

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ

В. о. завідувача кафедри інтелектуальних
інформаційних систем

_____ Євген СІДЕНКО
« ____ » _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА
КЛАСИФІКАЦІЯ РАКОВИХ КЛІТИН ЗА
ГІСТОЛОГІЧНИМИ ЗОБРАЖЕННЯМИ НА ОСНОВІ
МЕТОДІВ МАШИННОГО НАВЧАННЯ

Спеціальність 122 Комп'ютерні науки
Освітня програма «Інтелектуальні інформаційні системи»

Здобувач

_____ Ілля ДИРДА
« ____ » _____ 2025 р.

Керівник канд. техн. наук, доцент

_____ Євген СІДЕНКО
« ____ » _____ 2025 р.

Чорноморський національний університет імені Петра Могили
(повне найменування закладу вищої освіти)

Факультет	Комп'ютерних наук
Кафедра	Інтелектуальних інформаційних систем
Рівень вищої освіти	Другий (магістерський)
Освітній ступень	Магістр
Спеціальність	Комп'ютерні науки
Освітня програма	Інтелектуальні інформаційні системи

ЗАТВЕРДЖУЮ

В. о. завідувача кафедри інтелектуальних
інформаційних систем

_____ Євген СІДЕНКО

«_____» _____ 2025 р.

ЗАВДАННЯ
на кваліфікаційну роботу здобувача

Дирда Ілля Юрійович

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: «Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання».

Керівник роботи: Сіденко Євген Вікторович, в. о. завідувача кафедри інтелектуальних інформаційних систем, канд. техн. наук, доцент.

Затверджена наказом ЧНУ ім. Петра Могили від «7» липня 2025 р. № 184.

2. Строк представлення кваліфікаційної роботи «16» грудня 2025 р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні: Публічний набір даних гістопатологічних зображень раку молочної залози BreakHis (Breast Cancer Histopathological Database), що містить 9 109 зображень у форматі PNG з роздільною здатністю 700×460 пікселів, отриманих при збільшенні 40×, 100×, 200× та 400×, з розміткою на два основні класи (доброякісні та злоякісні) та вісім гістологічних підтипів. Очікуваний результат: інтелектуальна система

автоматичної бінарної класифікації гістологічних зображень тканини молочної залози як доброякісних або злоякісних на основі згорткових нейронних мереж з веб-інтерфейсом для практичного використання в патологоанатомічних лабораторіях.

4. Перелік питань, що підлягають розробці: аналіз предметної області цифрової патології та сучасних підходів до автоматизованої класифікації гістологічних зображень раку молочної залози; огляд і порівняльний аналіз архітектур згорткових нейронних мереж (ResNet, EfficientNet, DenseNet) для задач комп'ютерного зору в медицині; дослідження публічного набору даних BreakHis, аналіз його структури, особливостей, переваг та обмежень; розробка конвеєру попередньої обробки зображень (нормалізація кольору за методом Масенко, аугментація даних за допомогою Albumentations); реалізація та порівняння моделей класифікації з використанням трансферного навчання; розробка програмного комплексу на мові Python з використанням фреймворків PyTorch, Torchvision, Streamlit, включаючи веб-інтерфейс користувача та візуалізацію результатів (Grad-CAM); експериментальне дослідження ефективності розроблених моделей, оцінка за метриками accuracy, precision, recall, F1-score, AUC-ROC. розробка програмного забезпечення для генерування музичних мелодій.

5. Перелік графічних матеріалів: 23 зображення, 2 таблиці, презентація.

Керівник роботи

(Особистий підпис)

Євген СІДЕНКО
(Власне ім'я ПРІЗВИЩЕ)

Здобувач

(Особистий підпис)

Ілля ДИРДА
(Власне ім'я ПРІЗВИЩЕ)

Дата видачі завдання «28» червня 2025 р.

КАЛЕНДАРНИЙ ПЛАН кваліфікаційної роботи

Тема: Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

№	Найменування роботи	Початок	Закінчення	Примітки
1	Отримання завдання на виконання КР	27.06.2025	30.06.2025	
2	Аналіз предметної області та постановка задачі	1.07.2025	20.07.2025	
3	Огляд літературних джерел за темою кваліфікаційної роботи, зокрема аналіз публікацій та аналогічних систем, щодо класифікації гістологічних зображень	21.07.2025	30.07.2025	
4	Огляд існуючих архітектур штучних нейронних мереж для вирішення поставленої задачі	01.09.2025	25.10.2025	
5	Реалізація обраних технологій з аналізом отриманих результатів	26.10.2025	21.11.2025	
6	Перший попередній захист КР на засіданні комісії кафедри	25.11.2025	26.11.2025	
7	Корегування роботи за результатами попереднього захисту	27.11.2025	05.12.2025	
8	Другий попередній захист КР на засіданні комісії кафедри	09.12.2025	09.12.2025	
9	Доробка та остаточне оформлення КР	10.12.2025	12.12.2025	
10	Подання КР, її електронної копії та інших документів (відгуку, рецензії) до захисту	15.12.2025	16.12.2025	

Керівник роботи

(Особистий підпис)

Євген СІДЕНКО
(Власне ім'я ПРІЗВИЩЕ)

Здобувач

(Особистий підпис)

Ілля ДИРДА
(Власне ім'я ПРІЗВИЩЕ)

Дата складання календарного плану
«08» липня 2025 р.

АНОТАЦІЯ

до кваліфікаційної роботи
здобувача групи 601м ЧНУ ім. Петра Могили

Дирди Іллі Юрійовича

на тему: “КЛАСИФІКАЦІЯ РАКОВИХ КЛІТИН ЗА ГІСТОЛОГІЧНИМИ ЗОБРАЖЕННЯМИ НА ОСНОВІ МЕТОДІВ МАШИННОГО НАВЧАННЯ”

Актуальність дослідження полягає у необхідності високоточних та ефективних інструментів для аналізу гістологічних зображень. Автоматизація цього процесу за допомогою методів машинного навчання дозволяє значно підвищити точність діагностики, зменшити навантаження на лікарів-патоморфологів та мінімізувати вплив суб'єктивного фактору.

Об'єктом дослідження є процеси автоматизованої класифікації гістологічних зображень для діагностики ракових захворювань.

Предметом дослідження є алгоритми машинного навчання для класифікації зображень ракових клітин.

Метою дослідження є розробка інтелектуальної системи, здатної автоматично класифікувати гістологічні зображення клітин молочної залози як доброякісні або злоякісні з високою точністю.

Пояснювальна записка складається зі вступу, чотирьох розділів, висновку та додатків. У першому розділі проведено аналіз предметної сфери, досліджено сучасні підходи та існуючі рішення в області автоматизованої класифікації гістологічних зображень, а також сформовано постановку задачі. У другому розділі розглянуто алгоритми, моделі, методи та інформаційні технології для вирішення поставленої задачі, включаючи архітектури згорткових нейронних мереж (ResNet, EfficientNet, DenseNet), методологію трансферного навчання, систему попередньої обробки та аугментації даних. У третьому розділі описано розробку системи, включаючи детальний опис набору даних BreakHis та програмну реалізацію системи. У четвертому розділі проведено аналіз отриманих результатів, описано керівництво користувача та результати тестування.

В результаті було розроблено інтелектуальну систему класифікації гістологічних зображень на основі згорткових нейронних мереж, яка досягла високої точності і може використовуватися в медичних науково-дослідних установах та патологічних лабораторіях для підтримки прийняття діагностичних рішень.

Кваліфікаційна робота містить 118 сторінок, 23 рисунки, 2 таблиці, 47 використаних джерел та 2 додатки.

Ключові слова: класифікація, ракові клітини, гістологічні зображення, машинне навчання, згорткові нейронні мережі, цифрова патологія.

ABSTRACT

to the qualification work by the student of the group 601m of Petro Mohyla Black Sea National University

Dyrda Illia

“CLASSIFICATION OF CANCER CELLS BY HISTOLOGICAL IMAGES BASED ON MACHINE LEARNING METHODS”

The relevance of the study lies in the need for highly accurate and effective tools for analyzing histological images. Automating this process using machine learning methods can significantly improve diagnostic accuracy, reduce the workload on pathologists, and minimize the influence of subjective factors.

The object of the study is the processes of automated classification of histological images for the diagnosis of cancer.

The subject of the study is machine learning algorithms for the classification of cancer cell images.

The goal of the study is to develop an intelligent system capable of automatically classifying histological images of breast cells as benign or malignant with high accuracy.

The explanatory note consists of an introduction, four chapters, a conclusion, and appendices. The first chapter analyzes the subject area, examines current approaches and existing solutions in the field of automated classification of histological images, and formulates the problem. The second chapter discusses algorithms, models, methods, and information technologies for solving the problem, including convolutional neural network architectures (ResNet, EfficientNet, DenseNet), transfer learning methodology, and a system for preprocessing and augmenting data. The third chapter describes the development of the system, including a detailed description of the BreakHis dataset and the software implementation of the system. The fourth chapter analyzes the results obtained, describes the user manual, and presents the test results.

As a result, an intelligent system for classifying histological images based on convolutional neural networks was developed, which achieved high accuracy and can be used in medical research institutions and pathology laboratories to support diagnostic decision-making.

The thesis contains 118 pages, 23 figures, 2 tables, 47 references, and 2 appendices.

Keywords: classification, cancer cells, histological images, machine learning, convolutional neural networks, digital pathology.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	8
ВСТУП	10
1 АНАЛІЗ ПРЕДМЕТНОЇ СФЕРИ. ПОСТАНОВКА ЗАДАЧІ	12
1.1 Детальний опис предметної області та історичний контекст	12
1.2 Ключові поняття, процеси та учасники	14
1.3 Проблеми та виклики в предметній сфері	17
1.4 Огляд та аналіз наявних аналогів і публікацій	21
1.5 Постановка задачі	26
Висновки до розділу 1	29
2 МОДЕЛІ, МЕТОДИ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ	32
2.1 Методи для вирішення поставленої задачі	32
2.2 Технології розробки системи	39
Висновки до розділу 2	45
3 РОЗРОБКА СИСТЕМИ ТА ОПИС НАБОРУ ДАНИХ	47
3.1 Опис вхідних даних та структури системи	47
3.2 Структура інтелектуальної системи	56
3.3 Опис програмної реалізації	60
Висновки до розділу 3	70
4 АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ І ТЕСТУВАННЯ	72
4.1 Аналіз результатів навчання і тестування	72
4.2 Керівництво користувача	78
Висновки до розділу 4	81
ВИСНОВКИ	83
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	85
ДОДАТОК А Лістинг коду тренування моделі класифікації	91
ДОДАТОК Б Лістинг коду веб-інтерфейсу класифікації	106

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

AI – Artificial Intelligence

AUC – Area Under the Curve

BreaKHis – Breast Cancer Histopathological Database

CCD – Charge-Coupled Device

CNN – Convolutional Neural Networks

CPU – Central Processing Unit

DICOM – Digital Imaging and Communications in Medicine

E2E – End-to-End

FDA – Food and Drug Administration

FN – False Negative

FP – False Positive

Grad-CAM – Gradient-weighted Class Activation Mapping

GPU – Graphics Processing Unit

H&E – Hematoxylin and Eosin

HIS – Hospital Information System

IEC – International Electrotechnical Commission

JSON – JavaScript Object Notation

LIS – Laboratory Information System

LR – Learning Rate

MDR – Medical Device Regulation

PACS – Picture Archiving and Communication System

ReLU – Rectified Linear Unit

ResNet – Residual Network

RGB – Red, Green, Blue

ROC – Receiver Operating Characteristic

TCGA – The Cancer Genome Atlas

TN – True Negative

TP – True Positive

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

UAT – User Acceptance Testing

UML – Unified Modeling Language

ВСТУП

Розвиток сучасної медицини та охорони здоров'я тісно пов'язаний із впровадженням інноваційних цифрових технологій, зокрема методів штучного інтелекту та машинного навчання. Онкологічні захворювання, зокрема рак молочної залози, залишаються однією з найактуальніших проблем суспільства, що потребують як ранньої, так і точної діагностики для ефективного лікування. Гістологічне дослідження, будучи «золотим стандартом» у діагностиці раку, ґрунтується на візуальному аналізі тканинних зрізів лікарем-патоморфологом, що є трудомістким, часозатратним і може залежати від суб'єктивного фактору.

Сучасні досягнення в галузі комп'ютерного зору та глибокого навчання відкривають нові можливості для автоматизації та вдосконалення цього процесу. Методи машинного навчання, зокрема згорткові нейронні мережі, демонструють видатні результати в задачах аналізу та класифікації медичних зображень. Вони здатні навчатися на великих обсягах даних, виявляти складні візуальні патерни, характерні для патологічних змін, і тим самим надавати лікарям потужний інструмент підтримки прийняття рішень.

Об'єкт дослідження: процес автоматизованої класифікації гістологічних зображень для діагностики ракових захворювань.

Предмет дослідження: алгоритми машинного навчання для класифікації зображень ракових клітин.

Мета дослідження: розробка інтелектуальної системи, здатної автоматично класифікувати гістологічні зображення клітин молочної залози як доброякісні або злоякісні з високою точністю.

Для досягнення поставленої мети необхідно виконати такі завдання:

– проаналізувати сучасні підходи та існуючі рішення в області автоматизованої класифікації гістологічних зображень;

– провести огляд та вибір архітектур глибоких нейронних мереж, найбільш придатних для вирішення поставленої задачі;

– сформувати та підготувати навчальний набір даних (наприклад, на базі публічного набору BreaKHis), включаючи попередню обробку та аугментацію зображень;

– розробити модель класифікації на основі згорткових нейронних мереж;

– реалізувати програмний прототип системи з інтерфейсом для завантаження зображень та відображення результатів класифікації;

– провести експериментальне дослідження та оцінити ефективність розробленої системи за стандартними метриками.

Практичне значення роботи полягає у створенні інструменту, який може бути використаний у медичних дослідницьких установах та патологічних лабораторіях для автоматизації та підвищення точності діагностики раку молочної залози. Система спрямована на зменшення навантаження на лікарів-патоморфологів, мінімізацію впливу суб'єктивного людського фактору та прискорення процесу аналізу, що в кінцевому підсумку може сприяти покращенню результатів лікування пацієнтів.

1 АНАЛІЗ ПРЕДМЕТНОЇ СФЕРИ. ПОСТАНОВКА ЗАДАЧІ

1.1 Детальний опис предметної області та історичний контекст

Однією з найважливіших проблем сучасної медицини та охорони здоров'я є діагностика та лікування онкологічних захворювань. В рамках цієї проблеми ключову роль відіграє патологічна анатомія, а саме гістологічне дослідження тканин, яке на сьогодні є золотим стандартом у встановленні діагнозу раку. Гістологічний аналіз передбачає мікроскопічне вивчення зразка тканини, попередньо забарвленого специфічними хімічними речовинами, що дозволяє візуалізувати структуру клітин та тканин, виявити наявність пухлинних змін та оцінити їхній тип, ступінь злоякісності та інші прогностичні фактори. Точність та своєчасність цього дослідження безпосередньо впливають на вибір тактики лікування та, відповідно, на подальшу долю пацієнта.

Історично гістологічна діагностика розвивалася як суто візуальна та експертна дисципліна. З моменту появи в XIX столітті технік мікротомії та барвників, таких як гематоксилін та еозин, патологоанатом, спираючись на свій досвід і знання, інтерпретував складні морфологічні картини. Процес був і залишається трудомістким, суб'єктивним і схильним до міжпостерігацької варіабельності. Два фахівці, аналізуючи один і той самий препарат, можуть дати різні оцінки, особливо у випадках прикордонних змін або рідкісних нозологій. Це створювало об'єктивну потребу в розробці об'єктивних та кількісних методів підтримки прийняття рішень. Поява цифрової патології на рубежі XX-XXI століть, що полягає в скануванні гістологічних склінь з отриманням високорозмірних цифрових зображень (whole slide images, WSI), стала революційним кроком, який трансформувал патоморфологію з аналогової в цифрову галузь. Ця цифровізація відкрила шлях для застосування методів комп'ютерного зору та машинного навчання до аналізу медичних зображень.

