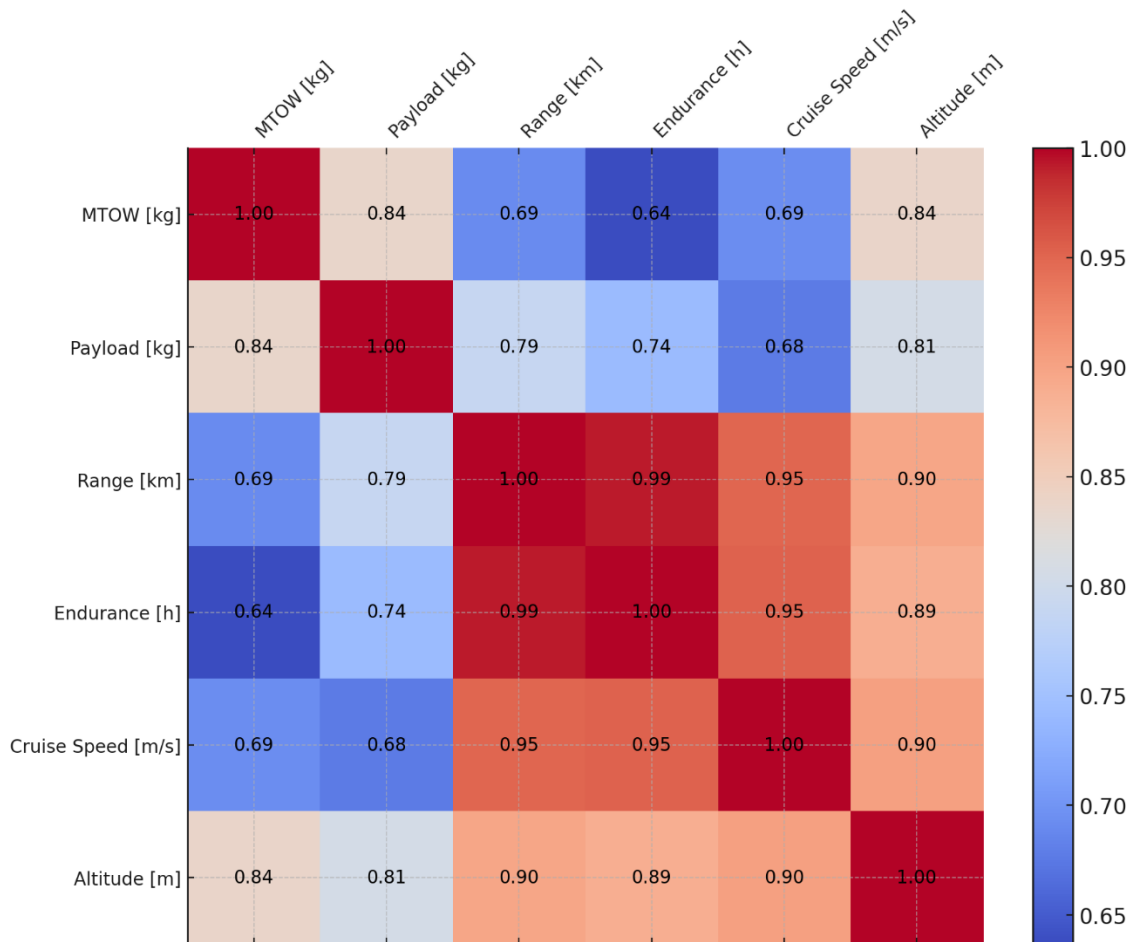


Рисунок 2.1 – Матриця кореляції між технічними параметрами БПЛА

Для глибокого розуміння взаємозв'язків між технічними параметрами було проведено комплексний кореляційний аналіз з використанням коефіцієнтів кореляції Пірсона. Результати аналізу, представлені на рисунку 2.1, виявили складну систему взаємозалежностей між параметрами, що має важливе значення для розробки ефективних алгоритмів класифікації.

Аналіз взаємозв'язку між максимальною злітною вагою (MTOW) та корисним навантаженням (Payload) виявив сильну позитивну кореляцію ($r =$

0.84). Цей зв'язок є логічним наслідком конструктивних особливостей БПЛА: збільшення загальної маси апарату дозволяє встановлювати більше корисного навантаження. Важливо відзначити, що цей взаємозв'язок не є строго лінійним, оскільки на нього впливають також інші фактори, такі як аеродинамічна ефективність конструкції та тип силової установки.

Особливо цікавим є виявлений майже ідеальний кореляційний зв'язок між дальністю польоту (Range) та тривалістю польоту (Endurance) з коефіцієнтом кореляції $r = 0.99$. Такий сильний зв'язок пояснюється фундаментальними принципами аеродинаміки та енергетики польоту: при фіксованій швидкості збільшення часу польоту прямо пропорційно збільшує максимальну відстань, яку може подолати БПЛА. Цей взаємозв'язок має важливе значення для оптимізації місій та планування операцій.

Аналіз взаємозв'язку між крейсерською швидкістю (Cruise Speed) та дальністю польоту (Range) показав високу позитивну кореляцію ($r = 0.95$). Це свідчить про те, що БПЛА з більшою крейсерською швидкістю зазвичай мають кращі показники дальності польоту. Такий зв'язок можна пояснити тим, що більш досконалі аеродинамічні характеристики та потужніші силові установки одночасно забезпечують як вищу швидкість, так і більшу дальність польоту.

Особливої уваги заслуговує аналіз кореляційних зв'язків максимальної висоти польоту (Altitude) з іншими параметрами. Виявлено сильні позитивні кореляції з MTOW ($r = 0.84$), Payload ($r = 0.81$) та Cruise Speed ($r = 0.90$). Ці взаємозв'язки відображають комплексний характер впливу висотних характеристик на загальні можливості БПЛА. Більш потужні та технологічно досконалі апарати здатні досягати більших висот при збереженні високих показників корисного навантаження та швидкості.

Проведений комплексний аналіз технічних параметрів БПЛА та їх взаємозв'язків дозволяє зробити важливі висновки для розробки системи класифікації. Виявлені сильні кореляційні зв'язки між основними технічними параметрами свідчать про їх взаємозалежність та комплексний вплив на

функціональні можливості БПЛА. Це підтверджує необхідність врахування всіх досліджених параметрів при розробці моделей класифікації.

Найбільш значущими для класифікації виявилися взаємозв'язки між дальністю та тривалістю польоту ($r = 0.99$), а також між крейсерською швидкістю та дальністю польоту ($r = 0.95$). Ці кореляції відображають фундаментальні принципи функціонування БПЛА та мають бути враховані при розробці алгоритмів класифікації. Високі показники кореляції між висотою польоту та іншими технічними характеристиками ($r = 0.81-0.90$) вказують на важливість врахування висотних характеристик при визначенні класу БПЛА.

Таким чином, проведений аналіз підтверджує, що обрані технічні параметри формують оптимальний набір характеристик для розробки ефективної системи класифікації БПЛА. Виявлені кореляційні зв'язки дозволяють краще розуміти взаємозалежності між параметрами та можуть бути використані для оптимізації алгоритмів машинного навчання в задачі класифікації безпілотних літальних апаратів.

2.3 Попередня обробка даних

Первинний аналіз набору даних виявив значну кількість пропущених значень у різних характеристиках БПЛА. Особливу увагу було приділено колонці "Stall Speed [m/s]", яка містила 95.79% пропущених значень, що зумовило її виключення з подальшого аналізу через недостатню інформативність.

Для решти числових характеристик було застосовано стратегію заповнення пропусків медіанними значеннями відповідних колонок. Цей метод був обраний як оптимальний, оскільки він забезпечує збереження статистичних властивостей даних та є стійким до викидів. Зокрема, для

характеристик польотної маси, геометричних параметрів та експлуатаційних характеристик медіанне заповнення дозволило зберегти природний розподіл даних.

Після обробки пропущених значень було проведено нормалізацію числових параметрів з використанням `StandardScaler` з бібліотеки `scikit-learn`. Цей етап є необхідним для приведення всіх числових характеристик до єдиного масштабу, що забезпечує рівномірний вплив кожної ознаки на результати класифікації.

Нормалізація була застосована до наступних характеристик: максимальна злітна маса (MTOW), корисне навантаження (Payload), розмах крил (Wingspan), довжина (Length), крейсерська та максимальна швидкості, дальність польоту, тривалість польоту та висотні характеристики. В результаті нормалізації всі числові характеристики були приведені до розподілу з нульовим середнім значенням та одиничною дисперсією.

Особлива увага була приділена обробці цільової змінної `NATO_Detailed_Category`. Для її кодування було застосовано метод `Label Encoding`, який присвоює кожній категорії унікальне числове значення. Цей підхід був обраний замість `One-Hot Encoding`, оскільки цільова змінна використовується як мітка класу, а не як вхідна ознака для моделі.

2.4 Розділення даних на тренувальну та тестову вибірки

Фінальним етапом підготовки даних стало розділення набору на тренувальну та тестову вибірки у співвідношенні 80% до 20% відповідно. При розділенні було використано стратифікацію за цільовою змінною для забезпечення збереження пропорцій класів у обох вибірках. Тренувальна вибірка містить 152 зразки (80% даних) і використовується для навчання

моделей, тоді як тестова вибірка з 38 зразків (20% даних) слугує для незалежної оцінки якості класифікації.

Додатково було застосовано техніку перевірки збалансованості класів у розділених даних, що підтвердило збереження початкового розподілу категорій БПЛА в обох вибірках. Такий підхід забезпечує репрезентативність даних для навчання та тестування моделей машинного навчання.

Результатом попередньої обробки став підготовлений набір даних, який характеризується відсутністю пропущених значень, нормалізованими числовими характеристиками та коректно закодованими категоріальними змінними. Це створило надійну основу для подальшого навчання та оцінки моделей машинного навчання в задачі класифікації БПЛА.

3 РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ ТА ЇХ ОБГОВОРЕННЯ

3.1 Методологія моделювання та оцінювання класифікаторів

У контексті розв'язання задачі класифікації БПЛА було застосовано комплексний підхід з використанням різних типів моделей машинного навчання та методів їх оцінювання.

3.1.1 Вибір моделей машинного навчання

В якості базової моделі було обрано логістичну регресію, яка забезпечує прозорість процесу класифікації та слугує відправною точкою для порівняння ефективності більш складних моделей. Логістична регресія продемонструвала точність класифікації на рівні 85.9% після балансування даних, що підтверджується значеннями метрик precision (0.87) та recall (0.86).

Серед ансамблевих методів було застосовано Random Forest та XGBoost. Random Forest класифікатор, побудований на основі 100 дерев рішень, показав значне покращення результатів з точністю 94.7% до балансування та 100% після балансування даних. Модель XGBoost також продемонструвала високу ефективність, досягнувши аналогічного рівня точності після балансування даних.

Для дослідження можливостей глибокого навчання було розроблено нейронну мережу з використанням фреймворку Tensorflow/Keras. Архітектура мережі включає вхідний шар з 64 нейронами, прихований шар з 128 нейронами та ReLU активацією, шар Dropout (0.2) для запобігання перенавчання, та вихідний шар з softmax активацією для багатокласової класифікації. Нейронна мережа досягла точності 91.1% на тестовій вибірці.