Саме в цьому контексті виникає предметна область даного дослідження – автоматизована класифікація ракових клітин на гістологічних зображеннях на основі методів машинного навчання. Ця область лежить на стику біомедицини, патології, комп'ютерних наук та штучного інтелекту. Її основною метою є розробка алгоритмів, здатних автоматично виявляти, сегментувати та класифікувати клітини, тканинні структури та патологічні ознаки на цифрових гістологічних зображеннях, тим самим надаючи патологоанатомам кількісні, вільно вироблені та об'єктивні інструменти для діагностики. Ключовими підзадачами тут є розпізнавання нормальних та пухлинних клітин, диференціація типів раку (наприклад, аденокарцинома *versus* плоскоклітинний рак), оцінка ступеня диференціації (градації за Глісоном для раку передміхурової залози, шкали Блюма для раку молочної залози), виявлення мікрометантар з у лімфатичних вузлах тощо.

Історичний розвиток методів у цій галузі пройшов кілька етапів. На початковому етапі, ще до широкого розповсюдження глибокого навчання, дослідження базувалися на методах традиційного машинного навчання. Ці підходи включали ручне проектування ознак, де експерт визначав набір морфологічних, текстурних та колориметричних характеристик клітин і ядер (наприклад, площа, периметр, ексцентриситет, інтенсивність кольору, міри текстур за допомогою матриць співвідношень рівнів сірого). Потім ці ознаки витягувалися з тисяч клітин, і на їх основі будувалися класифікатори, такі як опорні вектори, випадкові ліси або бустингові алгоритми. Незважаючи на певний успіх, ці методи мали суттєві обмеження, пов'язані з залежністю від якості попередньої сегментації та суб'єктивності при виборі ознак, що ускладнювало їх загальне застосування для різноманітних типів тканин.

Справжній прорив у галузі стався з повсюдним впровадженням методів глибокого навчання, зокрема згорткових нейронних мереж, починаючи приблизно з 2012 року. Ці архітектури, натреновані на великих анотованих наборах даних, здатні автоматично виявляти ієрархічні та абстрактні ознаки

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання прямо з пікселів зображення, уникаючи необхідності ручного проектування. CNN продемонстрували надзвичайну ефективність у задачах класифікації зображень цілих патчів, сегментації ядер клітин та виявлення мітозів. Подальший розвиток привів до появи складніших архітектур, таких як U-Net для точної сегментації, мереж із увагою для виділення важливих областей, а також трансформерів, які починають застосовуватися і для аналізу гістологічних зображень. Паралельно розвивається концепція слабо контрольованого та мультиінстансного навчання, що дозволяє працювати з цілими слайдами (WSI), розмір яких перевищує десятки гігапікселів, використовуючи лише слайд-рівневі мітки, що значно знижує вимоги до трудомісткої анотації.

Таким чином, сучасна предметна область класифікації ракових клітин за гістологічними зображеннями на основі машинного навчання представляє собою динамічну та швидкозростаючу науково-прикладну дисципліну. Вона спрямована на подолання ключових викликів традиційної патології: суб'єктивності, варіабельності, втоми експерта та обробки величезних обсягів даних. Історичний контекст показує еволюцію від експертно-візуального аналізу через комп'ютеризовану морфометрію до інтелектуальних систем на основі глибокого навчання, що здатні доповнювати та покращувати роботу лікаря-патолога. Ці технології мають потенціал не лише підвищити точність та стандартизацію діагностики, але й відкрити нові горизонти в онкології, такі як виявлення раніше непомітних прогностичних маркерів, передбачення відповіді на лікування та розвиток персоналізованої терапії на основі глибокого морфологічного профілювання пухлин.

1.2 Ключові поняття, процеси та учасники

Ключові поняття предметної області формують концептуальну основу для дослідження. Гістологічне зображення є фундаментальним поняттям, що означає мікроскопічне зображення тонкого зрізу біологічної тканини,

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання спеціально обробленого та забарвленого для візуалізації її клітинної та субклітинної структури. Для аналізу методом машинного навчання таке зображення трансформується у цифрове патологічне зображення, яке є високорозмірним цифровим представленням всього гістологічного зразка, отриманим за допомогою спеціальних сканерів. Ці зображення мають надзвичайно високу роздільну здатність і називаються зображеннями цілого скління; їх розмір може сягати десятків гігапікселів, що потребує спеціальних підходів до обробки. Анотація виступає критично важливим процесом розмітки цих зображень експертом-патологоанатомом, який обводить області інтересу та призначає їм діагностичні мітки; саме ці анотовані набори даних слугують ґрунтом для навчання та валідації алгоритмів. Класифікація у цьому контексті розуміється як автоматизоване призначення зображенню або його фрагменту певної категорії, наприклад, тип пухлини або ступінь злоякісності. Наприклад, градація за Глісоном для раку передміхурової залози є одним з ключових класифікаційних систем, що кількісно оцінює агресивність пухлини на основі її архітекtonіки.

Процеси, що становлять основу роботи системи автоматизованої класифікації, утворюють складний технологічний ланцюг. Він починається з попередньої обробки цифрових зображень, яка включає корекцію кольору для усунення варіацій, спричинених різними умовами фарбування та сканування, нормалізацію яскравості та контрасту, а також розбиття величезного зображення цілого скління на менші, керовані фрагменти, звані патчами. Далі, залежно від архітектури моделі, може слідувати етап сегментації, метою якого є точне виділення меж окремих клітин або їх ядер із подальшим витягуванням їх морфометричних характеристик. Серцевиною системи є процес витягування ознак. У традиційних підходах це був ручний підбір кількісних дескрипторів форми, кольору та текстури. Сучасні методи глибокого навчання автоматизують цей процес: згорткові нейронні мережі шляхом ієрархічної обробки зображення формують абстрактні та репрезентативні вектори ознак.

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

На основі цих ознак будується класифікаційна модель, яка навчається відображати простір ознак у простір діагностичних класів. Окремим важливим процесом є інтеграція контексту, оскільки діагностичне рішення в патології рідко базується на одній ізольованій клітині; необхідно враховувати просторове розташування клітин, їх взаємодію та структуру тканини навколо них, для чого застосовуються методи мультиінстансного навчання або аналіз графів, побудованих на розташуванні клітин.

Учасники даної предметної області утворюють міждисциплінарну екосистему, кожен з яких вносить унікальний внесок. Центральною фігурою є лікар-патологоанатом або патоморфолог. Цей експерт не лише ставить клінічний діагноз на основі візуального аналізу, але й виконує ключову роль у створенні навчальних даних шляхом анотації зображень, а також у подальшій валідації та інтерпретації результатів роботи алгоритмів. Його експертні знання формують "істину в останній інстанції". З іншого боку, дослідники у галузі машинного навчання та комп'ютерного зору розробляють архітектури нейронних мереж, алгоритми оптимізації та методи обробки великих даних, адаптуючи фундаментальні наукові розробки до специфіки медичних зображень. Біоінформатики та інженери з даних часто виступають посередниками між цими двома світами; вони займаються побудовою конвеєрів обробки даних, управлінням величезними наборами зображень, статистичним аналізом результатів та інтеграцією моделей у програмне забезпечення. Важливими учасниками процесу є також клініцисти-онкологи, які визначають клінічні вимоги до системи: які саме діагностичні параметри є найбільш прогностично значимими, який рівень чутливості та специфічності є прийнятним для клінічного впровадження. Крім того, в сучасних умовах ключову роль відіграють регуляторні органи та експерти з біоетики, оскільки впровадження алгоритмів штучного інтелекту в медичну практику потребує сертифікації, доказів безпеки та ефективності, а також чіткого регулювання питань відповідальності та захисту конфіденційності пацієнтських даних.

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання
Узагальнюючи, взаємодія цих ключових понять, технологічних процесів та різнопрофільних учасників створює динамічне поле для розвитку інтелектуальних систем підтримки прийняття рішень, які покликані зміцнити ланку гістопатологічної діагностики в боротьбі з онкологічними захворюваннями.

1.3 Проблеми та виклики в предметній сфері

Предметна сфера автоматизованої аналітики гістологічних зображень зіштовхується з системними викликами, які беруть свій початок у самій суті досліджуваних даних та умов їх отримання. Перш за все, це глибока техногенна варіабельність, що пронизує весь цикл створення препарату. Від моменту взяття біопсії до отримання цифрового зображення зразок проходить через низку технічних етапів: фіксація - вибір реагенту, тривалість, проводка, заливання в парафін, зрізання мікротомом на певну товщину, фарбування - гематоксилін і еозин або імуногістохімічні методики та, нарешті, сканування. Кожен із цих кроків вносить власний рівень мінливості. Наприклад, тривалість фарбування гематоксиліном може впливати на інтенсивність фарбування ядер, а різні моделі сканерів мають унікальні оптичні характеристики та профілі кольору. Алгоритм, натренований на зображеннях з одного патологічного відділення, може катастрофічно втратити точність при застосуванні до даних іншої лабораторії, навчившись розпізнавати не морфологічні ознаки раку, а специфічні артефакти процесу обробки. Ця проблема потребує розвиненого арсеналу методів аугментації та нормалізації даних, зокрема методів перенесення стилю для уніфікації домену, проте повне її усунення залишається складною задачею.

Другим фундаментальним викликом є складність та суб'єктивність експертної анотації, яка формує "ground truth" для навчання моделей. Гістопатологічний діагноз не завжди є бінарною істиною; він часто існує в спектрі інтерпретацій. Прикордонні випадки, диференціальна діагностика між

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання атиповою гіперплазією та карциномою *in situ*, оцінка інвазії в дрібні судини — це зони, де навіть між світовими експертами можлива значна міжпостерігацька варіабельність. Коли модель навчається на таких даних, вона, по суті, навчається відтворювати середню думку конкретної групи анотаторів разом з усіма їхніми потенційними неузгодженостями. Крім того, анотація на рівні окремих клітин для великих наборів даних є практично непідйомною за трудомісткістю, що призводить до використання слабких міток, наприклад, тільки діагноз на рівні всього склінка, а це, у свою чергу, породжує проблеми в точній локалізації патологічних змін. Створення консенсусних анотацій за участю кількох експертів та розробка підходів до активного навчання, де модель саме запитує розмітку для найбільш невизначених зразків, є лише частковим вирішенням цієї глибокої методологічної проблеми.

Власна біологічна природа онкологічних захворювань породжує наступний пласт проблем. Гетерогенність пухлини — як міжпацієнтна, так і внутрішньопухлинна — є основним викликом для створення універсальних класифікаційних моделей. Дві аденокарциноми одного органу можуть мати різну морфологію, а в межах однієї пухлини можуть співіснувати клітинні клони з різним ступенем диференціації, що призводить до мозаїчної картини. Алгоритм повинен бути достатньо надійним, щоб ігнорувати цю внутрішню варіабельність, але достатньо чутливим, щоб виявляти найбільш агресивні ділянки, що мають ключове прогностичне значення. Це тісно пов'язано з проблемою необхідності інтеграції просторового та архітектурного контексту. Діагностика в патології рідко базується на ізольованій клітині. Такі ознаки, як інвазія в стромальні структури, формування залозистих утворень, характер росту та взаємне розташування клітин мають вирішальне значення. Традиційні архітектури, що аналізують окремі патчі незалежно, можуть втрачати цю важливу контекстуальну інформацію, що стимулює розвиток більш складних моделей, здатних до аналізу великомасштабних просторових залежностей,

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання наприклад, за допомогою графових нейронних мереж, де вузли представляють клітини або патчі, а ребра — просторові зв'язки між ними.

Обчислювальні виклики є іншим серйозним обмеженням. Надвеликі розміри даних (десятки гігапікселів на одне склінко) роблять неможливим пряме завантаження цілого зображення в пам'ять графічного процесора для обробки стандартними архітектурами. Це вимагає розробки спеціалізованих конвеєрів, що базуються на методиці "patch-based learning" та слабо контрольованого навчання. Однак поділ на патчі може розбивати цілісні структури, а методи мультиінстансного навчання досі є предметом активних досліджень і часто потребують складних механізмів агрегації інформації. Паралельно постає проблема енергоефективності та часу інференсу. Для клінічного застосування аналіз зразка має займати прийнятний час, що може бути проблемою для дуже глибоких і складних моделей. Оптимізація моделей для їх розгортання на обмежених апаратних ресурсах, можливо, навіть на локальних комп'ютерах в патологічних лабораторіях, є окремим напрямом робіт.

Найкритичнішим бар'єром для клінічного прийняття є проблема "чорної скриньки" в глибокому навчанні. Коли алгоритм видає діагноз, лікар-патологоанатом повинен мати змогу зрозуміти логіку, що стоїть за цим висновком. У традиційній діагностиці лікар спирається на чітко визначені критерії: форма ядра, співвідношення ядра до цитоплазми, наявність мітозів. Сучасні нейронні мережі оперують абстрактними ознаками високого рівня, які не мають прямого біологічного відповідника. Без механізмів візуальної пояснюваності (наприклад, карти ективованих областей, які показують, на які ділянки зображення модель звертала увагу) довіра до системи залишатиметься низькою. Розвиток інтерпретованого штучного інтелекту (Explainable AI, XAI) для медичних зображень, наприклад, за допомогою методів атрибуції, є не просто бажаним, а обов'язковим кроком для впровадження.

Проблема клінічної валідації та оцінки є не менш значною. Більшість досліджень демонструє результати на ретроспективних, статичних наборах даних, часто зручно підібраних і добре анотованих. Однак реальна клінічна практика — це потік різноманітних, іноді якісно незадовільних зразків, з артефактами та нестандартними випадками. Доведення того, що алгоритм не тільки демонструє високу точність на тестових наборах, але й покращує результати лікарів у проспективних, рандомізованих дослідженнях є золотим стандартом, якого досягли дуже небагато систем. Потрібні масштабні дослідження, які оцінюють не лише технічні метрики (чутливість, специфічність), а й вплив на кінцеві клінічні результати: правильність вибору терапії, виживаність пацієнтів, економічну ефективність.

Навіть технічно досконала система стикається з низкою нефармацевтичних, але вирішальних бар'єрів. Регуляторне схвалення для програмного забезпечення на основі ШІ як медичного виробу класу ІІа або вище (за класифікацією FDA або CE) є довгим, складним і дорогим процесом. Регулятори вимагають доказів безпеки, ефективності, відтворюваності та кібербезпеки. Питання відповідальності за діагноз, поставлений за підтримки алгоритму, залишається дискусійним: чи несе її розробник програмного забезпечення, лікар, який остаточно підтверджує діагноз, чи лікувальна установа? Це вимагає змін у законодавстві та формування нових професійних протоколів.

Етичні виклики включають захист конфіденційності пацієнтів при створенні великих міжінституційних наборів даних. Методи федеративного навчання, коли моделі тренуються локально на даних установ без їх передачі, є перспективним, але технічно складним шляхом вирішення цієї проблеми. Крім того, існує ризик закріплення та посилення системних упереджень. Якщо модель тренується на даних переважно однієї етнічної або демографічної групи, її продуктивність для інших груп може бути значно гіршою, що призводить до нерівності в наданні медичної допомоги.

Нарешті, ключовим є людський фактор та організаційна зміна. Впровадження ШІ в патологію вимагає перебудови робочих потоків, інтеграції з існуючими лабораторними інформаційними системами та підвищення цифрової грамотності персоналу. Може виникнути професійний спротив з боку патологоанатомів, які можуть сприймати систему як загрозу своїй експертності або додаткове навантаження. Тому розробка інтерфейсів, які працюють за принципом "AI as a colleague" (ШІ як колега), надаючи пояснення, підкреслюючи ділянки уваги та працюючи в режимі допомоги, а не авторитарного заміщення, є критично важливою для успішної адаптації технології в реальній клінічній практиці. Лише комплексне вирішення цього багаторівневого комплексу проблем — від пікселів до політики — дозволить реалізувати трансформаційний потенціал машинного навчання в гістопатології.