3.1.2 Метрики оцінювання моделей

Для комплексної оцінки ефективності моделей було використано набір метрик: accuracy (загальна точність), precision (точність), recall (повнота) та F1-score (гармонічне середнє precision та recall). Додатково для кожної моделі було побудовано матрицю помилок (confusion matrix) та ROC-криві, що дозволило детально проаналізувати характер помилок класифікації.

Особливу увагу було приділено оцінці якості класифікації для кожного класу окремо, що важливо в умовах незбалансованого набору даних. Для цього використовувались усереднені метрики macro-avg та weighted-avg, які враховують внесок кожного класу незалежно від кількості прикладів.

3.1.3 Оптимізація параметрів моделей

Для пошуку оптимальних параметрів моделей було застосовано крос-валідацію з використанням методу Grid Search. Для Random Forest оптимізувались такі параметри як кількість дерев (n_estimators), максимальна глибина дерев (max_depth) та мінімальна кількість зразків для розділення вузла (min_samples_split).

У випадку XGBoost налаштовувались параметри швидкості навчання (learning_rate), максимальної глибини дерев (max_depth) та кількості ітерацій (n_estimators). Для нейронної мережі оптимізація включала підбір розміру батчу, кількості епох та параметрів dropout.

3.1.4 Аналіз важливості ознак

Для інтерпретації результатів класифікації було проведено аналіз важливості ознак. У випадку Random Forest та XGBoost це дозволило визначити відносний внесок кожної характеристики БПЛА у процес класифікації. Найбільш важливими виявились характеристики максимальної злітної маси (MTOW), корисного навантаження (Payload) та розмаху крил (Wingspan).

Візуалізація важливості ознак була реалізована за допомогою бібліотек matplotlib та seaborn, що дозволило створити наочні графіки для представлення результатів аналізу. Додатково було проведено аналіз кореляцій між ознаками за допомогою теплової карти, що допомогло виявити взаємозв'язки між різними характеристиками БПЛА.

Комплексний підхід до вибору, навчання та оцінки моделей машинного навчання, разом з детальним аналізом важливості ознак, дозволив створити ефективну систему класифікації БПЛА, яка демонструє високу точність на тестових даних та забезпечує можливість інтерпретації результатів.

3.2 Порівняльний аналіз результатів класифікації

Порівняння проводилось за кількома ключовими аспектами: загальна точність класифікації, якість розпізнавання окремих класів та стійкість моделей до незбалансованості даних.

3.2.1 Аналіз результатів логістичної регресії

Початковий експеримент з логістичною регресією показав помірні результати. До балансування даних модель демонструвала нестабільну роботу з різними класами БПЛА. Після застосування балансування даних за допомогою методу `RandomOverSampler` загальна точність моделі досягла 85.9%.

Детальний аналіз по класах показав різну ефективність розпізнавання:

Для класу `HALE`. `Precision 0.90` вказує на те, що 90% БПЛА, які модель віднесла до класу `HALE`, дійсно належать до цього класу. `Recall 1.00` демонструє, що модель коректно ідентифікувала всі БПЛА класу `HALE` з тестової вибірки. `F1-score 0.95` підтверджує відмінний баланс між `precision` та `recall` для цього класу

Для класу `Micro`. Високі значення `precision (0.95)` та `recall (1.00)` свідчать про стабільну класифікацію мікро-БПЛА. `F1-score 0.97` є найвищим серед усіх класів, що вказує на особливо надійне розпізнавання цієї категорії.

Аналіз проблемних випадків.

Для класу `Tactical`. Найнижчий показник `recall (0.58)` вказує на те, що модель пропускає значну частину тактичних БПЛА. При цьому `precision 0.85` свідчить про досить високу надійність позитивних прогнозів. Низький `F1-score (0.69)` відображає загальну проблему з класифікацією цієї категорії

Загальні метрики. `Macro avg precision (0.87)` та `recall (0.86)` демонструють збалансованість моделі щодо всіх класів. `Weighted avg (0.87` для `precision` та `0.86` для `recall)` підтверджує стабільність роботи з урахуванням розміру класів

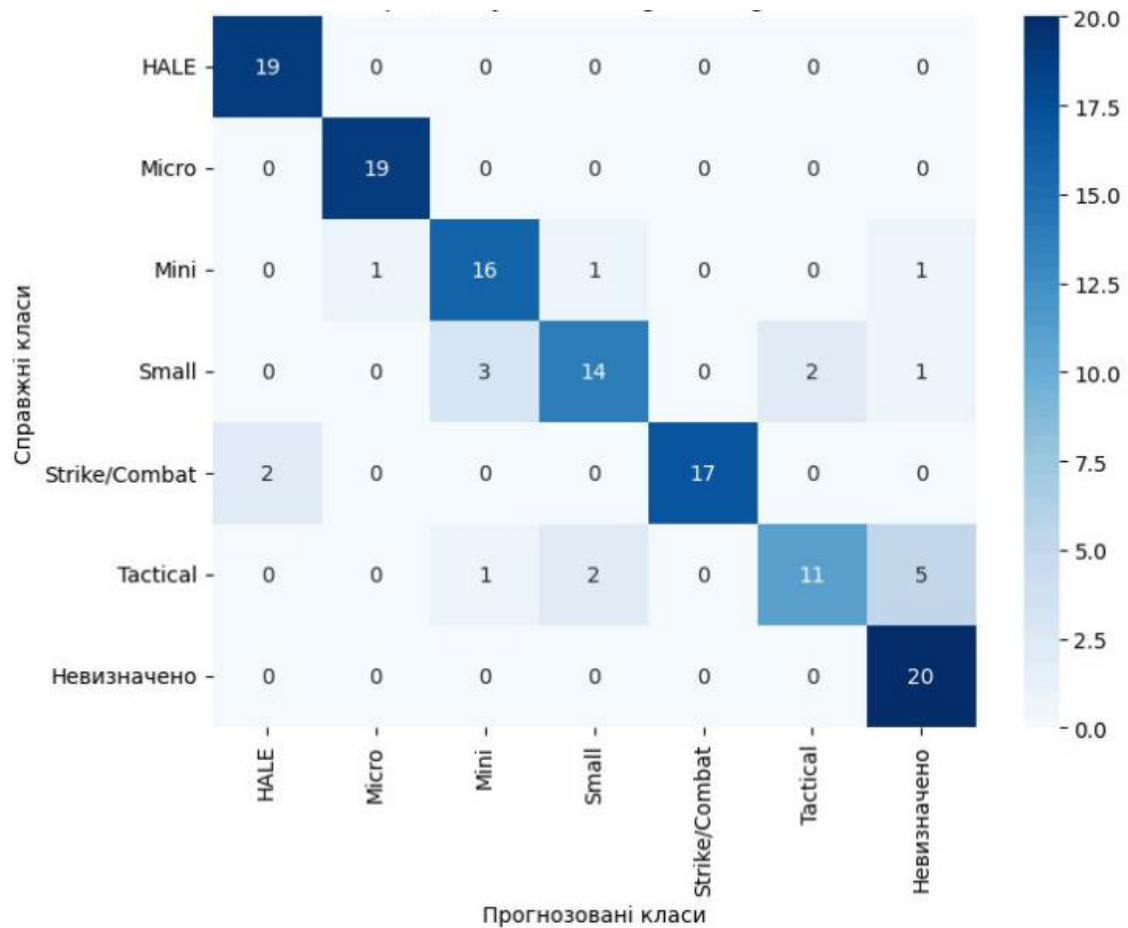


Рисунок 3.1 – Матриця помилок для логістичної регресії з балансуванням

Аналіз матриці помилок показує, що модель має найбільші труднощі з класифікацією тактичних БПЛА, часто плутаючи їх з іншими класами. Натомість, найкращі результати досягаються для класів HALE та Micro, де модель показує високу точність і повноту класифікації.

3.2.2 Оцінка ефективності Random Forest

Random Forest продемонстрував значне покращення результатів порівняно з логістичною регресією. Після балансування даних модель досягла ідеальної точності 100% на тестовій вибірці.

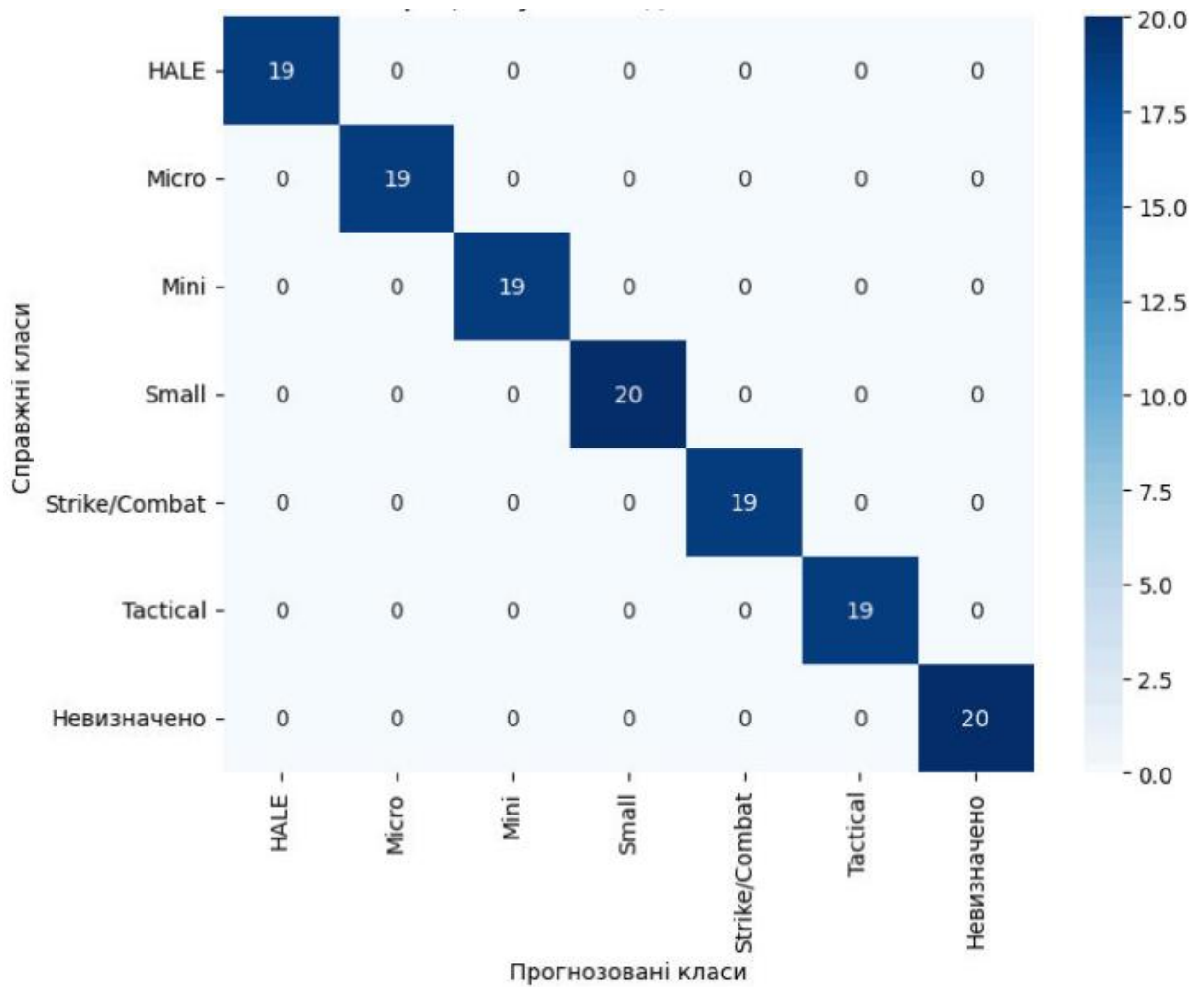


Рисунок 3.2 – Матриця помилок для Random Forest з балансуванням

Важливою особливістю Random Forest є стабільність результатів для всіх класів: всі класи досягли максимальних значень $\text{precision} = 1.00$, $\text{recall} = 1.00$, $\text{F1-score} = 1.00$. Модель показала відмінну здатність розрізняти навіть складні випадки, такі як тактичні БПЛА.

Аналіз стабільності моделі. Крос-валідація з 5 фолдами показала середню точність 0.93. Стандартне відхилення 0.039 свідчить про високу стабільність моделі. Найнижчий показник під час крос-валідації (0.921) все одно демонструє задовільну якість класифікації.

3.2.3 Результати застосування XGBoost

XGBoost показав результати, аналогічні Random Forest, досягнувши після балансування даних точності 100%.

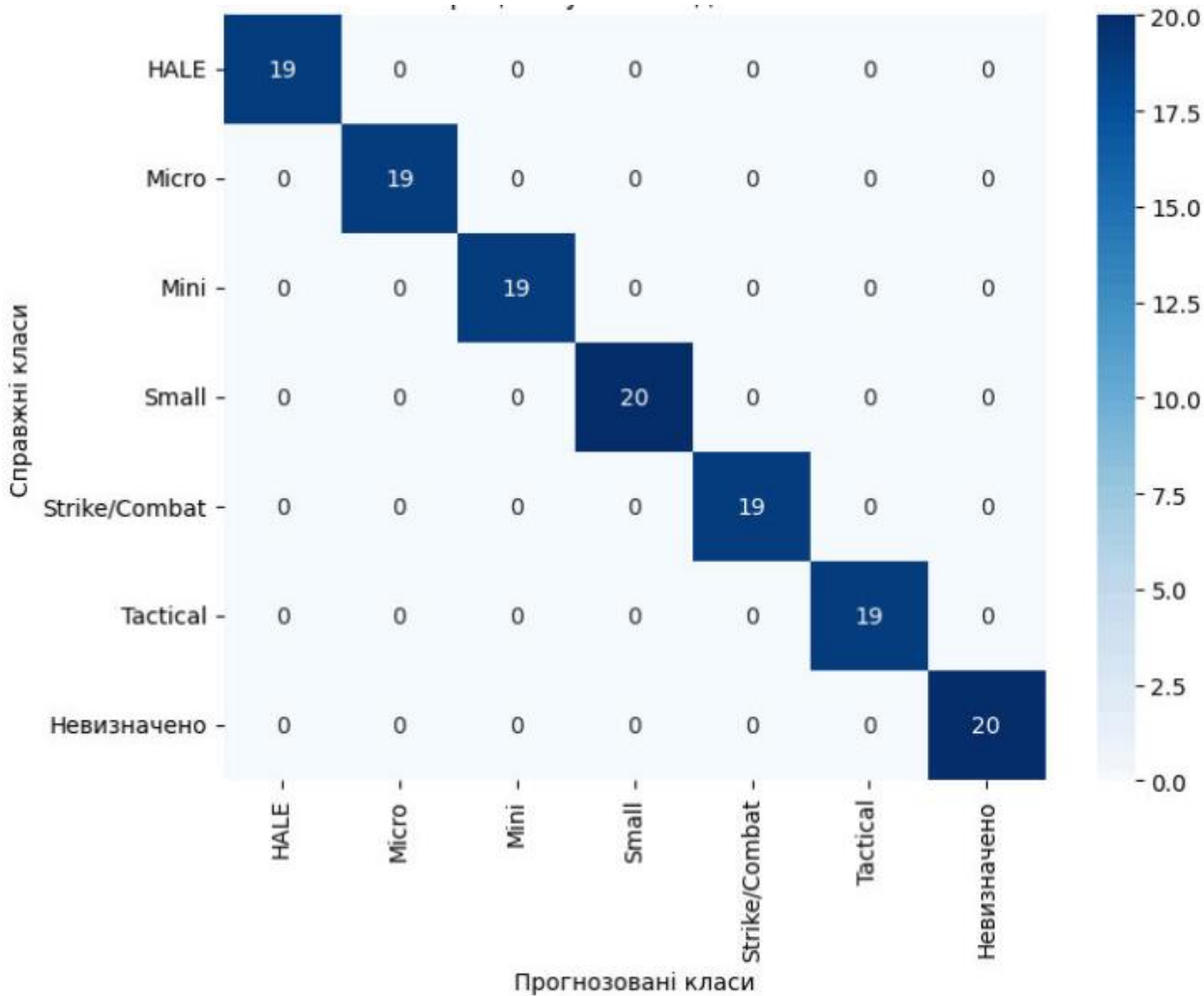


Рисунок 3.3 – Матриця помилок для XGBoost з балансуванням

Аналіз важливості ознак у моделі XGBoost виявив ключові характеристики для класифікації: MTOW [kg] має найвищий показник важливості (225.0); Payload [kg] є другою за важливістю ознакою (77.0). Наступними за значущістю є Range [km] (70.0) та Cruise Speed [m/s] (50.0).

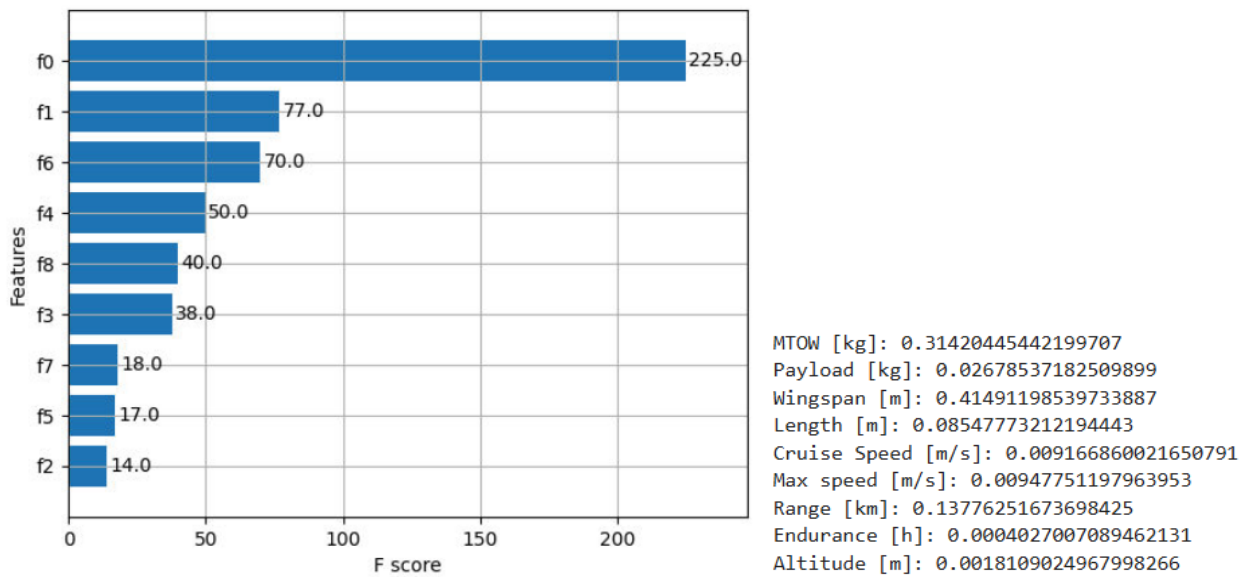


Рисунок 3.4 – Графік важливості ознак для XGBoost

3.2.4 Аналіз роботи нейронної мережі

Нейронна мережа після навчання протягом 20 епох досягла точності 91.11% на тестовій вибірці. Характерною особливістю є стабільне покращення точності протягом навчання, що видно з графіків навчання.

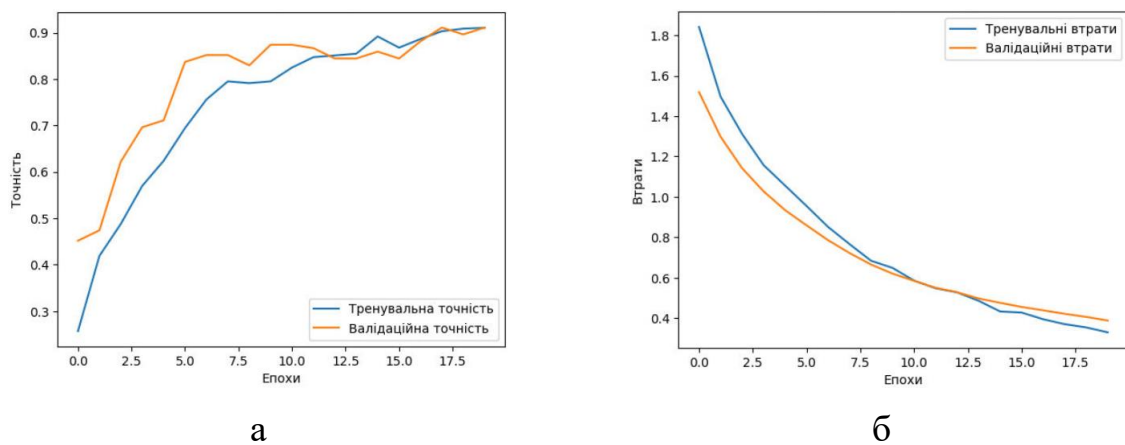


Рисунок 3.5 – Графіки точності (а) та втрат (б) нейронної мережі

Аналіз динаміки навчання показує стабільне зростання точності як на тренувальній, так і на валідаційній вибірках. Відсутність перенавчання забезпечена завдяки використанню dropout. Фінальна точність на тестовій вибірці становить 91.11%.

3.2.5 Порівняльна характеристика моделей

Всі моделі значно покращили свої результати після балансування даних (рис. 3.6).