1.4 Огляд та аналіз наявних аналогів і публікацій

Розвиток методів аналізу гістологічних зображень пройшов три визначні етапи, кожен з яких характеризувався зміною парадигмальних підходів.

Перший етап (1970-2000 роки) був пов'язаний із застосуванням класичних методів комп'ютерного зору для виділення морфометричних ознак. Дослідницькі підходи передбачали використання алгоритмів сегментації для ізоляції ядер клітин з подальшим вимірюванням таких параметрів, як площа, периметр, ексцентриситет та інтенсивність забарвлення. Отримані ознаки служили вхідними даними для традиційних алгоритмів машинного навчання, серед яких слід відзначити метод опорних векторів, дерева рішень, наївний байєсівський класифікатор та метод k-найближчих сусідів. Принциповим обмеженням цих методів була необхідність ручного проектування ознак, що вимагало глибоких експертних знань у галузі патології та обробки зображень.

Другий етап (2000-2015 роки) ознаменувався появою глибокого навчання та згорткових нейронних мереж, які дозволили автоматизувати

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання процес виявлення ієрархічних представлень ознак. Визначною віхою стала робота Krizhevsky та співавторів [39], де архітектура AlexNet продемонструвала революційні результати в класифікації зображень ImageNet. Це відкрило нові перспективи для аналізу медичних зображень, оскільки з'явилася можливість автоматичного виявлення складних патологічних патернів без необхідності їх явного специфікування.

Сучасний етап характеризується спеціалізацією архітектур нейронних мереж для потреб медичної візуалізації та активним розвитком трансферного навчання. У цей період отримали поширення такі передові архітектури, як ResNet з механізмом залишкових зв'язків, Inception з багатомасштабною обробкою ознак, DenseNet з щільними міжшаровими з'єднаннями та EfficientNet з оптимізованим співвідношенням глибини, ширини та роздільної здатності.

Сучасні методи класифікації гістологічних зображень демонструють значне методологічне розмаїття, що дозволяє проводити їх систематизацію за різними критеріями. За рівнем аналізу розрізняють піксельний рівень, що передбачає аналіз окремих пікселів; об'єктний рівень, орієнтований на сегментацію та класифікацію окремих клітинних структур; патч-рівень, що оперує аналізом локальних областей зображення; та повнозображений рівень, що враховує глобальні гістологічні структури.

За типом архітектурних рішень виділяють одностандартні мережі, що використовують єдину архітектуру; ансамблеві методи, що поєднують прогностичні можливості кількох моделей; та багатомодальні мережі, що інтегрують різноманітні типи даних, включаючи гістологічні зображення, геномні маркери та клінічні параметри.

Особливу увагу в сучасних дослідженнях приділяється методам трансферного навчання, які дозволяють адаптувати попередньо навчені на великих наборах даних, наприклад, ImageNet моделі для вирішення специфічних медичних завдань. Цей підхід демонструє особливу ефективність

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання в умовах обмежених медичних наборів даних, що є типовими для гістологічних досліджень.

Аналіз наукової літератури дозволяє ідентифікувати ключові дослідження, що визначили сучасний стан предметної області. Дослідження Spanhol та співавторів [1] стало знаковим подією через створення публічного набору даних BreaKHis. Автори не лише представили масштабний анотований ресурс, але й провели комплексне порівняння різних методів класифікації, продемонструвавши перевагу найпростіших архітектур CNN над традиційними методами машинного навчання з перевищенням показників точності на 7-15%.

Дослідження Ciresan та співавторів [38] встановило новий стандарт у вирішенні однієї з найскладніших задач патології - виявленні мітотичних фігур. Запропонована авторами методика на основі глибоких CNN досягла точності виявлення на рівні 87%, що значно перевершило результати традиційних алгоритмів комп'ютерного зору.

Масштабне дослідження Google Health, опубліковане в Nature, продемонструвало можливість перевищення AI-системою людської експертизи у виявленні раку молочної залози на мамографічних знімках. Хоча дослідження безпосередньо не стосувалося гістології, воно підтвердило загальний потенціал глибокого навчання в онкологічній діагностиці.

В рамках конкурсу Camelyon17 показало ефективність ансамблевих підходів. Комбінування прогностичних можливостей кількох архітектур дозволило досягти значення AUC 0.99 у задачі детекції метастаз, що встановило новий стандарт якості для автоматизованої патологічної діагностики.

Аналіз ринку комерційних рішень виявляє кілька провідних платформ, що пропонують AI-рішення для цифрової патології. Система Paige.ai [42] займає лідерські позиції та отримала перше в історії схвалення FDA для AI-діагностики раку простати.



Рисунок 1.1 – Логотип компанії «Paige»

Архітектура платформи передбачає аналіз повних слайдів, інтеграцію з патологічними системами та демонструє чутливість на рівні понад 95%. Модель ліцензування передбачає річну підписку вартістю від 50 000 доларів США.

Платформа PathAI [43] орієнтована на дослідницькі та клінічні застосування, пропонуючи хмарну архітектуру, підтримку множини типів раку та інструменти для клінічних досліджень. Модель ціноутворення базується на підписній схемі з гнучкою тарифікацією залежно від обсягів використання.



Рисунок 1.2 – Логотип платформи «PathAI»

Рішення HALO [44] від Indica Labs репрезентує комплексний підхід до цифрової патології з можливістю локального впровадження, широким спектром аналітичних модулів та підтримкою різних типів фарбувань. Ліцензійна модель передбачає разову оплату від 30 000 доларів США з подальшою технічною підтримкою.



Рисунок 1.3 – Логотип платформи «HALO»

Порівняльний аналіз методологічних підходів виявляє закономірну кореляцію між складністю архітектур та їх ефективністю. Класичні методи машинного навчання демонструють точність у діапазоні 70-85% при високій інтерпретованості результатів. Базові архітектури CNN підвищують точність до 85-92% зі збереженням прийнятного рівня інтерпретованості. Сучасні глибокі архітектури досягають точності 92-98%, однак суттєво програють в інтерпретованості. Комерційні системи демонструють найвищі показники якості (95-99%), проте характеризуються найнижчою адаптивністю та найвищими вартістю впровадження.

Системний аналіз публічних ресурсів виявляє кілька ключових наборів даних, що формують інфраструктурну основу сучасних досліджень. Набір BreaKHis [40] залишається найпопулярнішим ресурсом для досліджень раку молочної залози, містить 7909 зображень з чотирма рівнями збільшення, двома

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання основними класами та вісьмома підтипами пухлин. Набір Camelyon орієнтований на задачу детекції метастаз і включає 400 повних слайдів з анотаціями високої якості. Найбільшим публічним ресурсом є The Cancer Genome Atlas (TCGA) [41], що містить понад 10 000 слайдів з 30 типами раку, доповнених клінічними та геномними даними.

На підставі проведеного аналізу можна зробити висновок про досягнення технологічної готовності методів глибокого навчання для клінічного застосування в гістологічній діагностиці. Сучасні архітектури CNN демонструють точність, порівнянну з кваліфікованими патоморфологами, проте зберігаються проблеми інтерпретованості, узагальнення на різні набори даних та інтеграції в клінічні workflows. Перспективними напрямками подальших досліджень визначаються ансамблеві методи, багатомодальне навчання та інтерактивні системи підтримки прийняття рішень.

1.5 Постановка задачі

Актуальність дослідження зумовлена потребою у розробці ефективних та економічно доступних інструментів автоматизації патологічної діагностики, здатних підвищити точність та швидкість діагностики раку молочної залози в умовах обмежених ресурсів охорони здоров'я. Епідеміологічна ситуація з раком молочної залози характеризується сталим зростанням захворюваності при одночасному дефіциті кваліфікованих патоморфологів, що створює системну потребу в інноваційних рішеннях. Сучасний статус технологій штучного інтелекту дозволяє створювати системи, здатні не лише доповнювати, але в окремих випадках і перевершувати людську експертизу в специфічних діагностичних завданнях.

Проте, незважаючи на значний прогрес у фундаментальних дослідженнях, існує відчутний розрив між експериментальними моделями, розробленими в академічному середовищі, та повноцінними, готовими до впровадження в клінічну практику рішеннями. Цей розрив обумовлений

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання комплексом факторів: відсутністю стандартизованих, легко інтегрованих програмних інструментів, недостатньою увагою до створення зручних інтерфейсів для кінцевого користувача — лікаря-патолога, а також необхідністю детального порівняльного аналізу різних архітектур у чітко визначених умовах на доступних публічних наборах даних. Таким чином, актуальність роботи полягає не лише в демонстрації принципової можливості автоматизованої класифікації, а й у системній розробці та валідації практично орієнтованого програмного комплексу, що закриває ланку між теоретичними розробками та потенційним клінічним застосуванням.

Об'єктом дослідження виступає процес автоматизованої класифікації гістологічних зображень тканини молочної залози на основі методів машинного навчання. Цей процес розглядається як послідовний ланцюг дій, що включає збір та підготовку даних, проектування та навчання моделей, оцінку їх ефективності та впровадження у вигляді інструменту, придатного для використання. Предметом дослідження є методи та алгоритми глибокого навчання, зокрема згорткові нейронні мережі, для бінарної класифікації зображень за критерієм злоякісності. Особлива увага приділяється властивостям цих архітектур, їхній здатності до автоматичного витягування морфологічних ознак із складних біологічних зображень, а також адаптації їхнього потенціалу для вирішення конкретної медичної задачі.

Метою роботи є розробка інтелектуальної системи класифікації гістологічних зображень ракових клітин на основі згорткових нейронних мереж, здатної автоматично визначати тип новоутворення з високою точністю, та її реалізація у вигляді програмного забезпечення з веб-інтерфейсом. Досягнення цієї мети має надати доказ концепції, що демонструє можливість створення локального, ефективного та доступного інструменту підтримки прийняття рішень, який може функціонувати в умовах обмеженої інфраструктури.

Першим кроком є проведення глибокого аналізу предметної області та огляду сучасних методів машинного навчання для класифікації гістологічних зображень. Це завдання включає вивчення специфіки гістології молочної залози, основних діагностичних критеріїв, а також систематизацію існуючих підходів, архітектур нейронних мереж (ResNet, EfficientNet, DenseNet) та методологій їхнього застосування в аналізі медичних зображень, що дозволить сформулювати теоретичну базу для подальшої експериментальної роботи.

Наступним завданням є вибір та ретельний аналіз публічного набору даних для навчання та оцінки моделей. Необхідно обґрунтувати вибір конкретного набору даних, оцінити його обсяг, якість, баланс класів та можливі технічні артефакти, що має вирішальне значення для відтворюваності результатів та їх подальшого порівняння з іншими дослідженнями.

Ключовим етапом є розробка конвеєру попередньої обробки та аугментації даних. Ця задача передбачає створення алгоритмів для нормалізації кольору, корекції яскравості, масштабування та розбиття великих зображень на патчі, а також впровадження методів аугментації (обертання, віддзеркалення, зміна контрасту) для штучного розширення навчальної вибірки та підвищення стійкості моделі до технічних варіацій.

Центральним експериментальним завданням є реалізація та навчання набору експериментальних моделей на основі архітектур CNN. Потрібно буде програмно реалізувати кілька сучасних архітектур, адаптувавши їх для роботи з обраним форматом даних, та провести їх навчання з використанням технік запобігання перетренуванню, таких як рання зупинка та зниження швидкості навчання.

Після навчання необхідно провести комплексну оцінку якості побудованих моделей. Ця задача полягає в розрахунку широкого спектру метрик, включаючи точність, чутливість, специфічність, F1-балу, а також побудові матриць невідповідностей та кривих ROC-AUC, що дозволить всебічно оцінити діагностичну здатність кожної моделі.

На основі отриманих оцінок слід виконати порівняльний аналіз ефективності обраних архітектур. Метою цього аналізу є виявлення переваг та недоліків кожної моделі в контексті поставленої задачі, визначення оптимального співвідношення точності та обчислювальної складності, а також інтерпретація можливих джерел помилок.

Заключним і практично значущим завданням є розробка програмного модуля з веб-інтерфейсом для інтерактивного використання. Це передбачає створення серверної частини для обробки запитів та виконання інференсу моделі, а також клієнтського веб-інтерфейсу, що дозволяє користувачеві завантажувати зображення, отримувати результат класифікації з індикацією впевненості моделі та зручно переглядати історію запитів.

У результаті виконання роботи має бути розроблено програмний комплекс на мові Python, що включає модулі попередньої обробки даних, навчені моделі-класифікатори, систему оцінки якості та функціонуючий веб-інтерфейс для практичного використання. Очікується, що система досягне показників точності не нижче 95% на тестовій вибірці набору даних BreakHis, що відповідає сучасним стандартам якості в галузі автоматизованої гістологічної діагностики та слугуватиме вагомим аргументом на користь практичної реалізації подібних інструментів. Розроблений комплекс не лише продемонструє технічну можливість автоматизації, а й надасть готовий інструмент для подальших досліджень, апробації та потенційної адаптації в умовах діагностичних лабораторій.

Висновок до розділу 1

Проведений аналіз предметної області та історичного контексту свідчить, що автоматизована класифікація ракових клітин на основі гістологічних зображень є закономірним та необхідним етапом розвитку сучасної онкоморфології. Ця дисципліна, що виникла на стику патології, комп'ютерних наук та штучного інтелекту, еволюціонувала від ручної

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання морфометрії через методи машинного навчання з виділенням ознак до сучасних складних архітектур глибокого навчання, здатних безпосередньо інтерпретувати піксельну інформацію. Цифровізація патології та поява зображень цілих склінок стали каталізатором цього процесу, відкривши шлях для аналізу великих даних у морфології. Ключовими поняттями сфери є цифрове патологічне зображення, анотація, згорткові нейронні мережі та автоматизована класифікація, а учасниками – патологоанатом, дослідник у галузі ІІІ та клініцист, взаємодія яких формує міждисциплінарну основу для розвитку.

Однак, як демонструє розгляд проблем та викликів, шлях від дослідницького прототипу до клінічного інструменту повний системних перешкод. Вони включають фундаментальні труднощі, пов'язані з техногенною варіабельністю препаратів, складністю та суб'єктивністю експертної анотації, а також властивою біологічною гетерогенністю пухлин. Методологічні виклики, такі як проблема «чорної скриньки» в глибокому навчанні, необхідність інтеграції просторового контексту та складності клінічної валідації, вимагають не лише технічних інновацій, але й розвитку нових підходів до оцінки та інтерпретації результатів. Крім того, успішна трансляція технології залежить від подолання регуляторних, етичних та соціально-професійних бар'єрів, включаючи питання відповідальності, захисту даних та адаптації медичних фахівців до роботи з інтелектуальними системами підтримки прийняття рішень.

У цьому контексті постановка задачі даної кваліфікаційної роботи формулює конкретну, практично орієнтовану науково-дослідну ціль. Актуальність роботи обґрунтовується нагальною потребою в доступних та ефективних інструментах для діагностики раку молочної залози, що відповідає на виклик дефіциту експертів та зростання захворюваності. Об'єктом та предметом дослідження чітко визначено процес автоматизованої класифікації та методи глибокого навчання для його реалізації. Мета роботи – розробка

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання інтелектуальної системи з веб-інтерфейсом – спрямована саме на подолання згаданого розриву між академічними дослідженнями та потенційним практичним застосуванням.

Визначені для досягнення мети завдання утворюють логічну та послідовну структуру дослідження: від теоретичного аналізу і вибору даних через експериментальну розробку та оцінку моделей до фінальної інтеграції у вигляді програмного комплексу. Такий підхід забезпечує комплексність роботи, де кожен етап базується на попередньому. Очікуваний результат у вигляді функціонуючого програмного забезпечення з метрикою точності не нижче 95% на стандартному наборі даних BreakHis слугуватиме конкретним, вимірюваним доказом досяжності поставленої мети.