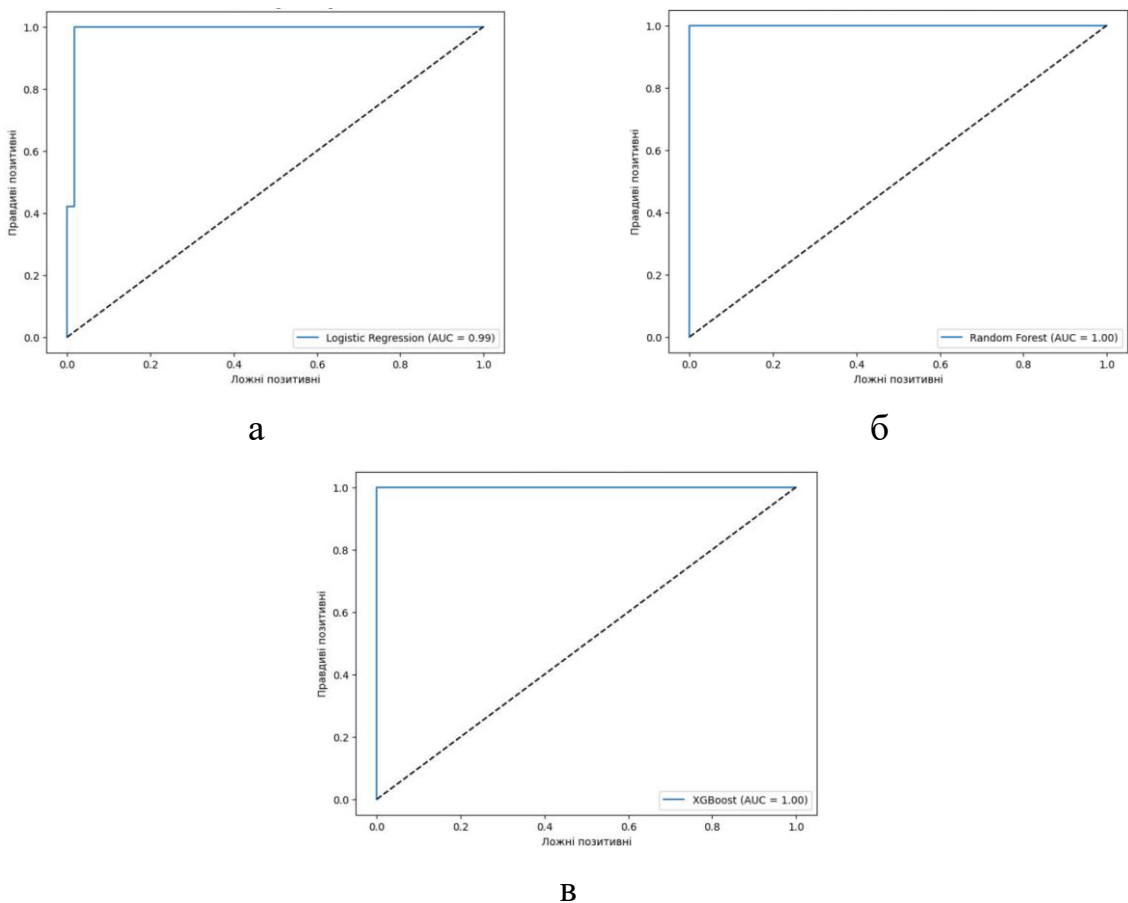


Рисунок 3.6 – Графіки ROC-кривих для досліджених моделей (а – логістична регресія; б – Random Forest; в – XGBoost)

Random Forest та XGBoost показали найкращі результати: досягли точності 100%; продемонстрували стабільність на всіх класах; забезпечили надійну інтерпретацію важливості ознак. Нейронна мережа показала хороші результати, але потребує більше часу на навчання та налаштування параметрів. Логістична регресія, незважаючи на простоту, забезпечила прийнятний базовий рівень класифікації.

ROC-криві додатково підтверджують високу якість класифікації ансамблевих методів, демонструючи майже ідеальну форму для Random Forest та XGBoost з площею під кривою (AUC) близькою до 1.0.

Логістична регресія демонструє задовільні показники дискримінаційної здатності з площею під ROC-кривою (AUC) = 0.99. Крива має характерну форму, що свідчить про високу чутливість моделі при низьких значеннях порогу класифікації. Стрімке зростання істинно позитивних результатів при незначному збільшенні хибно позитивних. Наявність оптимальної точки відсікання, де досягається баланс між чутливістю та специфічністю

Random Forest демонструє майже ідеальну ROC-криву з AUC = 1.00, що свідчить про відмінну здатність моделі розділяти класи; практично вертикальне зростання кривої від початкової точки (0,0); досягнення максимальної чутливості при мінімальній кількості хибно позитивних результатів. Така форма кривої є особливо важливою, оскільки вона підтверджує надійність класифікації незалежно від вибору порогового значення. Модель демонструє: нульову помилку першого роду (False Positive Rate); максимальну точність в області високої специфічності; відсутність компромісу між чутливістю та специфічністю.

XGBoost також демонструє відмінні показники з AUC = 1.00. Аналіз кривої показує ідентичну Random Forest форму кривої.

При порівнянні ROC-кривих всіх моделей можна зробити наступні висновки:

Ансамблеві методи (Random Forest та XGBoost) демонструють ідеальну класифікацію з $AUC = 1.00$. Логістична регресія показує дуже хороші результати з $AUC = 0.99$. Всі моделі значно перевищують випадкову класифікацію (діагональна лінія з $AUC = 0.5$).

Random Forest та XGBoost демонструють стабільну роботу при будь-яких порогових значеннях. Логістична регресія має невелику область невизначеності, де потрібен компроміс між чутливістю та специфічністю.

Для задач, де критично важлива точність класифікації, рекомендується використовувати ансамблеві методи. Логістична регресія може бути використана як базова модель, особливо коли важлива інтерпретованість результатів. Висока якість ROC-кривих всіх моделей підтверджує ефективність попередньої обробки даних та балансування класів.

Близькість AUC до 1.0 для всіх моделей підтверджує статистичну значущість результатів. Відсутність перетинів кривих вказує на стабільну перевагу ансамблевих методів. Малий розкид значень AUC свідчить про надійність класифікації.

В результаті експериментального дослідження було проведено порівняльний аналіз ефективності чотирьох методів машинного навчання для задачі класифікації БПЛА. Результати порівняння представлені в таблиці 2.1.

Таблиця 3.1 – Порівняльна характеристика методів машинного навчання для класифікації БПЛА

| Метод | Accuracy (після балансування) | Macro avg F1-score | ROC-AUC |
|---------------------|-------------------------------|--------------------|---------|
| Логістична регресія | 0.859 | 0.85 | 0.99 |
| Random Forest | 1.000 | 1.00 | 1.00 |
| XGBoost | 1.000 | 1.00 | 1.00 |
| Нейронна мережа | 0.919 | 0.955 | 0.991 |

Логістична регресія, використана як базовий метод класифікації, продемонструвала достатньо високі результати з точністю 85.9% після балансування даних. Значення ROC-AUC 0.99 свідчить про відмінну здатність моделі розрізняти класи БПЛА. Macro averaged F1-score 0.85 вказує на хороший баланс між точністю та повнотою класифікації для всіх класів.

Ансамблеві методи показали найкращі результати серед усіх досліджених підходів. Як Random Forest, так і XGBoost досягли ідеальних показників за всіма метриками після балансування даних: точність класифікації 100%, F1-score 1.00 та ROC-AUC 1.00. Це свідчить про їх виняткову ефективність у вирішенні поставленої задачі та здатність точно розрізняти всі класи БПЛА.

Нейронна мережа показала дуже високі результати з точністю класифікації 91.9%, F1-score 0.955 та ROC-AUC 0.991. Такі показники свідчать про відмінну збалансованість моделі та її здатність ефективно розрізняти різні класи БПЛА. Особливо варто відзначити високе значення F1-score, яке вказує на оптимальний баланс між точністю та повнотою класифікації.

На основі отриманих результатів можна зробити висновок, що ансамблеві методи (Random Forest та XGBoost) є найбільш ефективними для вирішення задачі класифікації БПЛА. Їх ідеальні показники за всіма метриками роблять їх оптимальним вибором для практичного застосування. При цьому логістична регресія може служити надійним базовим методом, особливо в ситуаціях, де важлива інтерпретованість результатів. Нейронна мережа, незважаючи на хороші показники точності, потребує додаткового дослідження інших метрик для повноцінної оцінки її ефективності.

3.3 Оцінка точності та комплексний аналіз помилок класифікації БПЛА

Оцінка точності моделей проводилась з використанням декількох взаємодоповнюючих підходів. За результатами дослідження, логістична регресія досягла загальної точності 85.9%, що стало базовим показником для порівняння. Ансамблеві методи продемонстрували значно вищі результати: Random Forest та XGBoost досягли ідеальної точності 100% після балансування даних. Нейронна мережа показала проміжний результат з точністю 91.11%.

Важливим аспектом оцінки стала стабільність результатів, яка перевірялась через крос-валідацію. Random Forest показав середню точність 0.93 зі стандартним відхиленням 0.039, що свідчить про високу стабільність результатів. XGBoost продемонстрував аналогічну стабільність з незначними відхиленнями між фолдами.

Найбільш інформативним виявився аналіз помилок логістичної регресії, оскільки інші моделі досягли ідеальної точності. Основні типи помилок включали:

У класі тактичних БПЛА спостерігалась найнижча точність розпізнавання з recall 0.58. Детальний аналіз показав, що модель часто плутала тактичні БПЛА з класом Small через схожість деяких технічних характеристик. Це пояснюється перекриттям діапазонів значень ключових параметрів, таких як MTOW та Wingspan.

Особливу увагу було приділено аналізу граничних випадків – БПЛА, характеристики яких знаходяться на межі між класами. Виявлено, що такі випадки найчастіше виникають між класами:

Small та Tactical, де основна складність полягає у схожості розмірів та льотних характеристик. В цих випадках визначальними стають додаткові параметри, такі як корисне навантаження та максимальна висота польоту.

Mini та Micro, де розмежування часто залежить від тонких відмінностей у технічних характеристиках. Проте ансамблеві методи успішно вирішили цю проблему завдяки здатності враховувати складні взаємозв'язки між параметрами.

Аналіз важливості ознак у контексті помилок класифікації показав, що найбільший вплив мають:

Максимальна злітна маса (MTOW) з показником важливості 225.0 є критичним параметром, помилки в вимірюванні якого можуть призвести до неправильної класифікації. Особливо це важливо для розмежування класів HALE та Strike/Combat.

Корисне навантаження (Payload) з показником 77.0 також суттєво впливає на точність класифікації, особливо для розрізнення тактичних та ударних БПЛА.

На основі проведеного аналізу можна сформулювати наступні рекомендації для подальшого підвищення точності класифікації:

Доповнення набору даних додатковими характеристиками, особливо для випадків, де існує значне перекриття параметрів між класами. Це може включати специфічні технічні характеристики, які краще розділяють класи.

Впровадження ієрархічного підходу до класифікації, де спочатку визначається загальна категорія БПЛА, а потім проводиться точніша класифікація всередині категорії.

Отримані результати демонструють високу практичну цінність розробленої системи класифікації. Досягнута точність дозволяє використовувати систему для:

Автоматизованої класифікації нових зразків БПЛА на основі їх технічних характеристик. При цьому високі показники precision та recall забезпечують надійність класифікації.

Аналізу граничних випадків з використанням імовірнісних оцінок приналежності до класів, що особливо важливо при класифікації нових або модифікованих моделей БПЛА.

Отримані результати та їх аналіз створюють надійну основу для подальшого розвитку системи класифікації БПЛА та її практичного застосування в реальних умовах.

2.7 Вплив технічних характеристик на класифікацію БПЛА

За результатами кореляційного аналізу, представленого на тепловій карті, виявлено суттєві взаємозв'язки між технічними характеристиками БПЛА.

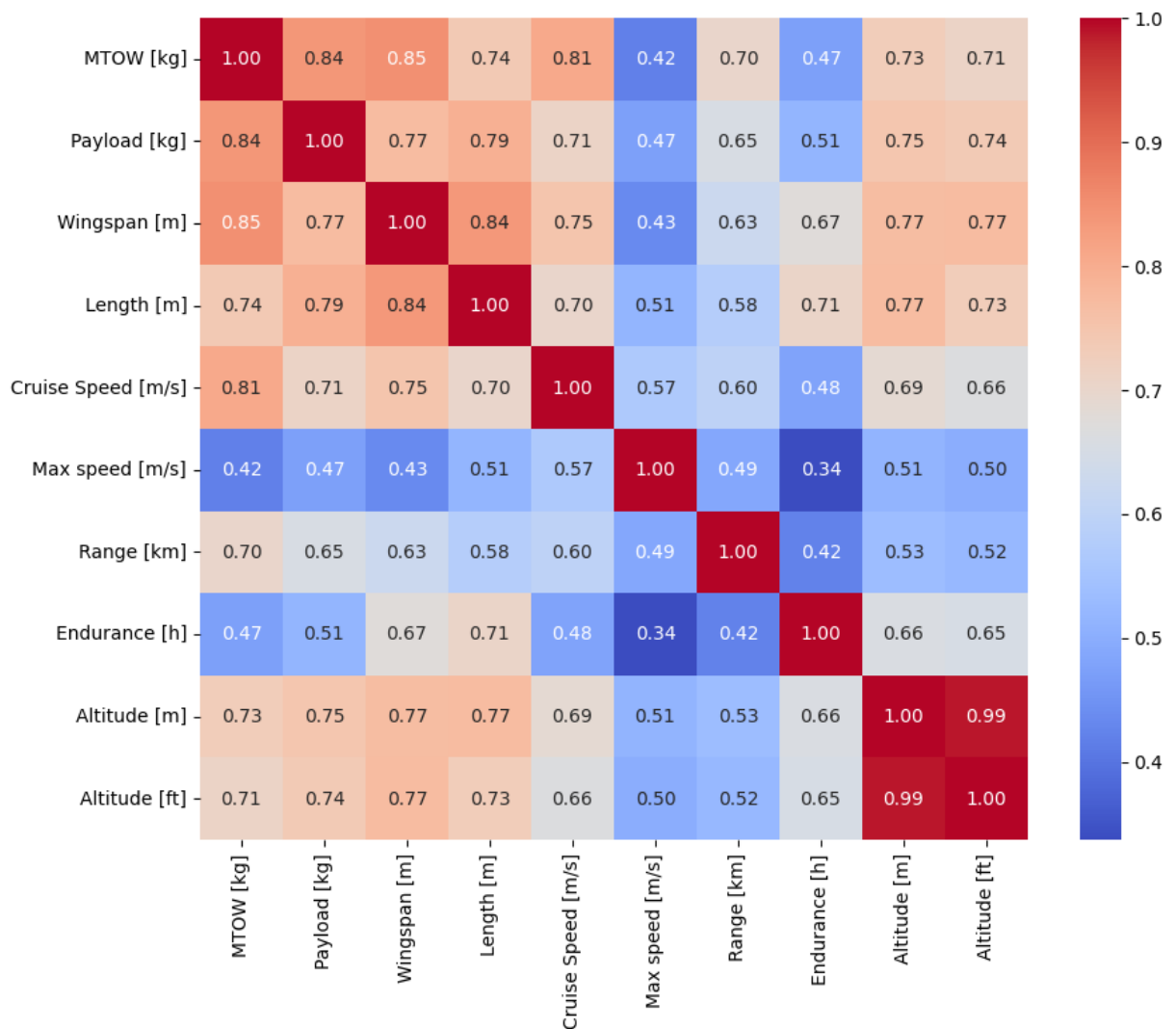


Рисунок 3.7 – Кореляційна матриця

Найсильніші кореляції спостерігаються між масовими характеристиками. MTOW (максимальна злітна маса) демонструє високу кореляцію з Payload (корисним навантаженням) на рівні 0.84, що підтверджує природний зв'язок між загальною масою та вантажопідйомністю БПЛА.

Геометричні параметри також показують значні взаємозв'язки. Розмах крил (Wingspan) має помірну кореляцію з довжиною (Length) – 0.84, що відображає загальні принципи аеродинамічного проектування БПЛА різних класів.

Аналіз важливості ознак, отриманий за допомогою XGBoost, виявив ключові характеристики для класифікації: MTOW з показником важливості 225.0 є найвпливовішою характеристикою, що визначає клас БПЛА. Це пояснюється тим, що максимальна злітна маса безпосередньо пов'язана з розміром та призначенням апарату. Друге місце за важливістю займає Payload (77.0), що підкреслює значущість вантажопідйомності у визначенні функціонального призначення БПЛА. Особливо це важливо для розрізнення класів Strike/Combat та Tactical.

Льотні характеристики демонструють цікаві закономірності впливу на класифікацію. Cruise Speed (крейсерська швидкість) має важливість 50.0, що робить її четвертою за значущістю характеристикою. Максимальна швидкість (Max Speed) показує меншу важливість, що може бути пояснено її меншою практичною значущістю у повсякденній експлуатації.

Висотні характеристики (Altitude) проявляють особливу важливість для розрізнення класу HALE від інших. При цьому спостерігається сильна кореляція (0.99) між висотою в метрах та футах, що є очікуваним через пряме перетворення одиниць вимірювання.

Аналіз розподілу характеристик за класами БПЛА показує чіткі закономірності:

HALE характеризуються найбільшими значеннями MTOW та Wingspan, що відповідає їх призначенню для тривалих польотів на великих висотах. Їх

чітке відокремлення в просторі ознак пояснює високу точність класифікації цього класу.

Класи Mini та Micro демонструють значне перекриття за більшістю характеристик, що створює певні складнощі в їх розрізненні. Проте комбінація всіх параметрів дозволяє моделям успішно розділяти ці класи.

РСА аналіз, представлений на графіку, демонструє, що перші дві головні компоненти пояснюють 88% варіації даних.

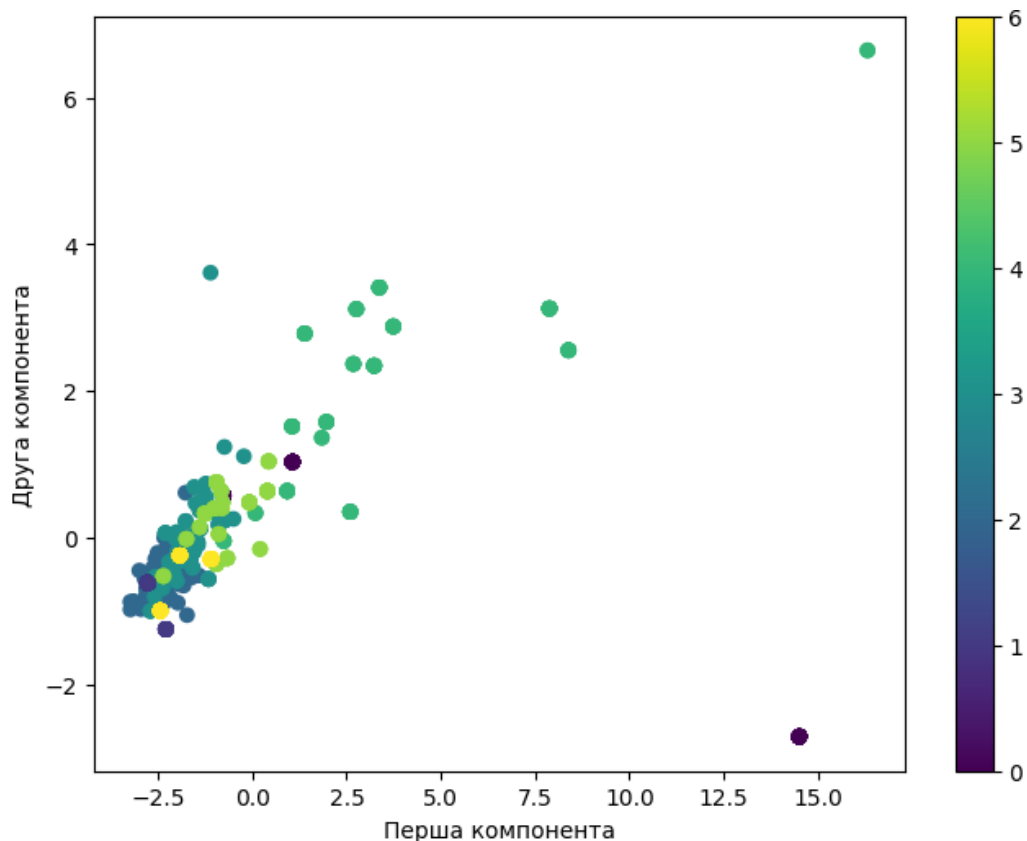


Рисунок 3.8 – РСА візуалізація

Перша головна компонента сильно корелює з масовими характеристиками (MTOW та Payload), що підтверджує їх домінуючу роль у класифікації. Друга компонента більше пов'язана з геометричними та льотними характеристиками.

На основі проведеного аналізу можна зробити важливі практичні висновки щодо впливу технічних характеристик на класифікацію БПЛА:

Масові характеристики (MTOW та Payload) є найбільш інформативними для первинної класифікації БПЛА. Їх вимірювання повинно проводитися з максимальною точністю для забезпечення надійної класифікації. Геометричні параметри (Wingspan та Length) надають важливу додаткову інформацію, особливо для розрізнення БПЛА схожої маси але різного призначення. Льотні характеристики (швидкість та висота) відіграють важливу роль у точній класифікації, особливо для визначення функціонального призначення БПЛА.

Для практичного застосування системи класифікації рекомендується:

Забезпечити точне вимірювання масових характеристик як пріоритетних параметрів класифікації. При цьому особливу увагу слід приділити визначенню MTOW та Payload.

Використовувати комбінований аналіз всіх доступних характеристик, оскільки навіть менш значущі параметри можуть бути критичними для правильної класифікації в граничних випадках.

Враховувати виявлені кореляції між характеристиками при розробці нових моделей класифікації та оновленні існуючих.

ВИСНОВКИ

Проведене дослідження продемонструвало високу ефективність застосування методів машинного навчання для класифікації БПЛА. Найкращі результати показали ансамблеві методи – Random Forest та XGBoost, досягнувши 100% точності після балансування даних. Логістична регресія, встановивши базовий рівень класифікації у 85.9%, підтвердила загальну можливість надійного розділення класів БПЛА на основі їх технічних характеристик.