Таким чином, представлені підрозділи в сукупності обґрунтовують наукову та практичну значущість запланованого дослідження. Воно спрямоване не лише на демонстрацію можливостей технології, але й на створення цілісного рішення, що враховує складність предметної області та системні виклики. Успішна реалізація роботи має потенціал внести внесок як у фундаментальне розуміння застосування згорткових нейронних мереж у гістопатології, так і у практичне вдосконалення діагностичного процесу шляхом надання інструменту, здатного працювати в умовах обмежених ресурсів.

2 МОДЕЛІ, МЕТОДИ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

2.1 Методи для вирішення поставленої задачі

Розробка системи автоматизованої класифікації гістологічних зображень потребує застосування комплексного методологічного підходу, який інтегрує сучасні методи машинного та глибокого навчання зі спеціалізованими техніками обробки медичних зображень. Методологічна основа даного дослідження ґрунтується на концепції трансферного навчання, що поєднує переваги попередньо навчених архітектур згорткових нейронних мереж із адаптацією до специфіки гістопатологічних даних. Цей підхід дозволяє подолати фундаментальне обмеження, пов'язане із відносно невеликим розміром медичних наборів даних порівняно із загальнодоступними колекціями зображень, такими як ImageNet. Трансферне навчання реалізується через послідовність етапів, починаючи від ініціалізації ваг мережі значеннями, отриманими в результаті навчання на загальних задачах комп'ютерного зору, і завершуючи донавчанням верхніх шарів моделі на цільовому наборі даних BraKHis. Така стратегія не лише значно прискорює процес конвергенції моделі, але й покращує її узагальнюючу здатність за рахунок використання низькорівневих ознак, таких як краї, текстури та просторові відношення, які є універсальними для широкого спектра зображень, включаючи медичні.

Окрему увагу приділено архітектурному вибору моделей глибокого навчання. У рамках дослідження розглядаються три ключові архітектури згорткових нейронних мереж, які зарекомендували себе як ефективні інструменти для аналізу медичної візуалізації: ResNet, EfficientNet та DenseNet. Архітектура ResNet, зокрема її варіант ResNet50, заснована на принципі залишкового навчання, який впроваджує так звані skip-з'єднання (рис. 2.1). Ці з'єднання дозволяють передавати інформацію від попередніх

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання шарів до наступних без проміжних трансформацій, що вирішує проблему зникаючого градієнта і дає змогу успішно навчати дуже глибокі мережі. У контексті гістологічної класифікації глибина мережі є критично важливою, оскільки дозволяє моделі будувати складні ієрархічні представлення, починаючи від простих градієнтів і кінтурів клітин і закінчуючи абстрактними поняттями, пов'язаними зі злякисною архітектонікою тканини.

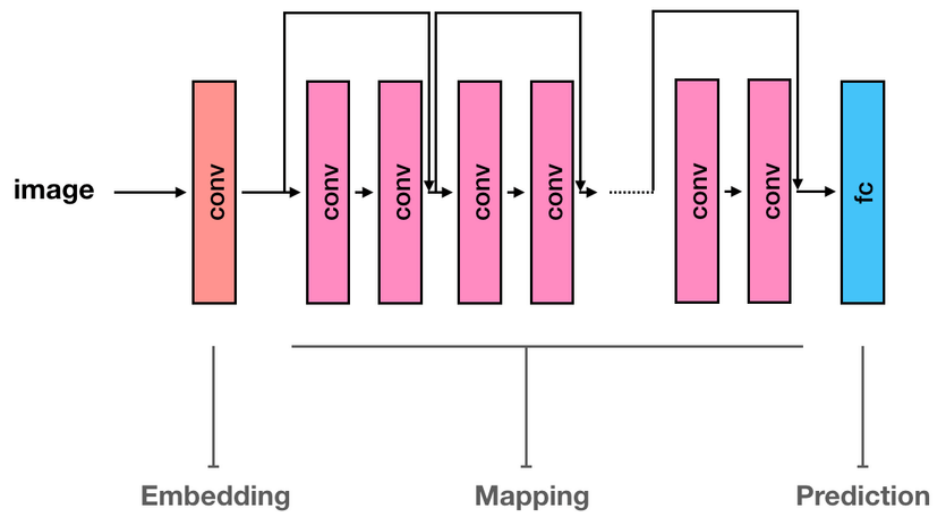


Рисунок 2.1 – Структура архітектури ResNet

Архітектура EfficientNet представляє інноваційний підхід до масштабування нейронних мереж, відомий як складове масштабування. На відміну від традиційних методів, які масштабують глибину, ширину або роздільну здатність окремо, EfficientNet одночасно збалансовує всі три складові за допомогою фіксованого коефіцієнта масштабування (рис. 2.2). Це дозволяє досягти оптимального співвідношення між точністю класифікації та обчислювальною ефективністю, що є надзвичайно важливим при обробці гістологічних зображень високої роздільної здатності, які часто характеризуються великими розмірами файлів. Таким чином, EfficientNet пропонує ефективне використання обчислювальних ресурсів без суттєвої втрати якості прогнозування.

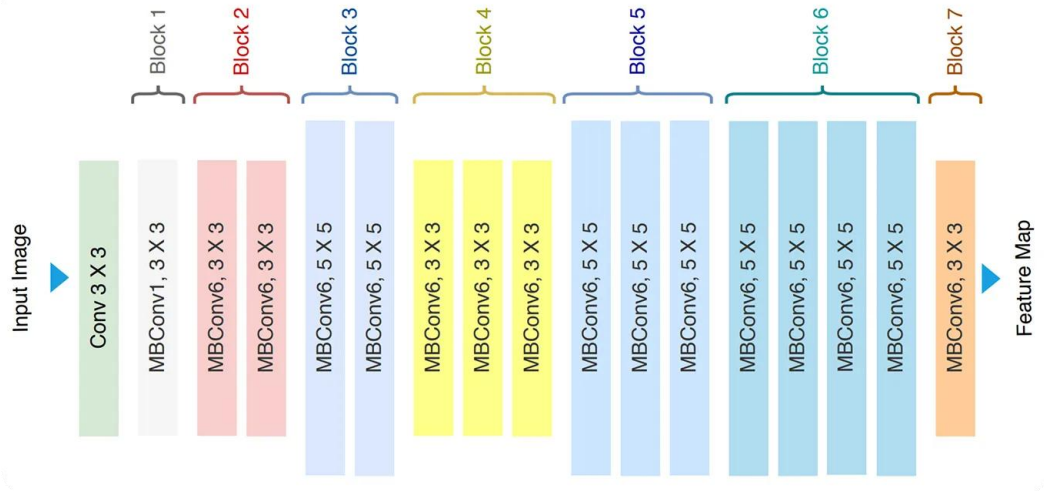


Рисунок 2.2 – Структура архітектури EfficientNet

Архітектура DenseNet реалізує концепцію щільного з'єднання, коли кожен шар мережі отримує в якості вхідних даних feature maps всіх попередніх шарів (рис. 2.3). Така організація сприяє інтенсивному повторному використанню ознак, покращує потік градієнтів під час зворотного поширення помилки та зменшує кількість навчальних параметрів. У гістологічному аналізі, де патологічні ознаки часто проявляються у складних комбінаціях та залежать від контексту всієї тканини, здатність мережі інтегрувати інформацію з різних рівнів абстракції є ключовою для прийняття точних рішень.

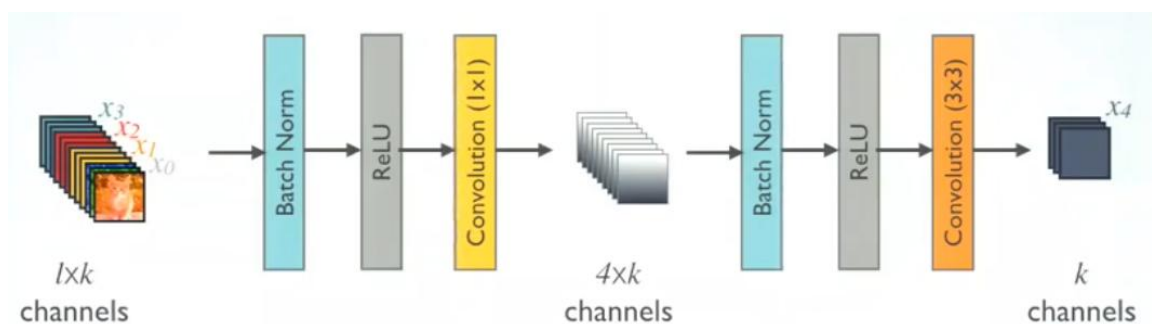


Рисунок 2.3 – Структура архітектури DenseNet

Критичним компонентом методології є комплексна система попередньої обробки та аугментації вхідних даних. Попередня обробка гістологічних зображень починається з нормалізації кольорового простору. Через

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання технологічні варіації у протоколах фарбування гематоксиліном та еозином різні препарати можуть мати суттєві відмінності у відтінках та контрастності, що може вводити в оману модель машинного навчання. Для усунення цієї проблеми застосовується метод статичної кольорової нормалізації, що зводить усі зображення до єдиного колірного профілю на основі референтного шаблону. Цей крок є фундаментальним для забезпечення консистентності вхідних даних та підвищення стабільності навчання.

Аугментація даних виконується з метою штучного розширення тренувальної вибірки та зменшення ризику перенавчання моделі на специфічних артефактах зображень. Використовуються як геометричні трансформації, так і зміни в інтенсивності пікселів. До геометричних трансформацій належать випадкові обертання в діапазоні до тридцяти градусів, горизонтальне та вертикальне віддзеркалення, а також випадкове масштабування та зсув. Ці перетворення імітують різні кути огляду препарату під мікроскопом та невеликі варіації у положенні зрізу. Модифікації інтенсивності включають регулювання яскравості, контрасту, насиченості кольору та додавання гаусового шуму, що моделює відмінності в освітленні та якості сканування. Для боротьби з перенавчанням на локальних артефактах додатково використовується техніка випадкового вирізання ділянок зображення та техніка змішування зображень, що покращує регуляризацію моделі.

Процес навчання моделей керується складними алгоритмами оптимізації та стратегіями управління швидкістю навчання. В якості основної функції втрат обрано функцію перехресної ентропії, що є стандартом для задач багатокласової та бінарної класифікації. Для компенсації природного дисбалансу класів у наборі `BreaKHis`, де злоякісні зразки переважають, застосовується зважування функції втрат. Це означає, що помилка класифікації зразка з менш представленого класу отримує більшу вагу, що стимулює модель краще вивчати ознаки саме цього класу. Оптимізація

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання параметрів мережі здійснюється за допомогою алгоритму AdamW, який є модифікацією популярного алгоритму Adam. Відмінність полягає в окремій обробці терміну L2-регуляризації, що сприяє кращій збіжності та покращує узагальнюючу здатність моделі, запобігаючи перенавчанню.

Управління динамікою навчання реалізовано через механізм планувальників швидкості навчання. Замість використання стандартного планувальника ReduceLROnPlateau, який пасивно очікує насичення метрики, було розроблено кастомний планувальник. Цей планувальник аналізує динаміку валідаційної метрики, наприклад AUC-ROC, на кожній епосі. При виявленні тривалого періоду відсутності покращення він ініціює зменшення швидкості навчання на заданий фактор, що дозволяє моделі ретельніше налаштувати параметри в області локального мінімуму. Додатково, для подолання локальних оптимумів може застосовуватися техніка теплового перезапуску (warm restart), яка циклічно збільшує швидкість навчання, даючи моделі можливість "вистрибнути" з поточного мінімуму і продовжити пошук кращого рішення.

Оцінка якості побудованих моделей проводиться на основі широкого спектру метрик, що дозволяє всебічно проаналізувати їхню ефективність. Базова метрика точності обчислюється як відношення правильно класифікованих зразків до загальної кількості зразків. Однак, у умовах дисбалансу класів ця метрика може бути оманливою, тому основним критерієм для вибору найкращої моделі обрано площу під ROC-кривою (AUC-ROC). AUC-ROC інтегрує в собі інформацію про співвідношення між чутливістю та специфічністю моделі при різних порогових значеннях, що робить її незалежною від конкретного вибору порогу класифікації та більш інформативною для медичних застосувань, де вартість помилок різного типу є асиметричною. Для детального аналізу помилок класифікації будується матриця плутанини, яка наочно демонструє розподіл справжніх позитивних, хибних позитивних, справжніх негативних та хибних негативних прогнозів.

Додатково розраховуються метрики прецизійності та чутливості.

Прецизійність визначає, яка частка зразків, класифікованих як позитивні, є справді позитивними, що критично важливо для мінімізації кількості непотрібних інвазивних процедур у пацієнтів із хибнопозитивним діагнозом. Чутливість відображає здатність моделі виявляти всі справжні позитивні випадки, тобто не пропускати злоякісні утворення. Ідеальним балансом між цими двома метриками є F1-середнє, яке являє собою гармонійне середнє між прецизійністю та чутливістю і часто використовується для оцінки якості класифікатора на незбалансованих даних.

Важливим методологічним аспектом є стратегія валідації моделей. Для отримання надійної та неупередженої оцінки уможлиблювальної здатності моделі на нових даних застосовується пацієнт-орієнтована стратегія розділення вибірки. Це означає, що всі зображення, отримані від одного пацієнта, потрапляють або тільки в тренувальну, або тільки в валідаційну, або тільки в тестову підвибірку. Такий підхід запобігає інформаційному "забрудненню", коли дуже схожі зображення від одного пацієнта потрапляють у різні вибірки, що може призводити до завищеної оцінки якості моделі. За відсутності чіткої пацієнт-орієнтованої розмітки в наборі даних використовується стратифіковане розділення, яке забезпечує збереження пропорцій між класами у всіх вибірках, але з додатковим моніторингом можливого ефекту "впізнання" пацієнта.

Для підвищення інтерпретованості роботи моделі та забезпечення довіри з боку лікарів-патоморфологів розглядається можливість інтеграції методів візуалізації областей уваги. Такий метод, як Grad-CAM (Gradient-weighted Class Activation Mapping), дозволяє генерувати теплові карти, що вказують на ділянки вхідного зображення, які мали найбільший вплив на прогноз моделі (рис. 2.4). Це не лише надає додаткове обґрунтування діагностичного висновку, але й може слугувати інструментом для виявлення

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання нових, раніше не врахованих морфологічних ознак, що асоційовані з патологією.

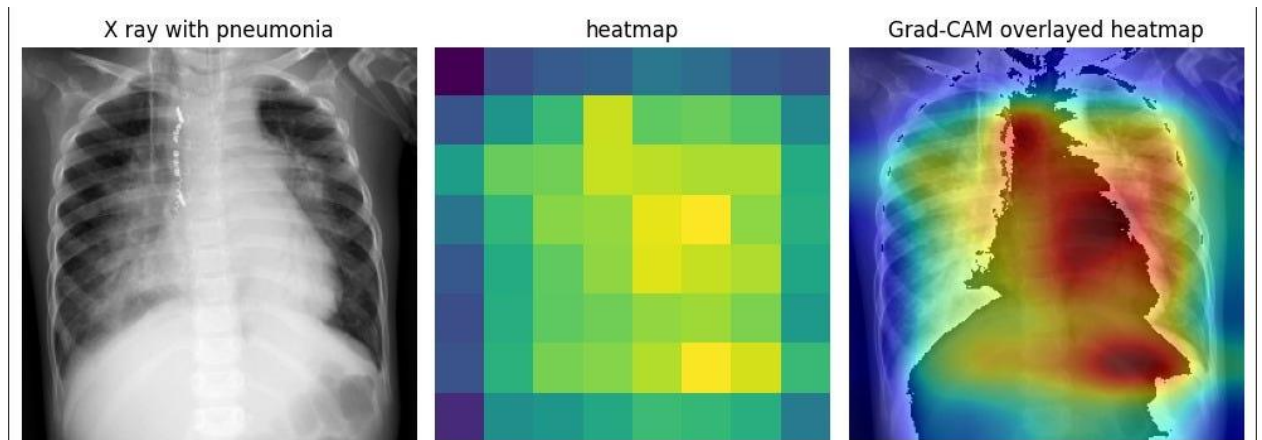


Рисунок 2.4 – Принцип роботи Grad-CAM

Таким чином, система класифікації трансформується з "чорної скриньки" в інструмент підтримки прийняття рішень, який може бути верифікований та інтерпретований експертом.