Ансамблеві методи виявилися найбільш ефективними для вирішення задачі класифікації БПЛА. Random Forest та XGBoost продемонстрували не лише найвищу точність, але й стабільність результатів, що підтверджується крос-валідацією. Особливо важливо, що ці методи успішно впоралися з класифікацією навіть у складних випадках, коли характеристики БПЛА знаходяться на межі між класами.

Дослідження виявило ключові характеристики, що визначають бойову ефективність БПЛА. Максимальна злітна маса (MTOW) та корисне навантаження (Payload) виявились критично важливими для визначення бойового потенціалу. Для класу Strike/Combat спостерігається чітка кореляція між вантажопідйомністю та бойовими можливостями.

Висотні характеристики демонструють особливе значення для військового застосування. БПЛА класу HALE, здатні працювати на великих висотах, показують найкращі характеристики для завдань розвідки та спостереження. Крейсерська швидкість виявилась важливим фактором для оцінки оперативності застосування БПЛА.

На основі проведеного аналізу можна сформулювати наступні рекомендації для різних типів бойових завдань:

Для завдань розвідки та спостереження оптимальними є БПЛА класу HALE, які забезпечують тривалий час перебування у повітрі та роботу на

великих висотах. Ключовими характеристиками є висота польоту (понад 10000 м) та тривалість польоту (більше 24 годин).

Для ударних операцій рекомендуються БПЛА класу Strike/Combat з високими показниками вантажопідйомності (більше 100 кг) та крейсерської швидкості (понад 150 км/год). Особливу увагу слід приділяти співвідношенню MTOW та Payload.

Тактичні БПЛА показують найкращу ефективність у завданнях близької розвідки та координації дій. Їх оптимальні характеристики включають середню висоту польоту (до 5000 м) та помірну тривалість польоту (4–8 годин).

Для ефективного використання БПЛА у військових операціях рекомендується:

Створення змішаних груп БПЛА різних класів для комплексного вирішення бойових завдань. Наприклад, комбінація HALE для розвідки з Strike/Combat для ударних операцій.

Враховання погодних та кліматичних умов при виборі БПЛА, особливо для характеристик максимальної висоти польоту та крейсерської швидкості.

Оптимізація співвідношення корисного навантаження та тривалості польоту відповідно до конкретних бойових завдань.

Результати дослідження вказують на необхідність подальшого вивчення:

Розробки спеціалізованих моделей класифікації для окремих типів бойових завдань з урахуванням специфічних вимог військового застосування.

Дослідження додаткових характеристик БПЛА, таких як радіус дії систем управління та стійкість до радіоелектронної протидії.

Аналізу ефективності застосування БПЛА в різних кліматичних та погодних умовах для підвищення надійності їх бойового застосування.

Розроблена система класифікації та рекомендації щодо вибору БПЛА створюють надійну основу для прийняття рішень у військовому плануванні та оперативному застосуванні безпілотних літальних апаратів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Такініна О. М. Classification of Modern Unmanned Aerial Vehicles / О. М. Такініна, А. І. Лисенко, В. О. Кутєпов // *Electronics and Control Systems*. – 2022. – № 4(74). – P. 79–86. – DOI: 10.18372/1990-5548.74.17354.
2. Handbook of Unmanned Aerial Vehicles / ed. by K. P. Valavanis, G. J. Vachtsevanos. – Dordrecht : Springer, 2015. – 3030 p.
3. Criollo L. Classification, military applications, and opportunities of unmanned aerial vehicles / L. Criollo, C. Mena-Arciniega, S. Xing // *Aviation*. – 2024. – Vol. 28, Issue 2. – P. 115–127. – DOI: 10.3846/aviation.2024.21672.
4. Vajda A. New Complex Military and Civil UAV/RPA Classification System for Registration, Administrative Purposes, and Special Operation Preparation / A. Vajda, I. Szalkai // *Repüléstudományi Közlemények*. – 2022. – Vol. 34, Issue 3. – P. 91–107. – DOI: 10.32560/rk.2022.3.7.
5. Глотов В. Analysis of application of the UAVs for military tasks / В. Глотов, А. Гуніна, С. Князєв, В. Колесніченко, О. Прохорчук // *Сучасні досягнення геодезичної науки та виробництва*. – 2019. – Вип. 1(37). – С. 69–77.
6. Chaturvedi S. K. Comparative Review Study of Military and Civilian Unmanned Aerial Vehicles (UAVs) / S. K. Chaturvedi, R. Sekhar, S. Banerjee, H. Kamal // *INCAS Bulletin*. – 2019. – Vol. 11, Issue 3. – P. 183–198. – DOI: 10.13111/2066-8201.2019.11.3.16.
7. Hann R. UAV Database [Electronic resource] / R. Hann, J. Wallisch // *DataverseNO*. – 2020. – Mode of access: <https://doi.org/10.18710/L41IGQ>. – Date of access: 21.12.2024.
8. Zhang T. Small sample vehicle target recognition method for unmanned aerial vehicle system based on deep learning / T. Zhang, D. Wan // *Journal of Physics: Conference Series*. – 2021. – Vol. 1982. – P. 012077. – DOI: 10.1088/1742-6596/1982/1/012077.

9. Mrabet M. Machine learning algorithms applied for drone detection and classification: benefits and challenges / M. Mrabet, M. Sliti, L. B. Ammar // *Frontiers in Communications and Networks*. – 2024. – Vol. 5. – P. 1440727. – DOI: 10.3389/frcmn.2024.1440727.
10. Alzboon M. S. AI in the Sky: Developing Real-Time UAV Recognition Systems to Enhance Military Security / M. S. Alzboon, M. Alqaraleh, M. S. Al-Batah // *Data and Metadata*. – 2024. – Vol. 3. – P. 417. – DOI: 10.56294/dm2024.417.
11. Гусак Ю. А. Кластеризація та класифікація ударних безпілотних літальних апаратів на основі нейронних мереж / Ю. А. Гусак, О. А. Василенко // *Повітряна міць України*. – 2024. – № 1(6). – С. 141–152.
12. Гусак Ю. А. Алгоритм створення нейронної мережі для класифікації ударних безпілотних літальних апаратів / Ю. А. Гусак, О. А. Василенко // *Сучасні інформаційні технології у сфері безпеки та оборони*. – 2024. – № 2(50). – С. 54–68. – DOI: 10.33099/2311-7249/2024-50-2-54-68.
13. Deep learning for unmanned aerial vehicles detection: A review / N. Al-Iqubaydhi [et al.] // *Computer Science Review*. – 2024. – Vol. 51. – P. 100614. – DOI: 10.1016/j.cosrev.2023.100614.
14. Islam Khan A. Unmanned Aerial Vehicle in the Machine Learning Environment / A. Islam Khan, Y. Al-Mulla // *Procedia Computer Science*. – 2019. – Vol. 160. – P. 46–53. – DOI: 10.1016/j.procs.2019.09.442.
15. Haque A. A Comprehensive Review of Classification and Application of Machine Learning in Drone Technology / A. Haque, M. Chowdhury, M. Hassanalian // *Preprints*. – 2023. – DOI: 10.20944/preprints202306.1901.v1.
16. Hastie T. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* / T. Hastie, R. Tibshirani, J. Friedman. – 2nd ed. – New York : Springer, 2009. – 745 p. – (Springer Series in Statistics). – DOI: 10.1007/978-0-387-84858-7.
17. Chen T. XGBoost: A Scalable Tree Boosting System / T. Chen, C. Guestrin // *Proceedings of the 22nd ACM SIGKDD International Conference on*

Knowledge Discovery and Data Mining. – San Francisco, 2016. – P. 785–794. – DOI: 10.1145/2939672.2939785.

18. Heaton J. Deep learning (Ian Goodfellow, Yoshua Bengio and Aaron Courville) / J. Heaton // Genetic Programming and Evolvable Machines. – 2018. – Vol. 19. – P. 305–307. – DOI: 10.1007/s10710-017-9314-z.

ДОДАТОК 1

ПРОГРАМНИЙ КОД КЛАСИФІКАЦІЇ БПЛА

Automatically generated by Colab.

Original file is located at

<https://colab.research.google.com/drive/16o3w0ngBGY0Tr0LYQ5PvosCu4bhWh5g5>

```
# **Завантаження набору даних**
```

```
"""
```

```
# Підключаємо необхідні бібліотеки
```

```
import pandas as pd
```

```
import numpy as np
```

```
# Завантаження Excel-файлу
```

```
from google.colab import files
```

```
uploaded = files.upload() # Завантажуємо файл через Colab
```

```
# Читання Excel-файлу
```

```
file_name = list(uploaded.keys())[0] # Отримуємо ім'я завантаженого файлу
```

```
data = pd.ExcelFile(file_name)
```

```
# Виведення доступних аркушів
```

```
print("Доступні листи в файлі:", data.sheet_names)
```

```
# Завантаження потрібного аркуша
```

```
df = data.parse('UAV_Classifications')
```

```
# Перегляд перших рядків таблиці
```

```
df.head(3)
```

```
"""\n# **Аналіз пропущених значень**\n"""
```

```
# Аналіз пропущених значень
```

```
missing_data = df.isnull().sum()
```

```
print("Кількість пропущених значень у кожній колонці:")
```

```
print(missing_data)
```

```
# Відсоток пропущених значень
```

```
missing_percent = (missing_data / len(df)) * 100
```

```
print("\nВідсоток пропущених значень:")
```

```
print(missing_percent)
```

```
# Описова статистика для числових колонок
```

```
print("\nОписова статистика числових даних:")
```

```
print(df.describe())
```

```
"""\n# **Обробка пропущених значень**\n"""
```

```

# Константа для перетворення між метрами і футами
conversion_factor = 3.28084

# Визначення числових колонок
numerical_columns = ['MTOW [kg]', 'Payload [kg]', 'Wingspan [m]', 'Length [m]',
                    'Cruise Speed [m/s]', 'Max speed [m/s]', 'Range [km]',
                    'Endurance [h]', 'Altitude [m]', 'Altitude [ft]']

# Перетворення значень колонок на числовий тип, некоректні значення замінюються на NaN
df['Altitude [m]'] = pd.to_numeric(df['Altitude [m]'], errors='coerce')
df['Altitude [ft]'] = pd.to_numeric(df['Altitude [ft]'], errors='coerce')

# Заповнення пропусків у колонках Altitude [m] та Altitude [ft]
df['Altitude [m]'] = df['Altitude [m]'].fillna(df['Altitude [ft]'] / conversion_factor)
df['Altitude [ft]'] = df['Altitude [ft]'].fillna(df['Altitude [m]'] * conversion_factor)

# Перетворення числових колонок: некоректні значення замінюються на NaN
for col in numerical_columns:
    df[col] = pd.to_numeric(df[col], errors='coerce')

# Заповнення пропусків у числових колонках медіанним значенням
df[numerical_columns] = df[numerical_columns].fillna(df[numerical_columns].median())

# Видалення колонки з надто великою кількістю пропусків
if 'Stall Speed [m/s]' in df.columns:
    df = df.drop(columns=['Stall Speed [m/s]'])

# Перевірка наявності пропущених значень
remaining_missing = df.isnull().sum()

# Додатковий контроль, щоб впевнитись, що всі пропуски оброблені
if remaining_missing.sum() == 0:
    print("Усі пропуски успішно оброблені!")
else:
    print("Залишились пропуски в наступних колонках:")
    print(remaining_missing[remaining_missing > 0])

# Підсумкова перевірка
assert remaining_missing.sum() == 0, "Деякі пропуски залишились після обробки!"

"""# **Нормалізація числових змінних та розділення набору даних**"""

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

# Вибір числових ознак для нормалізації
features = ['MTOW [kg]', 'Payload [kg]', 'Wingspan [m]', 'Length [m]',
           'Cruise Speed [m/s]', 'Max speed [m/s]', 'Range [km]',
           'Endurance [h]', 'Altitude [m]']

# Вибір цільової змінної
target = 'NATO_Detailed_Category'

```

```

# Розділення даних на ознаки (X) та цільову змінну (y)
X = df[features]
y = df[target]

# Нормалізація числових ознак
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Розділення на тренувальну і тестову вибірки (80% тренування, 20% тестування)
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
random_state=42, stratify=y)

# Перевірка розмірів вибірок
X_train.shape, X_test.shape, y_train.shape, y_test.shape

"""# **Логістична регресія**"""

from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score

# Навчання моделі логістичної регресії
log_reg = LogisticRegression(max_iter=1000, random_state=42)
log_reg.fit(X_train, y_train)

# Прогнозування на тестовій вибірці
y_pred = log_reg.predict(X_test)

# Оцінка результатів
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

accuracy, report

"""**Балансування даних**"""

from imblearn.over_sampling import RandomOverSampler
from collections import Counter
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score

# Балансування даних за допомогою Random Oversampler
ros = RandomOverSampler(random_state=42)
X_resampled, y_resampled = ros.fit_resample(X_scaled, y)

# Перевірка нового розподілу класів
print("Розподіл класів до балансування:", Counter(y))
print("Розподіл класів після балансування:", Counter(y_resampled))

# Розділення даних на тренувальну і тестову вибірки після балансування
X_train_balanced, X_test_balanced, y_train_balanced, y_test_balanced = train_test_split(
    X_resampled, y_resampled, test_size=0.2, random_state=42, stratify=y_resampled
)

```

```

# Навчання моделі логістичної регресії з балансованими даними
log_reg_balanced = LogisticRegression(max_iter=1000, random_state=42)
log_reg_balanced.fit(X_train_balanced, y_train_balanced)

# Прогнозування на тестовій вибірці
y_pred_balanced = log_reg_balanced.predict(X_test_balanced)

# Оцінка результатів
accuracy_balanced = accuracy_score(y_test_balanced, y_pred_balanced)
report_balanced = classification_report(y_test_balanced, y_pred_balanced)

# Виведення результатів
print("Точність моделі після балансування даних:", accuracy_balanced)
print("Звіт класифікації після балансування даних:\n", report_balanced)

from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns

# Матриця плутанини для Logistic Regression
cm = confusion_matrix(y_test_balanced, y_pred_balanced)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=log_reg_balanced.classes_,
yticklabels=log_reg_balanced.classes_)
plt.title("Матриця плутанини Logistic Regression")
plt.xlabel('Прогнозовані класи')
plt.ylabel('Справжні класи')
plt.show()

from sklearn.metrics import classification_report

# Класифікаційний звіт
report_balanced = classification_report(y_test_balanced, y_pred_balanced)
print("Звіт класифікації після балансування даних:\n", report_balanced)

from sklearn.metrics import roc_curve, auc
from sklearn.preprocessing import label_binarize

# Бінаризація міток для багатокласової задачі
y_true_bin = label_binarize(y_test_balanced,
classes=np.arange(len(log_reg_balanced.classes_)))
y_pred_proba = log_reg_balanced.predict_proba(X_test_balanced)

# Обчислення ROC для кожної моделі
fpr, tpr, _ = roc_curve(y_true_bin[:, 0], y_pred_proba[:, 0]) # Для кожного класу
roc_auc = auc(fpr, tpr)

# Візуалізація AUC-ROC
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, label=f'Logistic Regression (AUC = {roc_auc:.2f})')
plt.plot([0, 1], [0, 1], 'k--')

```

```

plt.title('AUC-ROC для Logistic Regression')
plt.xlabel('Ложні позитивні')
plt.ylabel('Правдиві позитивні')
plt.legend(loc='lower right')
plt.show()

"""# **Random Forest**"""

from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score

# Розділення даних на тренувальну і тестову вибірки без балансування
X_train_unbalanced, X_test_unbalanced, y_train_unbalanced, y_test_unbalanced =
train_test_split(
    X_scaled, y, test_size=0.2, random_state=42, stratify=y
)

# Навчання моделі Random Forest без балансування
rf_classifier_unbalanced = RandomForestClassifier(random_state=42, n_estimators=100)
rf_classifier_unbalanced.fit(X_train_unbalanced, y_train_unbalanced)

# Прогнозування на тестовій вибірці
y_pred_rf_unbalanced = rf_classifier_unbalanced.predict(X_test_unbalanced)

# Оцінка результатів
accuracy_rf_unbalanced = accuracy_score(y_test_unbalanced, y_pred_rf_unbalanced)
report_rf_unbalanced = classification_report(y_test_unbalanced, y_pred_rf_unbalanced)

# Виведення результатів
print("Точність моделі Random Forest без балансування даних:", accuracy_rf_unbalanced)
print("Звіт класифікації для Random Forest без балансування:\n", report_rf_unbalanced)

from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score
from imblearn.over_sampling import RandomOverSampler
from collections import Counter

# Повторне балансування даних за допомогою Random Oversampler
ros = RandomOverSampler(random_state=42)
X_resampled, y_resampled = ros.fit_resample(X_scaled, y)

# Перевірка нового розподілу класів
print("Розподіл класів до балансування:", Counter(y))
print("Розподіл класів після балансування:", Counter(y_resampled))

# Розділення на тренувальну і тестову вибірки після балансування
X_train_balanced, X_test_balanced, y_train_balanced, y_test_balanced = train_test_split(
    X_resampled, y_resampled, test_size=0.2, random_state=42, stratify=y_resampled
)

# Навчання моделі Random Forest
rf_classifier = RandomForestClassifier(random_state=42, n_estimators=100)
rf_classifier.fit(X_train_balanced, y_train_balanced)

```

```

# Прогнозування на тестовій вибірці
y_pred_rf = rf_classifier.predict(X_test_balanced)

# Оцінка результатів
accuracy_rf = accuracy_score(y_test_balanced, y_pred_rf)
report_rf = classification_report(y_test_balanced, y_pred_rf)

# Виведення результатів
print("Точність моделі Random Forest після балансування даних:", accuracy_rf)
print("Звіт класифікації для Random Forest:\n", report_rf)

from sklearn.model_selection import cross_val_score
import numpy as np

# Використання Random Forest із крос-валідацією
rf_classifier_cv = RandomForestClassifier(random_state=42, n_estimators=100,
class_weight='balanced')

# Виконання крос-валідації
cv_scores = cross_val_score(rf_classifier_cv, X_scaled, y, cv=5, scoring='accuracy')

# Результати крос-валідації
print("Результати крос-валідації:", cv_scores)
print("Середня точність:", np.mean(cv_scores))
print("Стандартне відхилення:", np.std(cv_scores))

import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix

# Матриця плутанини для Random Forest
cm_rf = confusion_matrix(y_test_balanced, y_pred_rf)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm_rf, annot=True, fmt='d', cmap='Blues', xticklabels=rf_classifier.classes_,
yticklabels=rf_classifier.classes_)
plt.title("Матриця плутанини для Random Forest")
plt.xlabel('Прогнозовані класи')
plt.ylabel('Справжні класи')
plt.show()

from sklearn.metrics import classification_report

# Класифікаційний звіт
report_rf = classification_report(y_test_balanced, y_pred_rf)
print("Звіт класифікації для Random Forest:\n", report_rf)

from sklearn.metrics import roc_curve, auc
from sklearn.preprocessing import label_binarize

# Бінаризація міток для багатокласової задачі

```

```

y_true_bin = label_binarize(y_test_balanced,
classes=np.arange(len(rf_classifier.classes_)))
y_pred_rf_proba = rf_classifier.predict_proba(X_test_balanced)

# Обчислення ROC для кожної моделі
fpr_rf, tpr_rf, _ = roc_curve(y_true_bin[:, 0], y_pred_rf_proba[:, 0])
roc_auc_rf = auc(fpr_rf, tpr_rf)

# Візуалізація AUC-ROC для Random Forest
plt.figure(figsize=(8, 6))
plt.plot(fpr_rf, tpr_rf, label=f'Random Forest (AUC = {roc_auc_rf:.2f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.title('AUC-ROC для Random Forest')
plt.xlabel('Ложні позитивні')
plt.ylabel('Правдиві позитивні')
plt.legend(loc='lower right')
plt.show()

"""# **XGBoost**"""

from sklearn.preprocessing import LabelEncoder
import xgboost as xgb
from sklearn.metrics import classification_report, accuracy_score

# Перетворення категоріальних міток на числові
label_encoder = LabelEncoder()
y_train_unbalanced_encoded = label_encoder.fit_transform(y_train_unbalanced)
y_test_unbalanced_encoded = label_encoder.transform(y_test_unbalanced)

# Створення та тренування моделі XGBoost на незбалансованих даних
xgb_classifier_unbalanced = xgb.XGBClassifier(random_state=42, use_label_encoder=False,
eval_metric='mlogloss')

# Навчання моделі
xgb_classifier_unbalanced.fit(X_train_unbalanced, y_train_unbalanced_encoded)

# Прогнозування на тестовій вибірці
y_pred_xgb_unbalanced = xgb_classifier_unbalanced.predict(X_test_unbalanced)

# Оцінка результатів
accuracy_xgb_unbalanced = accuracy_score(y_test_unbalanced_encoded, y_pred_xgb_unbalanced)
report_xgb_unbalanced = classification_report(y_test_unbalanced_encoded,
y_pred_xgb_unbalanced)

# Виведення результатів
print("Точність моделі XGBoost без балансування даних:", accuracy_xgb_unbalanced)
print("Звіт класифікації для XGBoost без балансування:\n", report_xgb_unbalanced)

from sklearn.preprocessing import LabelEncoder
import xgboost as xgb
from sklearn.metrics import classification_report, accuracy_score
from imblearn.over_sampling import RandomOverSampler
from collections import Counter