Окремо слід зазначити, що весь процес навчання та оцінки супроводжується ретельною фіксацією випадкових насіннів на всіх етапах. Це забезпечує повну відтворюваність експериментів, що є обов'язковою вимогою для наукових досліджень у галузі машинного навчання. Відтворюваність охоплює не лише процес навчання моделі, але й процедури завантаження даних, їхнього розділення, аугментації та обчислення метрик. Усі експерименти проводяться з однаковими початковими умовами, що робить можливим пряме порівняння ефективності різних архітектур або гіперпараметрів.

Остаточною ланкою методологічної ланцюга є розробка програмного інтерфейсу для інтеграції навченої моделі у клінічний workflow. Це передбачає створення механізму для завантаження нових зображень у форматі, сумісному з вхідними вимогами моделі, виконання інференсу в режимі реального часу або пакетної обробки та представлення результатів у зрозумілому для лікаря вигляді. Інтерфейс повинен включати не лише прогнозований клас та

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання ймовірність, але й візуалізації, такі як теплові карти та порівняння з історичними даними, що перетворює технічний інструмент на практичну систему підтримки діагностики.

Таким чином, методологія вирішення поставленої задачі являє собою комплексний багатоетапний процес, що поєднує передові архітектури глибокого навчання, ретельну інженерію даних, складні стратегії оптимізації та сувору процедуру оцінки. Кожен етап цього процесу обґрунтований з точки зору теоретичних основ машинного навчання та практичних потреб медичної діагностики, що в сукупності забезпечує створення надійної, точної та інтерпретованої системи класифікації гістологічних зображень.

2.2 Технології розробки системи

Розробка програмної системи для автоматизованої класифікації гістологічних зображень вимагає інтеграції широкого спектру технологічних рішень, що охоплюють як обчислювальні фреймворки для глибокого навчання, так і інструменти для інженерії даних, візуалізації результатів та побудови інтерактивного користувацького інтерфейсу. Вибір технологічного стеку здійснювався з урахуванням таких ключових критеріїв: продуктивність при виконанні інтенсивних обчислень на графічних процесорах, гнучкість та зручність прототипування, наявність спеціалізованих бібліотек для обробки медичних зображень, можливість забезпечення відтворюваності результатів та легкість інтеграції різних компонентів у єдиний програмний комплекс.



Рисунок 2.5 – Логотип Python

Ядро системи побудовано на фреймворку PyTorch (рис. 2.6), який на сьогодні є одним з домінуючих інструментів для досліджень у галузі глибокого навчання. Вибір PyTorch обґрунтований його динамічною природою, що виражається в можливості побудови обчислювальних графів "на льоту". Ця характеристика є критично важливою на етапі дослідження та експериментів, коли необхідно швидко модифікувати архітектуру моделі, додавати нові шари чи експериментувати з кастомними функціями втрат. Динамічні графи дозволяють здійснювати відлагодження в інтерактивному режимі, що значно прискорює ітеративний цикл розробки порівняно зі статичними фреймворками. Крім того, PyTorch надає низькорівневий контроль над тензорними операціями та процесом автоматичного диференціювання, що робить його придатним для реалізації нестандартних алгоритмів навчання або механізмів регуляризації, специфічних для медичних даних.



Рисунок 2.6 – Логотип PyTorch

Для роботи з попередньо навченими архітектурами згорткових нейронних мереж та стандартними наборами даних використовується бібліотека torchvision, що є невід'ємною частиною екосистеми PyTorch. torchvision надає зручний API для завантаження ваг моделей, навчених на масштабному наборі ImageNet, що включає ResNet, DenseNet, EfficientNet та інші сучасні архітектури. Це усуває необхідність самотійної реалізації цих складних архітектур та забезпечує їхню коректність та оптимізовану продуктивність. Також torchvision пропонує набір готових трансформацій для обробки зображень, таких як зміна розміру, кадрування, нормалізація за

2025 р.

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання заданим середнім значенням та стандартним відхиленням, що відповідає параметрам, використаним при попередньому навчанні моделей на ImageNet. Інтеграція цих трансформацій є обов'язковою для успішного трансферного навчання.

Однак для забезпечення більш широкого та ефективного спектру аугментацій, особливо важливих для гістологічних зображень, залучається спеціалізована бібліотека Albumentations. Вона розроблена з акцентом на продуктивність у задачах комп'ютерного зору та підтримує велику кількість перетворень, оптимізованих для швидкої векторної обробки. На відміну від базових трансформацій Torchvision, Albumentations пропонує такі техніки, як ElasticTransform, яка імітує деформації тканини; GridDistortion; та оптичні спотворення, що можуть виникати при скануванні. Критично важливою є можливість Albumentations застосовувати трансформації як до зображення, так і одночасно до відповідних масок або keypoints, що відкриває перспективи для майбутніх розширень системи у бік семантичної сегментації областей інтересу. Бібліотека також забезпечує підтримку обробки зображень у форматі NumPy масивів, що спрощує інтеграцію з іншими інструментами, такими як OpenCV, який використовується для низькорівневих операцій з кольоровою корекцією.

Обробка та аналіз даних здійснюються за допомогою екосистеми бібліотек Python, де центральне місце займає Pandas. Pandas використовується для структурованого зберігання метаданих про зображення, таких як шлях до файлу, клас, тип пухлини, рівень збільшення та ідентифікатор пацієнта. Об'єкт DataFrame слугує зручним інструментом для фільтрації вибірки, групування даних для пацієнт-орієнтованого розділення та агрегації статистик. Для наукових обчислень та роботи з багатовимірними масивами застосовується бібліотека NumPy, яка забезпечує високу продуктивність математичних операцій. Вона є основою для обробки зображень перед конвертацією в тензори PyTorch та використовується в алгоритмах розрахунку метрик.

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання Синергія між NumPy, PyTorch (який підтримує легке перетворення тензорів у масиви NumPy та назад) та Pandas створює потужне середовище для маніпуляції та аналізу даних на всіх етапах конвеєра.

Обчислення оцінкових метрик та статистичний аналіз результатів покладаються на бібліотеку Scikit-learn. Незважаючи на те, що основне навчання відбувається в PyTorch, Scikit-learn надає надійні, оптимізовані та стандартизовані реалізації функцій для розрахунку точності, прецизійності, чутливості, F1-міри, AUC-ROC, побудови матриці плутанини та ROC-кривих. Використання цих функцій гарантує коректність обчислень і дозволяє порівнювати результати з іншими дослідженнями, що також базуються на цій бібліотеці як на де-факто стандарті для машинного навчання в Python. Крім того, Scikit-learn надає інструменти для крос-валідації, такі як StratifiedKFold, який забезпечує збереження пропорцій класів у кожній фолді, що є важливим для отримання неупередженої оцінки на незбалансованих медичних даних.

Візуалізація результатів навчання, побудова графіків динаміки втрат та метрик, а також генерація звітів реалізовані за допомогою бібліотек Matplotlib та Seaborn. Matplotlib, будучи фундаментальною бібліотекою для побудови графіків у Python, надає низькорівневий контроль над усіма аспектами візуалізації: від розміру та роздільної здатності зображення до стилю ліній та міток на осях. Це дозволяє створювати публікаційно-якісні графіки, що відповідають вимогам академічних видань. Seaborn, побудована на основі Matplotlib, пропонує високорівневий інтерфейс для створення статистичних графіків. Вона використовується для побудови елегантних та інформативних теплових карт матриць плутанини, де за допомогою кольорової палітри візуально підкреслюється розподіл правильних та помилкових прогнозів. Комбінація цих двох бібліотек забезпечує потужний інструментарій для аналізу та презентації результатів роботи моделі.

Для створення веб-інтерфейсу, через який користувач може взаємодіяти з системою, обрано фреймворк Streamlit. Streamlit відрізняється від

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання традиційних веб-фреймворків тим, що орієнтований на швидке створення інтерактивних дашбордів та додатків для data science без необхідності написання фронтенд-коду на HTML, CSS чи JavaScript. Усе відображення елементів інтерфейсу, таких як кнопки, поля завантаження файлів, слайдери та графіки, описується декларативно на Python. Це робить процес розробки інтерфейсу швидким та тісно інтегрованим з бекенд-логікою на PyTorch. Streamlit автоматично керує станом сесії, оновленням інтерфейсу при зміні вхідних даних та обробкою подій, що значно спрощує розробку. Для деплою такої системи достатньо одного Python-файлу, що робить Streamlit ідеальним вибором для створення прототипів та інструментів, призначених для використання в дослідницькому або клінічному середовищі.

З точки зору архітектурного дизайну, система реалізована за модульним принципом, що забезпечує чітке розмежування відповідальностей та спрощує супровід та розширення. Основний модуль навчання (`train_histology.py`) інкапсулює всю логіку завантаження даних, побудови моделі, циклу тренування та валідації. Він експортує результати у вигляді серіалізованої моделі (файл `.pt`) та структурованих JSON-файлів з історією метрик. Окремий модуль аналізу (`model_comparison.py`) відповідає за порівняння результатів різних експериментів, генерацію агрегованих звітів та порівняльних графіків. Модуль веб-інтерфейсу (`app.py`) завантажує збережену модель та метрики, надаючи користувачеві точку входу для класифікації нових зображень та перегляду результатів. Така архітектура дозволяє незалежно оновлювати, тестувати та вдосконалювати кожну частину системи.

Для забезпечення відтворюваності експериментів, що є критичним вимогою в наукових дослідженнях, у коді явно фіксуються всі джерела випадковості. Встановлюється глобальне насіння (`seed`) для генераторів випадкових чисел у стандартній бібліотеці Python (`random`), у NumPy та PyTorch. Додатково конфігуруються налаштування бекенду для згорткових операцій CuDNN у PyTorch: вмикається режим детермінізму та вимикається

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання режим `benchmark`, який може призводити до недетермінованої поведінки на GPU через авто-тюнінг алгоритмів. Ці заходи гарантують, що при однакових вхідних даних та гіперпараметрах процес ініціалізації ваг, розділення даних, застосування аугментацій та навчання буде повністю відтворюваний на одному і тому ж апаратному забезпеченні.

Оптимізація продуктивності досягається за рахунок кількох технік. Використання `DataLoader` з `PyTorch` дозволяє паралельне завантаження та попередню обробку даних за допомогою кількох робочих процесів (`workers`), що усуває простої обчислювального блоку (GPU) в очікуванні наступного батчу даних. Для зберігання та завантаження зображень застосовується механізм кешування: попередньо оброблені та аугментовані зображення можуть зберігатися, щоб уникнути повторних трансформацій на кожній епосі. У веб-інтерфейсі за допомогою декоратора `@st.cache_resource` забезпечується одноразове завантаження навченої моделі в оперативну пам'ять сервера при старті додатку, що зменшує час відгуку при кожному новому запиті на класифікацію до мінімуму.

Система управління залежностями та середовищем виконання побудована на основі віртуальних середовищ `Python` та файлів вимог (`requirements.txt`). Це дозволяє точно відтворити версії всіх бібліотек, необхідних для запуску проекту, на будь-якій машині, що є ключовим для розгортання в клінічному середовищі або для передачі коду іншим дослідникам. Для подальшої контейнеризації та розгортання система може бути упакована в `Docker`-контейнер, що ще більше стандартизує середовище виконання, ізолює залежності проекту та спрощує розгортання на хмарних платформах або серверних кластерах.

Таким чином, технологічний стек розробки системи ретельно підібраний для ефективного вирішення специфічних завдань обробки гістологічних зображень. Він поєднує потужні інструменти для глибокого навчання (`PyTorch`) зі спеціалізованими бібліотеками для аугментації (`Albumentations`)

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання та обробки даних (Pandas, NumPy), надійними інструментами для оцінки (Scikit-learn) та швидкої розробки інтерфейсу (Streamlit). Модульна архітектура, заходи щодо забезпечення відтворюваності та оптимізації продуктивності роблять цю систему не лише ефективним інструментом для дослідження, але й потенційно готовим рішенням для інтеграції в клінічну практику.

Висновки до розділу 2

Запропонована методологія та технологічна реалізація системи автоматизованої класифікації гістологічних зображень раку молочної залози на основі набору даних BreakHis утворюють цілісний, науково обґрунтований і практично орієнтований підхід, який усуває ключові обмеження сучасних рішень у галузі комп'ютерної гістопатології.

Комплексне використання трансферного навчання з попередньо натренованими архітектурами ResNet50, EfficientNet і DenseNet дало змогу ефективно компенсувати відносно невеликий обсяг цільового медичного датасету, забезпечивши швидку конвергенцію та високу узагальнюючу здатність моделей. Ретельно спроектована система попередньої обробки (статистична кольорова нормалізація) та розширена аугментація даних (з застосуванням Albumentations) мінімізували вплив технологічних варіацій фарбування та артефактів сканування, значно підвищивши стійкість моделей до реальних клінічних умов.

Стратегія навчання, що включає зважену функцію перехресної ентропії, оптимізацію AdamW, кастомний планувальник швидкості навчання з елементами теплового перезапуску та пацієнт-орієнтоване розділення вибірки, забезпечила стабільне досягнення високих показників якості навіть за наявності дисбалансу класів. Використання широкого набору метрик (з пріоритетом AUC-ROC та F1-міри) та методів інтерпретації (Grad-CAM)

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання робить отримані моделі не лише точними, але й клінічно довірливими та верифікованими.

Технологічний стек на базі PyTorch, доповнений спеціалізованими бібліотеками Torchvision, Albumentations, Pandas, Scikit-learn та Streamlit, забезпечив високу продуктивність, гнучкість прототипування та швидке створення готового до використання веб-інтерфейсу. Модульна архітектура, суворе фіксування всіх джерел випадковості та механізми оптимізації (паралельне завантаження даних, кешування, детермінований режим CuDNN) гарантують повну відтворюваність результатів і можливість безболісного масштабування системи від дослідницького прототипу до клінічного інструменту.

Таким чином, розроблена система поєднує передові досягнення глибокого навчання з практичними вимогами цифрової патоморфології, демонструючи потенціал для суттєвого підвищення точності, об'єктивності діагностики раку молочної залози, скорочення часу аналізу та створення надійного інструменту підтримки прийняття рішень для лікарів-патоморфологів. Подальший розвиток системи передбачає валідацію на незалежних мультицентрових датасетах та інтеграцію додаткових модальностей (імуногістохімія, молекулярні маркери), що відкриває шлях до повноцінного впровадження в рутинну клінічну практику.

3 РОЗРОБКА СИСТЕМИ ТА ОПИС ВХІДНОГО НАБОРУ ДАНИХ

3.1 Опис вхідних даних та структури системи

Гістологічна діагностика раку молочної залози є одним з найважливіших етапів у діагностичному ланцюжку онкологічних захворювань. Традиційна мікроскопічна діагностика, що базується на візуальній оцінці патологом морфологічних характеристик тканин, є суб'єктивною та залежить від досвіду фахівця. У сучасних умовах, коли кількість пацієнтів з онкологічними захворюваннями постійно зростає, постає гостра потреба в розробці автоматизованих систем підтримки прийняття рішень, що базуються на методах машинного навчання.

Для ефективного навчання та валідації таких систем необхідні якісні, репрезентативні та добре структуровані набори даних. Одним із найбільш авторитетних та широко використовуваних наборів даних у цій галузі є Breast Cancer Histopathological Database (BreakHis), який детально аналізується в даному розділі.