```

```

# Балансування даних за допомогою RandomOversampler
ros = RandomOverSampler(random_state=42)
X_resampled, y_resampled = ros.fit_resample(X_scaled, y)

# Перевірка нового розподілу класів
print("Розподіл класів до балансування:", Counter(y))
print("Розподіл класів після балансування:", Counter(y_resampled))

# Кодування категоріальних міток на числові значення
label_encoder = LabelEncoder()
y_resampled_encoded = label_encoder.fit_transform(y_resampled)

# Розділення на тренувальну і тестову вибірки після балансування
X_train_balanced, X_test_balanced, y_train_balanced, y_test_balanced = train_test_split(
    X_resampled, y_resampled_encoded, test_size=0.2, random_state=42,
    stratify=y_resampled_encoded
)

# Навчання моделі XGBoost на збалансованих даних
xgb_classifier_balanced = xgb.XGBClassifier(random_state=42, use_label_encoder=False,
eval_metric='mlogloss')
xgb_classifier_balanced.fit(X_train_balanced, y_train_balanced)

# Прогнозування на тестовій вибірці
y_pred_xgb_balanced = xgb_classifier_balanced.predict(X_test_balanced)

# Оцінка результатів
accuracy_xgb_balanced = accuracy_score(y_test_balanced, y_pred_xgb_balanced)
report_xgb_balanced = classification_report(y_test_balanced, y_pred_xgb_balanced)

# Виведення результатів
print("Точність моделі XGBoost після балансування даних:", accuracy_xgb_balanced)
print("Звіт класифікації для XGBoost після балансування:\n", report_xgb_balanced)

import matplotlib.pyplot as plt
import xgboost as xgb

# Отримання важливості ознак з моделі XGBoost
xgb_classifier_balanced = xgb.XGBClassifier(random_state=42, use_label_encoder=False,
eval_metric='mlogloss')
xgb_classifier_balanced.fit(X_train_balanced, y_train_balanced)

# Показ важливості ознак
xgb.plot_importance(xgb_classifier_balanced, importance_type='weight',
max_num_features=10, height=0.8)
plt.title("Важливість ознак")
plt.show()

# Якщо хочете отримати числові значення важливості
feature_importance = xgb_classifier_balanced.feature_importances_
for i, val in enumerate(feature_importance):
    print(f"{numerical_columns[i]}: {val}")

```

```

import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Візуалізація розподілу класів у цільовій змінній
plt.figure(figsize=(8, 6))
sns.countplot(x=y_resampled)
plt.title("Розподіл класів")
plt.xlabel("Класи")
plt.ylabel("Частота")
plt.show()

# Кореляційна матриця для числових ознак
correlation_matrix = df[numerical_columns].corr() # Числові ознаки

plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f')
plt.title("Кореляційна матриця")
plt.show()

from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# Застосуємо PCA для зниження до 2 вимірів
pca = PCA(n_components=2)
pca_components = pca.fit_transform(X_resampled)

# Якщо y_resampled ще не є числовим, скористаємося кодуванням міток
y_numeric = label_encoder.transform(y_resampled)

# Візуалізація результатів PCA з кольорами для кожного класу
plt.figure(figsize=(8, 6))
plt.scatter(pca_components[:, 0], pca_components[:, 1], c=y_numeric, cmap='viridis')
plt.title("PCA: Візуалізація даних після зниження вимірності до 2D")
plt.xlabel("Перша компонента")
plt.ylabel("Друга компонента")
plt.colorbar()
plt.show()

# Візуалізація поясненої дисперсії для кожної компоненти
explained_variance_ratio = pca.explained_variance_ratio_

plt.figure(figsize=(8, 6))
plt.bar(range(1, len(explained_variance_ratio) + 1), explained_variance_ratio)
plt.xlabel("Компонента")
plt.ylabel("Пояснена дисперсія")
plt.title("Пояснена дисперсія для кожної компоненти")
plt.show()

# Сумарна пояснена дисперсія для перших N компонент
cumulative_variance = np.cumsum(explained_variance_ratio)
print(f"Сумарна пояснена дисперсія для перших 2 компонент: {cumulative_variance[1]:.2f}")

```

```

import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix

# Матриця плутанини для XGBoost
cm_xgb_balanced = confusion_matrix(y_test_balanced, y_pred_xgb_balanced)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm_xgb_balanced, annot=True, fmt='d', cmap='Blues',
xticklabels=label_encoder.classes_, yticklabels=label_encoder.classes_)
plt.title("Матриця плутанини для XGBoost")
plt.xlabel('Прогнозовані класи')
plt.ylabel('Справжні класи')
plt.show()

from sklearn.metrics import classification_report

# Класифікаційний звіт для XGBoost
report_xgb_balanced = classification_report(y_test_balanced, y_pred_xgb_balanced)
print("Звіт класифікації для XGBoost після балансування даних:\n", report_xgb_balanced)

from sklearn.metrics import roc_curve, auc
from sklearn.preprocessing import label_binarize

# Бінаризація міток для багатокласової задачі
y_true_bin = label_binarize(y_test_balanced,
classes=np.arange(len(label_encoder.classes_)))
y_pred_xgb_proba = xgb_classifier_balanced.predict_proba(X_test_balanced)

# Обчислення ROC для кожної моделі
fpr_xgb, tpr_xgb, _ = roc_curve(y_true_bin[:, 0], y_pred_xgb_proba[:, 0])
roc_auc_xgb = auc(fpr_xgb, tpr_xgb)

# Візуалізація AUC-ROC для XGBoost
plt.figure(figsize=(8, 6))
plt.plot(fpr_xgb, tpr_xgb, label=f'XGBoost (AUC = {roc_auc_xgb:.2f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.title('AUC-ROC для XGBoost')
plt.xlabel('Ложні позитивні')
plt.ylabel('Правдиві позитивні')
plt.legend(loc='lower right')
plt.show()

"""# **Нейронна мережа**"""

from sklearn.model_selection import train_test_split
from tensorflow.keras.utils import to_categorical

# Розділення даних на тренувальні та тестові набори
X_train_nn, X_test_nn, y_train_nn, y_test_nn = train_test_split(X_resampled,
y_resampled_encoded, test_size=0.2, random_state=42)

```

```

# Перетворення міток класів у one-hot кодування
y_train_nn = to_categorical(y_train_nn, num_classes=len(label_encoder.classes_))
y_test_nn = to_categorical(y_test_nn, num_classes=len(label_encoder.classes_))

# Перевірка розмірів даних
print(f"X_train_nn.shape: {X_train_nn.shape}, X_test_nn.shape: {X_test_nn.shape}")
print(f"y_train_nn.shape: {y_train_nn.shape}, y_test_nn.shape: {y_test_nn.shape}")

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.optimizers import Adam

# Створення моделі нейронної мережі
model = Sequential()

# Вхідний шар (Input Layer)
model.add(Dense(units=64, input_dim=X_train_nn.shape[1], activation='relu'))

# Приховані шари
model.add(Dense(units=128, activation='relu'))
model.add(Dropout(0.2)) # Dropout для запобігання перенавчанню

# Вихідний шар
model.add(Dense(units=len(label_encoder.classes_), activation='softmax')) #
Використовуємо softmax для багатокласової класифікації

# Компіляція моделі
model.compile(loss='categorical_crossentropy', optimizer=Adam(learning_rate=0.001),
metrics=['accuracy'])

# Огляд архітектури моделі
model.summary()

# Навчання моделі
history = model.fit(X_train_nn, y_train_nn, epochs=20, batch_size=32,
validation_data=(X_test_nn, y_test_nn))

# Оцінка на тестових даних
test_loss, test_accuracy = model.evaluate(X_test_nn, y_test_nn)
print(f"Точність на тестових даних: {test_accuracy * 100:.2f}%")

# Графік точності
plt.plot(history.history['accuracy'], label='Тренувальна точність')
plt.plot(history.history['val_accuracy'], label='Валідаційна точність')
plt.title('Точність нейронної мережі')
plt.xlabel('Епохи')
plt.ylabel('Точність')
plt.legend()
plt.show()

# Графік втрат
plt.plot(history.history['loss'], label='Тренувальні втрати')

```

```

plt.plot(history.history['val_loss'], label='Валідаційні втрати')
plt.title('Втрати нейронної мережі')
plt.xlabel('Епохи')
plt.ylabel('Втрати')
plt.legend()
plt.show()

from tensorflow.keras.callbacks import EarlyStopping

# Налаштовуємо раннє зупинення
early_stopping = EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)

# Перекомпіляція моделі з іншими параметрами (якщо необхідно)
model.compile(loss='categorical_crossentropy',
              optimizer=Adam(learning_rate=0.001),
              metrics=['accuracy'])

# Навчання моделі з раннім зупиненням
history = model.fit(X_train_nn, y_train_nn, epochs=100, batch_size=32,
                  validation_data=(X_test_nn, y_test_nn),
                  callbacks=[early_stopping])

# Збереження моделі
model.save('optimized_neural_network.h5')

from sklearn.metrics import confusion_matrix
import seaborn as sns

# Прогнозування на тестових даних
y_pred_nn = model.predict(X_test_nn)

# Отримання класів
y_pred_labels = np.argmax(y_pred_nn, axis=1)
y_true_labels = np.argmax(y_test_nn, axis=1)

# Матриця плутанини
cm = confusion_matrix(y_true_labels, y_pred_labels)

# Візуалізація матриці плутанини
plt.figure(figsize=(8, 6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=label_encoder.classes_,
yticklabels=label_encoder.classes_)
plt.title("Матриця плутанини")
plt.xlabel('Прогнозовані класи')
plt.ylabel('Справжні класи')
plt.show()

from sklearn.metrics import classification_report

# Класифікаційний звіт
report_nn = classification_report(y_true_labels, y_pred_labels,
target_names=label_encoder.classes_)
print("Класифікаційний звіт:")

```

```

print(report_nn)

from sklearn.metrics import roc_curve, auc
from sklearn.preprocessing import label_binarize

# Перетворення реальних значень для багатокласової задачі
y_true_bin = label_binarize(y_true_labels, classes=np.arange(len(label_encoder.classes_)))

# Обчислення ROC для кожного класу
fpr, tpr, _ = roc_curve(y_true_bin[:, 0], y_pred_nn[:, 0]) # Для кожного класу

# Візуалізація ROC-кривих
plt.figure(figsize=(8, 6))
for i in range(len(label_encoder.classes_)):
    fpr, tpr, _ = roc_curve(y_true_bin[:, i], y_pred_nn[:, i])
    roc_auc = auc(fpr, tpr)
    plt.plot(fpr, tpr, label=f'Клас {label_encoder.classes_[i]} (AUC = {roc_auc:.2f})')

plt.plot([0, 1], [0, 1], 'k--')
plt.title('ROC-криві для кожного класу')
plt.xlabel('Ложні позитивні')
plt.ylabel('Правдиві позитивні')
plt.legend(loc='lower right')
plt.show()

# Графік точності
plt.plot(history.history['accuracy'], label='Тренувальна точність')
plt.plot(history.history['val_accuracy'], label='Валідаційна точність')
plt.title('Точність нейронної мережі')
plt.xlabel('Епохи')
plt.ylabel('Точність')
plt.legend()
plt.show()

# Графік втрат
plt.plot(history.history['loss'], label='Тренувальні втрати')
plt.plot(history.history['val_loss'], label='Валідаційні втрати')
plt.title('Втрати нейронної мережі')
plt.xlabel('Епохи')
plt.ylabel('Втрати')
plt.legend()
plt.show()

```