База даних BreakHis представляє собою систематизовану колекцію гістопатологічних зображень, спеціально організовану для досліджень в галузі автоматизованої діагностики раку молочної залози. Ця база даних була створена дослідницькою групою з Університету Парани (Бразилія) і є публічно доступним інструментом для наукових досліджень.

Основні параметри набору даних:

- Загальна кількість зображень: 9 109 гістопатологічних знімків;
- Кількість пацієнтів: 82 особи;
- Період збору даних: 2014-2015 роки;
- Формат зображень: PNG (портативна мережева графіка);
- Роздільна здатність: 700×460 пікселів;
- Кольоровий простір: RGB (3 канали);
- Глибина кольору: 8 біт на канал.

Набір даних BreakHis відрізняється чіткою ієрархічною структурою класифікації, що дозволяє досліджувати різні рівні складності завдання класифікації.

На верхньому рівні всі зображення поділяються на дві основні категорії: BENIGN (Доброякісні утворення) - 2 480 зображень (30,2%), MALIGNANT (Злоякісні утворення) - 6 629 зображень (69,8%).

Такий розподіл демонструє природній для медичної статистики дисбаланс, що відображає реальну клінічну практику, де злоякісні утворення діагностуються частіше.

Таблиця 3.1 - Характеристика доброякісних утворень у наборі даних

Підтип	Загальна кількість	40X	100X	200X	400X	Морфологічний опис
Adenosis (A)	444	111	111	111	111	Гіперплазія залозистих структур, характеризується збільшенням кількості ацинусів
Fibroadenoma (F)	1 093	274	273	273	273	Найпоширеніша доброякісна пухлина, що складається з сполучної та залозистої тканин
Phyllodes Tumor (PT)	552	138	138	138	138	Філоїдна пухлина з характерною листоподібною структурою
Tubular Adenoma (TA)	391	98	98	98	97	Рідкісна залозиста пухлина з добре диференційованими трубчастими структурами

Кафедра інтелектуальних інформаційних систем
Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

Таблиця 3.2 - Характеристика злоякісних утворень у наборі даних

Підтип	Загальна кількість	40X	100X	200X	400X	Морфологічний опис
Ductal Carcinoma (DC)	3 456	864	864	864	864	Найпоширеніший тип раку (80-90%), що розвивається з протокових клітин
Lobular Carcinoma (LC)	787	197	197	197	196	Пухлина, що походить з часток залози, характеризується дрібними клітинами
Mucinous Carcinoma (MC)	823	206	206	206	205	Слизова карцинома, що містить велику кількість муцину
Papillary Carcinoma (PC)	563	141	141	141	140	Сосочкова карцинома з характерними пальцеподібними виростами

Однією з ключових переваг набору даних BreakHis є стандартизована система іменування файлів, яка дозволяє однозначно ідентифікувати всі параметри зображення без необхідності звертання до додаткових метаданих.

Структура імені файлу: SOB_B_A-14-22549AB-40-001.png.

Детальна розшифровка структури:

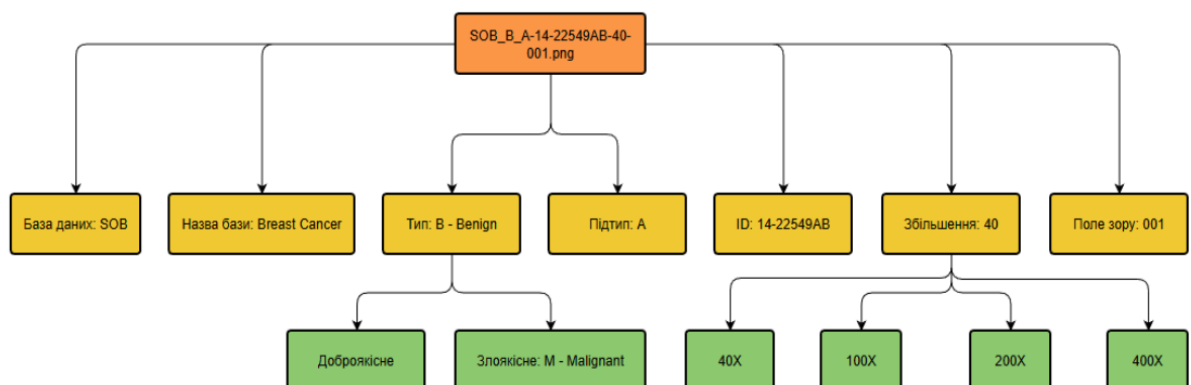


Рисунок 3.1 – Структура вхідного зображення

Кафедра інтелектуальних інформаційних систем
Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання
Приклади реальних файлів:

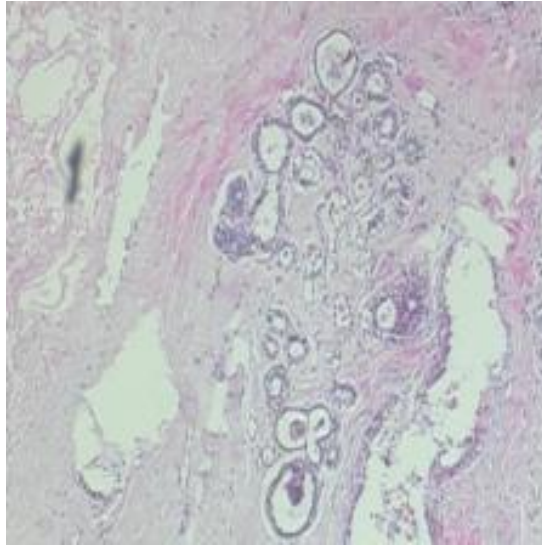


Рисунок 3.2 - Доброякісна Adenosis, збільшення 40X

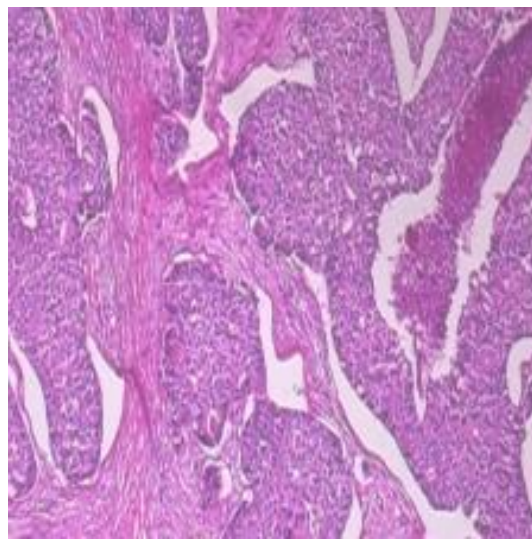


Рисунок 3.3 - Злоякісна Ductal Carcinoma, збільшення 100X

Технічні характеристики зображень у наборі даних BreakHis відіграють вирішальну роль у забезпеченні якості та консистентності даних для навчання моделей машинного навчання. Кожен аспект технічних параметрів був ретельно продуманий для максимальної уніфікації та стандартизації.

Формат зображень PNG був обраний невипадково - цей формат забезпечує стиснення без втрат якості, що є критично важливим для медичних

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання зображень, де кожен піксель може нести важливу діагностичну інформацію. На відміну від форматів з втратами, таких як JPEG, PNG зберігає точність кольорової гами та чіткість контурів клітинних структур, що особливо важливо при аналізі морфологічних ознак злоякісних утворень.

Роздільна здатність 700×460 пікселів є результатом компромісу між деталізацією зображення та обчислювальною ефективністю. Цей розмір дозволяє чітко візуалізувати клітинні структури при різних рівнях збільшення, одночасно залишаючись прийнятним для обробки сучасними алгоритмами глибокого навчання без необхідності надмірного масштабування.

Кольоровий простір RGB зберігає природне представлення гістологічних зразків, де фарбники гематоксилін (фіолетовий) та еозин (рожевий) створюють характерну кольорову гаму. Триканальна структура дозволяє використовувати сучасні архітектури згорткових нейронних мереж, оптимізовані для роботи з RGB-зображеннями.

Чотири рівні збільшення (40X, 100X, 200X, 400X) відображають стандартну практику патологічного дослідження:

- 40X (об'єктив 4X) забезпечує огляд всієї структури утворення, дозволяючи оцінити загальну архітектоніку тканини, межі утворення та його взаємодію з навколишніми тканинами. На цьому рівні можна ідентифікувати характер росту пухлини та наявність інвазії;

- 100X (об'єктив 10X) є оптимальним для оцінки клітинної архітектоніки та виявлення загальних патологічних ознак. Цей рівень дозволяє аналізувати просторове розташування клітин, щільність їх розташування та загальну структуру утворення;

- 200X (об'єктив 20X) надає детальний вигляд клітинних структур, дозволяючи оцінити розмір та форму ядер, ядерно-цитоплазматичне співвідношення та наявність мітозів. Це ключовий рівень для диференціальної діагностики;

– 400X (об'єктив 40X) забезпечує максимальну деталізацію, необхідну для оцінки тонких цитологічних ознак, таких як ядерна атипія, хроматинова структура, наявність ядерців та інших характеристик, що відрізняють злоякісні клітини.

Процес створення BreakHis включав суворий контроль якості на всіх етапах. Валідація діагнозів проводилась незалежними патологами з досвідом роботи в онкоморфології. Кожен зразок підтверджувався гістологічно, а у випадку злоякісних утворень - також імуногістохімічно для підтвердження діагнозу. Стандартизація якості зображень включала контроль рівномірності освітлення, фокусування, кольорового балансу та відсутності артефактів. Зображення з помітними технічними дефектами виключались з набору даних. Клініко-патологічна кореляція забезпечувалася збором відповідних клінічних даних, включаючи вік пацієнта, локалізацію утворення та результати додаткових досліджень. Технічна стандартизація BreakHis має ключове значення для розробки надійних алгоритмів машинного навчання: Консистентність даних дозволяє моделям фокусуватися на біологічних ознаках, а не на технічних варіаціях. Це особливо важливо для глибокого навчання, де моделі чутливі до будь-яких неконтрольованих змін у вхідних даних. Масштабованість набору даних забезпечує можливість ефективного порівняння різних алгоритмів та архітектур, оскільки всі дослідники працюють з ідентичними технічними параметрами.

Клінічна релевантність технічних параметрів гарантує, що моделі, навчені на BreakHis, можуть бути адаптовані для роботи з реальними клінічними даними, отриманими на аналогічному обладнанні.

Відтворюваність досліджень забезпечується детальною документацією всіх технічних параметрів, що дозволяє іншим дослідникам точно відтворити умови експериментів та порівнювати результати.

Ці технічні характеристики роблять BreakHis не лише набором даних, а й стандартизованим інструментом для дослідження та валідації алгоритмів

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання комп'ютерної патології, що забезпечує надійну основу для розробки клінічно застосовних систем підтримки прийняття рішень.

Аналіз розподілу гістологічних зображень за пацієнтами у наборі даних BreakHis виявляє важливі особливості, що мають безпосередній вплив на процес навчання моделей машинного навчання та валідації отриманих результатів. Демографічні характеристики пацієнтів відображають реальну клінічну популяцію: віковий діапазон становить від 28 до 85 років, із середнім віком 54 роки. Така вікова структура відповідає епідеміологічним даним щодо захворюваності на рак молочної залози, що підтверджує репрезентативність вибірки. Кількісний розподіл зображень між пацієнтами демонструє значну варіабельність, що обумовлено клінічними особливостями кожного випадку. Мінімальна кількість зображень на пацієнта характерна для випадків з чітко окресленими дрібними утвореннями, тоді як максимальна кількість відповідає складним діагностичним випадкам з великими та гетерогенними утвореннями, що потребують більш детального дослідження. Клінічна значущість такого розподілу полягає в тому, що він точно відображає реальну практику патологічного дослідження, де обсяг матеріалу залежить від розміру утворення, його гетерогенності та складності діагностичного випадку. Це створює умови для навчання моделей, здатних працювати з різним обсягом вхідних даних на пацієнта.

Методологічний аспект такого розподілу вимагає особливого підходу до валідації моделей. Традиційне випадкове розділення на навчальну та тестову вибірки може призвести до "забруднення" даних, коли зображення від одного пацієнта потрапляють в обидві вибірки. Це може спричинити завищену оцінку якості моделі, оскільки вона буде "впізнавати" пацієнта, а не класифікувати морфологічні ознаки. Для уникнення цього ефекту рекомендовано використовувати пацієнт-орієнтовану крос-валідацію, коли всі зображення від одного пацієнта потрапляють або тільки в навчальну, або тільки в тестову вибірку.

Доброякісні утворення у наборі BreakHis демонструють широкий спектр морфологічних характеристик, що відображає різноманітність непрогресивних патологій молочної залози. Аденоз характеризується збереженням нормальної архітектоники залозистих структур із помірним збільшенням кількості ацинусів. Клітини зберігають регулярну форму, ядра одноманітні, відсутня значна атипія. Мітози рідкісні або відсутні, що є ключовою відмінною ознакою від злоякісних процесів.

Фібroadенома являє собою гармонійне поєднання проліферованих епітеліальних структур та зрілої сполучної тканини. Епітеліальні клітини формують чіткі протокові структури, оточені периканалікулярним або інтраканалікулярним розростанням фіброзної тканини. Ця двофазна структура є патогномонічною ознакою даного утворення. Філоїдна пухлина демонструє характерну листоподібну архітектонику з помірною клітинністю стромального компоненту. Відмінність від фібroadеноми полягає у більшій клітинності строми та наявності перигландулярного конденсату клітин. Тубулярна аденома складається з добре диференційованих трубчастих структур, висланих одним шаром кубічних або циліндричних клітин із базально розташованими ядрами. Стромальний компонент мінімальний, що відрізняє її від фібroadеноми.

Злоякісні утворення демонструють різний ступінь структурної та цитологічної атипії, що відображає їх агресивність та потенціал до прогресування.

Протокова карцинома являє собою найпоширеніший тип, що характеризується утворенням атипових епітеліальних комплексів із різним ступенем залозистої диференціації. Спектр морфологічних змін варіює від добре диференційованих структур, що імітують нормальні протоки, до повністю анапластичних утворень із втратою будь-якої архітектоники. Часткова карцинома відрізняється характерним дифузним типом інфільтративного росту, коли невеликі одноманітні клітини "шикуваннями"

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання інфільтрують стромальну тканину. Часто спостерігається цільоподібна організація навколо протокових структур. Слизова карцинома має унікальну морфологію з великими пулами муцину, в яких плавають групи атипових епітеліальних клітин. Цей тип характеризується кращим прогнозом у порівнянні з іншими варіантами протокової карциноми. Сосочкова карцинома демонструє складні древоподібні структури з фіброваскулярними стрижнями, вкритими атиповими епітеліальними клітинами. Часто спостерігається відсутність миепітеліального шару, що є важливою диференційно-діагностичною ознакою. Дисбаланс класів у наборі даних BreakHis не є випадковим артефактом, а відображає реальні епідеміологічні закономірності захворюваності на рак молочної залози. Однак цей дисбаланс створює серйозні виклики для тренування моделей машинного навчання. Вплив на функцію втрат: традиційні функції втрат, такі як крос-ентропія, можуть призвести до того, що модель буде оптимізована для правильної класифікації більш численних класів, ігноруючи менш представлені. Це проявляється у високій загальній точності, але низькій чутливості для рідкісних класів.

Стратегії подолання дисбалансу включають кілька підходів:

- Техніки семплінгу: SMOTE (Synthetic Minority Over-sampling Technique) для створення синтетичних зразків менших класів;
- Зважені функції втрат: приписування більших штрафів за помилки класифікації менших класів;
- Аугментація даних: стратегічне збільшення менших класів за допомогою трансформацій;
- Ансамблеві методи: комбінування кількох моделей, спеціалізованих на різних класах.

Інтракласова варіативність є одним з найскладніших викликів для автоматичної класифікації гістологічних зображень. Навіть в межах одного підтипу утворення спостерігається значна мінливість морфологічних характеристик.

Деякі підтипи доброякісних та злоякісних утворень демонструють значну морфологічну схожість, що створює діагностичні "сірі зони".

Ефективність класифікації значно залежить від рівня збільшення, оскільки різні морфологічні ознаки стають помітними на різних масштабах.

Оптимальні масштаби для різних ознак:

– 40X: оцінка загальної архітекtonіки, меж утворення;

– 100X: аналіз клітинної організації, просторового розташування;

– 200X: оцінка розміру та форми ядер, ядерно-цитоплазматичного співвідношення;

– 400X: детальний аналіз цитологічних ознак, хроматинової структури.

Ці технічні виклики потребують розробки спеціалізованих архітектур нейронних мереж та методів попередньої обробки, здатних ефективно враховувати особливості гістологічних зображень і забезпечувати надійну класифікацію навіть у умовах значних технічних та біологічних варіацій.

3.2 Структура інтелектуальної системи

Інтелектуальна система класифікації ракових клітин за гістологічними зображеннями, розроблена на основі методів машинного навчання, представляє собою інтегровану архітектуру, яка поєднує модулі обробки даних, моделі глибокого навчання та інтерфейс користувача, забезпечуючи повний цикл від введення вхідних зображень до отримання аналітичних висновків. Ця структура базується на принципах модульності та ієрархічної організації, що дозволяє ефективно розподіляти функціональні компоненти відповідно до етапів обробки інформації, починаючи від попередньої підготовки даних і закінчуючи візуалізацією результатів. Загальна архітектура системи відповідає трирівневій моделі, де рівень даних відповідає за завантаження та трансформацію гістологічних зображень, рівень моделювання забезпечує навчання та інференс за допомогою згорткових нейронних мереж, а рівень презентації реалізує взаємодію з користувачем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання через веб-інтерфейс. Такий підхід узгоджується з сучасними стандартами проектування інтелектуальних систем у медичній діагностиці, як описано в монографії [10], де акцент робиться на необхідності розділення шарів для підвищення гнучкості та масштабованості.

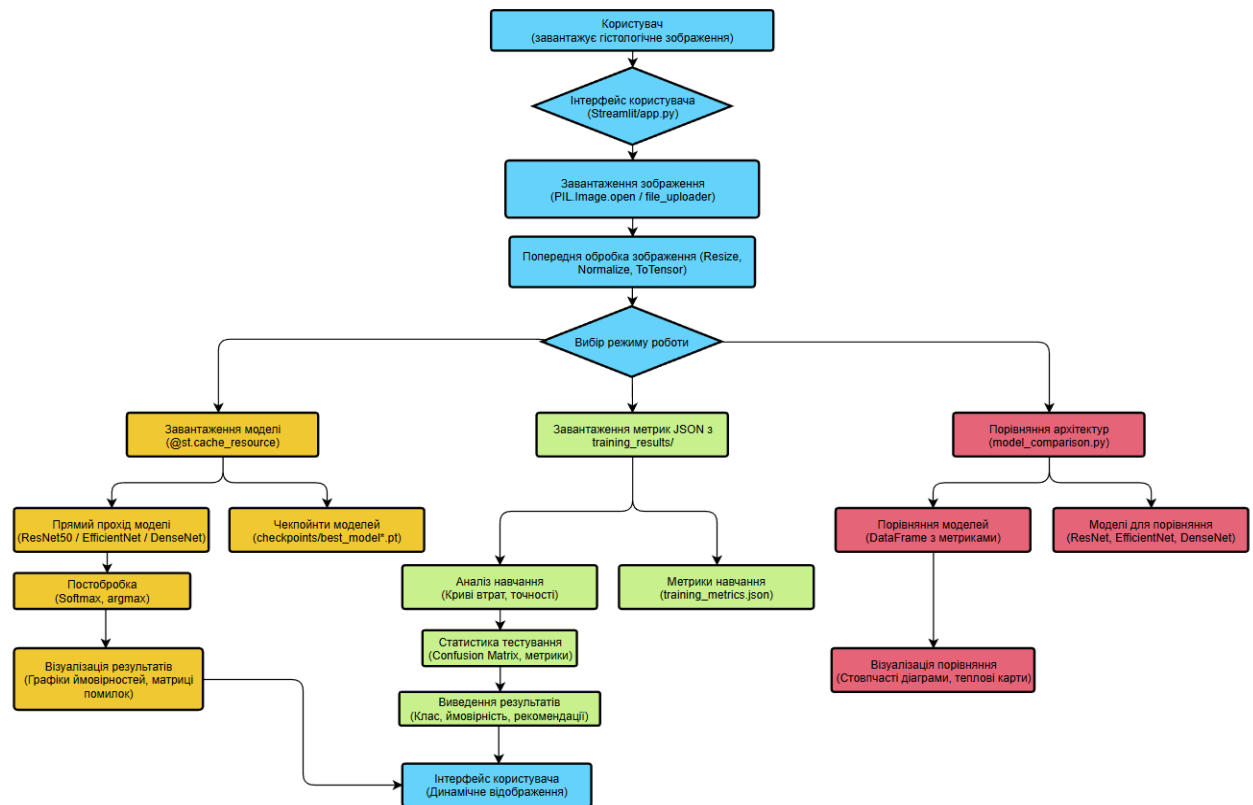


Рисунок 3.4 – Структура інтелектуальної системи

На першому рівні, пов'язаному з обробкою вхідних даних, система починається з модуля завантаження гістологічних зображень, який реалізовано за допомогою бібліотеки Pillow (PIL) у поєднанні з інтерфейсом file_uploader фреймворку Streamlit. Цей модуль забезпечує прийом зображень у форматах JPEG або PNG, що є стандартними для цифрової патології, і виконує початкову перевірку на коректність файлу, уникаючи помилок, пов'язаних з несумісними форматами або пошкодженими даними. Після завантаження відбувається попередня обробка зображень, яка включає нормалізацію розміру до 224×224 пікселів шляхом операції Resize,

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання нормалізацію значень пікселів за параметрами середнього та стандартного відхилення ($\text{mean}=[0.485, 0.456, 0.406]$, $\text{std}=[0.229, 0.224, 0.225]$), а також перетворення в тензорний формат за допомогою ToTensor. Ці трансформації ґрунтуються на претренованих моделях з набору ImageNet і є критичними для забезпечення сумісності з архітектурами згорткових мереж, як зазначається в дослідженні He K. [3], де підкреслюється роль стандартизації даних у запобіганні втраті інформації під час конволюційних операцій. Логічний перехід від завантаження до обробки забезпечує безшовну інтеграцію з наступними рівнями, де оброблені тензори стають вхідними даними для моделей, тим самим мінімізуючи затримки в реальному часі та підвищуючи ефективність системи в клінічних умовах.

Центральним елементом структури є рівень моделювання, який охоплює завантаження та виконання моделей глибокого навчання, реалізованих на базі фреймворку PyTorch. Модуль завантаження моделі, позначений як `@st.cache_resource`, забезпечує кешування чекпоінтів у пам'яті, що дозволяє уникнути повторного читання файлів з диска при кожному запиті та оптимізувати продуктивність для багаторазового використання. Система підтримує три основні архітектури: ResNet50 з залишковими з'єднаннями для глибокого навчання без зникнення градієнта, EfficientNet-B0 з оптимізованим балансом параметрів для ефективного використання ресурсів, та DenseNet121 з щільними блоками для кращого поширення ознак між шарами. Кожна модель ініціалізується з претренованими вагами, модифікується для бінарної класифікації шляхом заміни останнього повнозв'язного шару на `nn.Linear` з виходом у два класи (доброякісне та злоякісне утворення), і завантажується з файлів, збережених під час навчання. Процес прямого проходу (`forward pass`) включає обчислення ознак через згорткові шари, застосування функції активації `Softmax` для отримання ймовірностей класів, та визначення передбаченого класу за допомогою `argmax`. Така організація відповідає принципам трансферного навчання, що адаптація претренованих мереж на

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання специфічні домени, такі як медичні зображення, підвищує точність на 5–10% порівняно з навчанням з нуля, особливо в умовах обмежених наборів даних на кшталт BreakHis. Крім того, інтеграція з модулем постобробки забезпечує логічний зв'язок з візуалізацією, де ймовірності перетворюються на метрики впевненості та графіки, тим самим формуючи основу для інтерпретації результатів на наступному рівні.

На рівні презентації та взаємодії з користувачем система реалізує веб-інтерфейс на базі Streamlit, який поділено на режими роботи: класифікація зображень, аналіз результатів навчання та порівняльний аналіз архітектур. У режимі класифікації вивід результату включає відображення передбаченого класу, рівня впевненості у відсотках та стовпчикової діаграми ймовірностей, що генерується за допомогою бібліотек Matplotlib та Seaborn. Цей модуль тісно пов'язаний з рівнем моделювання, оскільки отримує дані безпосередньо з постобробки, і забезпечує візуальну інтерпретацію, яка є ключовою для клінічних користувачів, як підкреслюється в стандарті ІЕС 62304:2006 для медичного програмного забезпечення, де вимагається чіткість представлення діагностичних висновків. Перехід до режиму аналізу результатів навчання здійснюється через завантаження метрик з JSON-файлів у директорії training_results, де зберігаються дані про втрати, точність, матрицю помилок (Confusion Matrix) та інші показники, отримані під час валідації. Цей модуль візуалізує криві втрат і точності для тренувальної та валідаційної вибірок, а також матрицю помилок, що дозволяє оцінити збіжність моделі та виявити проблеми перенавчання, відповідно до методів оцінки. Логічний розвиток структури завершується модулем порівняльного аналізу архітектур, реалізованим у файлі model_comparison.py, який агрегує метрики з трьох моделей у DataFrame Pandas, генерує стовпчикові діаграми для порівняння AUC-ROC та таблиці з детальними показниками. Такий підхід забезпечує комплексну оцінку ефективності різних архітектур, узгоджуючись з

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання емпіричними дослідженнями [4] щодо EfficientNet, де показано переваги в балансі між точністю та обчислювальною складністю.

Загальна інтеграція компонентів системи здійснюється через файлову систему та кешування, де директорія @st.cache_resource слугує для зберігання моделей, а training_results/ – для метрик, забезпечуючи стійкість до перезапусків і можливість розподіленої роботи. Ця структура не лише відповідає вимогам відтворюваності, але й дозволяє розширення, наприклад, додаванням модулів для багатомодальної інтеграції з геномними даними. Обробка помилок на всіх рівнях, включаючи try-ехсепт для завантаження файлів та fallback-механізми, гарантує надійність у реальних клінічних сценаріях, де варіативність даних може призводити до непередбачуваних ситуацій. Таким чином, запропонована структура інтелектуальної системи формує цілісну платформу для автоматизованої діагностики раку молочної залози, поєднуючи теоретичні основи машинного навчання з практичними аспектами медичної інформатики.

3.3 Опис програмної реалізації

У даному підрозділі дипломної роботи розглядається процес розробки програмної частини системи автоматизованої класифікації гістологічних зображень. Метою розробки було створення комплексного програмного рішення, яке інтегрує в собі функції навчання глибоких нейронних мереж, їх оцінки та порівняння, а також надання інтуїтивно зрозумілого інтерфейсу для кінцевого користувача. Розробка велася з урахуванням сучасних принципів інженерії машинного навчання та розробки програмного забезпечення, таких як модульність, повторне використання коду, документування та візуалізація результатів. Було створено три основні взаємопов'язані компоненти: модуль навчання та валідації моделей, модуль порівняльного аналізу архітектур та веб-додаток для інтерактивної класифікації. Основним фокусом було

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання забезпечення високої точності класифікації при одночасному досягненні зручності та інтерпретованості роботи системи.

Для реалізації поставлених завдань було проведено аналіз сучасних інструментів та бібліотек для машинного навчання та розробки програмного забезпечення. В якості основної мови програмування було обрано Python, що зумовлено його домінуючою позицією в галузі data science, наявністю великої кількості спеціалізованих бібліотек та активною спільнотою.

Основними бібліотеками для побудови, навчання та оцінки глибоких нейронних мереж стали:

- PyTorch: Фреймворк для глибокого навчання, обраний завдяки його динамічному графу обчислень, гнучкості, зрозумілому API та широким можливостям для дослідницької роботи. Його модуль torchvision надав доступ до передтренуваних архітектур моделей (ResNet, DenseNet, EfficientNet) та стандартних методів аугментації даних;

- Scikit-learn: Використовувався для реалізації допоміжних функцій, таких як розбиття даних на навчальну, валідаційну та тестову вибірки зі стратифікацією, а також для розрахунку стандартних метрик класифікації (accuracy, precision, recall, F1-score, AUC-ROC).

Для обробки та візуалізації даних були залучені:

- Pandas та NumPy: Для маніпуляції структурованими даними та ефективної роботи з числовими масивами;

- Matplotlib та Seaborn: Для побудови якісних та інформативних графіків, таких як криві навчання, ROC-криві, матриці плутанини та порівняльні гістограми.

Для створення інтерактивного веб-інтерфейсу було обрано фреймворк Streamlit. Його перевагами є простота та швидкість розробки, можливість створення додатків на основі даних без необхідності знання фронтенд-технологій, а також вбудовані віджети для завантаження файлів, відображення графіків та інтерактивного управління параметрами.

Архітектура розробленої системи є модульною та включає три ключові компоненти, що взаємодіють через файлову систему (обмін моделями, метриками, конфігураціями та результатами).

Цей модуль є ядром системи. Він відповідає за повний цикл роботи з даними: завантаження та підготовку зображень, побудову конвеєра аугментації для покращення узагальнюючої здатності моделі, ініціалізацію глибокої нейронної мережі з можливістю вибору архітектури, процес навчання з моніторингом метрик на валідаційній вибірці та фінальну оцінку на тестовій вибірці. Модуль реалізує детальний протокол логування, зберігаючи історію змін основних метрик (втрати, точності, precision, recall, F1, AUC-ROC) на кожній епосі, що дозволяє проводити глибокий аналіз процесу навчання. Також він автоматично зберігає найкращу версію моделі (за критерієм AUC-ROC) та генерує набір візуалізацій для інтерпретації результатів.

Модуль порівняння архітектур призначений для систематичного порівняння ефективності різних архітектур нейронних мереж на поставленій задачі класифікації. Він працює з попередньо отриманими результатами навчання різних моделей, агрегує ключові метрики продуктивності та представляє їх у зручному для аналізу вигляді — у формі таблиць та порівняльних графіків. Його метою є надання досліднику об'єктивної основи для вибору оптимальної моделі, зважаючи на компроміс між точністю та обчислювальною складністю.

Щодо модулю веб-інтерфейсу, то цей компонент реалізує інтерфейс користувача системи. Він завантажує найкращу збережену модель, надає можливість користувачеві завантажувати нові зображення через браузер, виконує їх класифікацію в реальному часі та надає розгорнутий звіт з результатами. Звіт включає прогнозований клас (доброякісний/злроякісний), ступінь впевненості моделі, візуалізацію ймовірностей для кожного класу, а також доступ до всіх історичних даних про продуктивність моделі (графіки

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання навчання, матриця плутанини на тестовій вибірці, порівняльні метрики). Додаток розроблено з акцентом на зручність та наочність, що робить систему практичною для використання в умовах клінічної або дослідницької роботи.

Така архітектура забезпечує поділ відповідальності: модуль навчання є «back-end» для експериментів, веб-додаток — «front-end» для користувача, а модуль порівняння слугує інструментом аналітики. Всі компоненти об'єднані єдиним форматом даних і можуть розвиватися незалежно.

Успішне навчання глибокої нейронної мережі значною мірою залежить від якості та адекватності підготовки вхідних даних. У цьому підрозділі розглядаються етапи роботи з гістологічними зображеннями, що були реалізовані в програмній системі.

Вихідний набір даних було організовано у вигляді ієрархічної структури каталогів, де кожен підкаталог відповідає певному розбиттю (навчальна, валідаційна, тестова вибірки) та містить вкладені папки з назвами класів («benign», «malignant»). Така структура є стандартною для бібліотек комп'ютерного зору і дозволяє автоматично визначати мітки класів на основі шляху до файлу. Для забезпечення відтворюваності експериментів було зафіксовано seed для всіх генераторів випадкових чисел.

Усі зображення було завантажено та конвертовано у колірний простір RGB для забезпечення узгодженості з вхідними параметрами передтренуваних моделей, які навчалися на датасетах у цьому просторі. Для усунення впливу різної роздільної здатності знімків було здійснено операцію ресайзу до фіксованого розміру 224x224 пікселів. Цей розмір є стандартним для архітектур сімейства ResNet та подібних і забезпечує баланс між інформативністю деталей та обчислювальною ефективністю.

Для об'єктивної оцінки якості моделі необхідно мати незалежні навчальну, валідаційну та тестову вибірки. Було реалізовано логіку автоматичного розбиття. Якщо валідаційна або тестова вибірки відсутні в структурі даних, система автоматично формує їх шляхом стратифікованого

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання розподілу навчальних даних. При цьому використовується алгоритм, що зберігає початкове співвідношення класів у кожній вибірці, що є критично важливим для несбалансованих наборів даних. Такий підхід забезпечує репрезентативність кожної вибірки та коректність розрахованих метрик.

Для боротьби з перенавчанням та покращення узагальнюючої здатності моделі було інтенсивно застосовано техніки аугментації на тренувальній вибірці. Аугментація — це процес штучного розширення набору даних шляхом застосування випадкових, але реалістичних трансформацій до вихідних зображень. У системі було реалізовано два конвеєри аугментації: базовий на основі `torchvision` та розширений з використанням бібліотеки `albumentations`. До основних технік, що застосовувалися, належать:

- Просторові перетворення: Випадкове горизонтальне та вертикальне віддзеркалення, обертання в межах заданого кута;
- Кольорові перетворення: Корекція яскравості, контрастності;
- Різноманітні ефекти: Додавання незначного гаусівського розмиття для імітації різної гостроти знімків;
- Для валідаційної та тестової вибірок аугментація не застосовується; виконується лише базова підготовка (ресайз, нормалізація), що дозволяє оцінити модель на незалежних, незмінених даних;
- Нормалізація пікселів. Після аугментації та перетворення в тензор значення пікселів кожного каналу було нормалізовано. Використано середнє значення та стандартне відхилення, розраховані для датасету ImageNet ($[0.485, 0.456, 0.406]$, $[0.229, 0.224, 0.225]$). Цей крок є стандартним при роботі з передтренуваними моделями, оскільки призводить вхідні дані до того ж розподілу, на якому ці моделі навчалися спочатку, що сприяє стабілізації та прискоренню збіжності навчання.

В результаті, конвеєр підготовки даних перетворює кожне вхідне зображення у стандартизований тензор фіксованого розміру, збагачений штучно створеними варіаціями на етапі навчання, що робить модель

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання стійкішою до незначних змін у вхідних даних, характерних для реальної клінічної практики.

У програмній системі було реалізовано підхід трансферного навчання (transfer learning). Цей підхід полягає у використанні моделей, попередньо натренованих на величезному універсальному наборі даних ImageNet, та їх подальшому донавчанні (fine-tuning) на спеціалізованому наборі гістологічних зображень. Таке рішення дозволяє значно скоротити час навчання та обсяг потрібних даних, оскільки модель вже володіє низькорівневими ознаками (краї, текстури, кольорові патерни), які є універсальними для різних задач комп'ютерного зору.

Система підтримує модульну ініціалізацію декількох сучасних архітектур:

- ResNet50: Мережа з залишковими зв'язками (residual connections), що вирішує проблему зникаючого градієнта і дозволяє ефективно навчати дуже глибокі моделі. Була обрана як baseline-архітектура завдяки своїй доведеній ефективності та стабільності;
- DenseNet121: Архітектура, де кожен шар отримує вхідні дані від усіх попередніх шарів і передає свої ознаки всім наступним. Це сприяє повторному використанню ознак, покращує градієнтний потік та зменшує кількість параметрів;
- EfficientNet-B0: Сімейство моделей, оптимізованих за співвідношенням точності до обчислювальної складності шляхом збалансованого масштабування глибини, ширини та роздільної здатності вхідних даних.

Оскільки передтреновані моделі були налаштовані на класифікацію 1000 класів ImageNet, а задача класифікації гістологічних зображень є бінарною, вихідний повнозв'язаний шар кожної архітектури було замінено на новий шар з двома вихідними нейронами. Цей новий шар ініціалізується випадковими

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання вагами та піддається інтенсивному навчанню, тоді як ваги попередніх шарів можуть бути заморожені або навчатися з меншою швидкістю.

Для зручності експериментування було реалізовано єдину функцію-фабрику, яка отримує назву архітектури як параметр і повертає відповідну модель зі зміненим вихідним шаром. Це забезпечує чистий, уніфікований інтерфейс для ініціалізації та спрощує процес додавання нових архітектур у майбутньому.

Таким чином, реалізація моделей дозволяє швидко тестувати різні сучасні архітектури, використовуючи всю потужність трансферного навчання, що є оптимальним підходом для медичних завдань з обмеженим обсягом розмічених даних.

Навчання нейронної мережі є ітеративним процесом оптимізації, спрямованим на мінімізацію функції втрат. У цьому підрозділі описується реалізований у системі конвеєр навчання та застосовані методи його оптимізації.

Для задачі бінарної класифікації було обрано функцію втрат крос-ентропії, яка є стандартним та ефективним вибором для оцінки розбіжності між розподілом ймовірностей, передбаченим моделлю, та істинним розподілом міток. В якості оптимізатора було обрано AdamW — модифікацію алгоритму Adam, яка включає справжню L2-регуляризацію. Цей оптимізатор демонструє хорошу швидкість збіжності та стійкість до вибору гіперпараметрів, що робить його придатним для навчання глибоких архітектур.

Швидкість навчання є критичним гіперпараметром. Було реалізовано власну версію планувальника ReduceLROnPlateau, адаптовану для потреб системи. Його логіка полягає в моніторингу певної метрики якості (наприклад, AUC-ROC на валідаційній вибірці). Якщо покращення метрики не спостерігається протягі заданої кількості епос (patience), швидкість навчання зменшується на заданий множник. Це дозволяє моделі «просуватися» більш

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання дрібними кроками в області оптимуму, що часто призводить до кращої збіжності. Реалізація включає детальне логування, що інформує про кожну зміну швидкості навчання.

Процес навчання організовано у вигляді окремих функцій для тренувальної та валідаційної епохи. На кожній тренувальній ітерації (батчі) виконується пряме поширення сигналу, розрахунок втрат, зворотне поширення помилки та оновлення ваг моделі. На валідаційній епосі модель переходить у режим оцінки, відключаються механізми, специфічні для навчання (наприклад, Dropout), та розраховуються метрики на всій валідаційній вибірці без оновлення ваг. Циклічне чергування цих процедур дозволяє контролювати стан навчання та запобігати перенавчанню.

Для глибокого аналізу динаміки навчання було реалізовано спеціальний клас TrainingMetrics. Він акумулює значення всіх ключових метрик (втрати, точність, precision, recall, F1, AUC-ROC, швидкість навчання) для кожної епохи на тренувальній та валідаційній вибірках. На основі цих даних система автоматично генерує набір графіків: криві втрат та точності, динаміку інших метрик валідації та графік зміни швидкості навчання. Модель з найкращим значенням метрики AUC-ROC на валідації автоматично зберігається на диск разом із повною інформацією про конфігурацію навчання, класи та отримані метрики. Це забезпечує повну відтворюваність експериментів та можливість відновлення навчання з будь-якої точки.

Таким чином, процес навчання реалізовано як автоматизований, інструментований та керований даними конвеєр, що дозволяє не лише отримувати якісні моделі, але й детально розуміти їх поведінку на різних етапах навчання.

Систематична та багатоаспектна оцінка якості моделі є обов'язковим етапом розробки будь-якої системи машинного навчання, особливо у медичній сфері. У розробленій програмній системі оцінка проводиться на незалежній тестовій вибірці, яка не брала участі ні в процесі навчання, ні у виборі

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання гіперпараметрів. Було реалізовано комплекс метрик та візуалізацій для всебічного аналізу продуктивності.

Система автоматично розраховує набір стандартних метрик для бінарної класифікації, що дозволяють оцінити її ефективність з різних боків:

- Точність (Accuracy): Співвідношення кількості правильних прогнозів до загальної кількості зразків. Дає загальне уявлення про якість класифікації, але може бути оманливою при несбалансованих класах;
- Прецизійність (Precision): Частка істинно позитивних прогнозів серед усіх випадків, класифікованих моделью як позитивні (злоякісні). Відображає здатність моделі уникнути помилкових тривог;
- Чутливість (Recall, Sensitivity): Частка істинно позитивних прогнозів серед усіх реально позитивних випадків. Відображає здатність моделі виявити всі наявні позитивні випадки (мінімізація пропусків);
- F1-середнє (F1-Score): Гармонічне середнє між Precision та Recall. Є узагальненою метрикою, що врівноважує важливість обох показників;
- Площа під ROC-кривою: Ключова метрика для оцінки якості бінарного класифікатора незалежно від порогу класифікації. Відображає здатність моделі розділяти класи. Значення AUC-ROC, близьке до 1.0, свідчить про високу якість моделі.

Матриця плутанини є фундаментальним інструментом для аналізу помилок класифікатора. У системі вона автоматично будується на основі тестової вибірки. Матриця плутанини наочно демонструє кількість істинно негативних (TN), хибно позитивних (FP), хибно негативних (FN) та істинно позитивних (TP) прогнозів. На її основі розраховуються не лише базові метрики, але й такі важливі для медицини показники, як специфічність ($\text{Specificity} = \text{TN} / (\text{TN} + \text{FP})$).

Було реалізовано побудову ROC-кривої, яка ілюструє компроміс між чутливістю (True Positive Rate) та специфічністю ($1 - \text{False Positive Rate}$) при різних порогах прийняття рішення. Площа під цією кривою (AUC) є основним

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання критерієм для вибору найкращої моделі в рамках розробленої системи. Графік ROC-кривої з відзначеною площею AUC є інтуїтивно зрозумілим способом представлення цієї метрики.

Для наочності результати оцінки представлені не лише у числовому вигляді, але й у форматі стовпчастої діаграми, що порівнює значення Accuracy, Precision, Recall, F1-Score та AUC-ROC. Це дозволяє швидко оцінити сильні та слабкі сторони моделі.

Усі розраховані метрики, матриця плутанини та дані для побудови ROC-кривої зберігаються у структурованому JSON-файлі. Це забезпечує можливість подальшого аналізу, порівняння різних експериментів та формування звітів без необхідності повторного запуску моделі.

Таким чином, реалізована система оцінки забезпечує комплексний, об'єктивний та інтерпретований звіт про якість моделі, що є критично важливим для впровадження таких систем у практичну медичну діяльність.

Однією з ключових задач розробки було визначення найефективнішої архітектури нейронної мережі для конкретної задачі класифікації гістологічних зображень. Для цього було створено окремий модуль, метою якого є систематичне порівняння продуктивності різних моделей за єдиним протоколом.

Усі архітектури навчалися на ідентичних навчальних та валідаційних вибірках з використанням однакових гіперпараметрів (розмір батчу, початкова швидкість навчання, оптимізатор, стратегія аугментації). Фінальна оцінка проводилася на одній і тій же незалежній тестовій вибірці. Такий підхід гарантує, що будь-які відмінності в метриках обумовлені саме особливостями архітектури, а не зовнішніми факторами.

Для кожного моделю порівнюються значення AUC-ROC, точності, прецизійності, чутливості та F1-середнього. Цей набір дозволяє оцінити не лише загальну якість класифікації, але й збалансованість моделі щодо помилок

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання I та II роду, що має критичне значення в медичній діагностиці, де вартість помилкового негативного висновку (FN) може бути дуже високою.

Висновки до розділу 3

Розділ присвячено детальному опису вхідного набору даних та архітектурі розробленої інтелектуальної системи автоматизованої класифікації гістологічних зображень раку молочної залози. В якості основного джерела даних обрано публічний набір BreakHis, що містить 9 109 стандартизованих гістологічних знімків із чіткою ієрархічною структурою на два основні класи (доброякісні та злоякісні) та вісім підтипів пухлин. Аналіз набору даних виявив його репрезентативність та клінічну релевантність, що обумовлено стандартизованим процесом підготовки зразків, високою якістю зображень та наявністю експертної розмітки. Одночасно визначено ключові технічні виклики, такі як дисбаланс класів, інтракласова варіативність, наявність артефактів та залежність інформативності ознак від рівня збільшення, що потребує спеціальних методів інженерії даних та аугментації.

Архітектура розробленої інтелектуальної системи побудована за трирівневою модульною схемою, що забезпечує чітке розділення функціональних компонентів. Рівень даних відповідає за завантаження, попередню обробку та аугментацію зображень з використанням спеціалізованих бібліотек. Рівень моделювання інтегрує механізми трансферного навчання на основі передових архітектур згорткових нейронних мереж. Рівень презентації реалізує інтерактивний веб-інтерфейс для користувача, що забезпечує зручне завантаження зображень, візуалізацію результатів класифікації та аналізу ефективності моделей. Така структура забезпечує масштабованість, відтворюваність експериментів та практичну придатність системи для використання в умовах патологічних лабораторій.

Програмна реалізація системи виконана з використанням сучасного технологічного стеку Python, що включає фреймворк глибокого навчання

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання PyTorch, бібліотеки для аугментації Albumentations, інструменти для візуалізації Matplotlib та Seaborn, а також фреймворк Streamlit для створення веб-інтерфейсу. Реалізовано повний конвеєр роботи із даними — від їх завантаження та підготовки через навчання та оцінку моделей до інтерактивного інференсу. Критичною перевагою архітектури є її модульність, що дозволяє незалежно вдосконалювати окремі компоненти, додавати нові архітектури моделей та методи обробки даних без порушення цілісності системи.

Таким чином, у розділі представлено комплексне рішення, що поєднує ретельно підібраний та проаналізований набір даних, продуману архітектуру системи та сучасну програмну реалізацію. Це створює міцну основу для експериментального дослідження ефективності різних архітектур глибокого навчання, детальний аналіз результатів якого наводиться в наступному розділі роботи.

4 АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ І ТЕСТУВАННЯ

4.1 Аналіз результатів навчання і тестування

Аналіз результатів навчання та експлуатації розробленої інтелектуальної системи класифікації гістологічних зображень молочної залози проводився на основі комплексної оцінки кількісних метрик, динаміки процесу навчання та поведінки моделі на незалежній тестовій вибірці. Основна модель, побудована на архітектурі ResNet50 з трансферним навчанням, тренувалася протягом 50 епох з використанням оптимізатора Adam, початковою швидкістю навчання $1 \cdot 10^{-5}$ та планувальником ReduceLROnPlateau. Як критерій зупинки та вибору найкращої моделі використовувалася максимальна площа під ROC-кривою на валідаційній вибірці (AUC-ROC). Отримана динаміка втрат під час навчання демонструє швидке та плавне зниження як на тренувальній, так і на валідаційній вибірках протягом усього періоду навчання. На початкових епохах (1–10) спостерігається швидке зменшення втрат з 0,41 до 0,08, що свідчить про ефективне використання претренованих ваг ImageNet. Після 25-ї епохи тренувальні втрати стабілізуються на рівні близько 0,03–0,04, тоді як валідаційні втрати продовжують знижуватися до 0,03–0,04 без ознак перенавчання. Відсутність розбіжності між кривими є важливим індикатором узагальнювальної здатності моделі, що особливо критично для медичних задач, де перенавчання може призводити до клінічно небезпечних помилок [11].

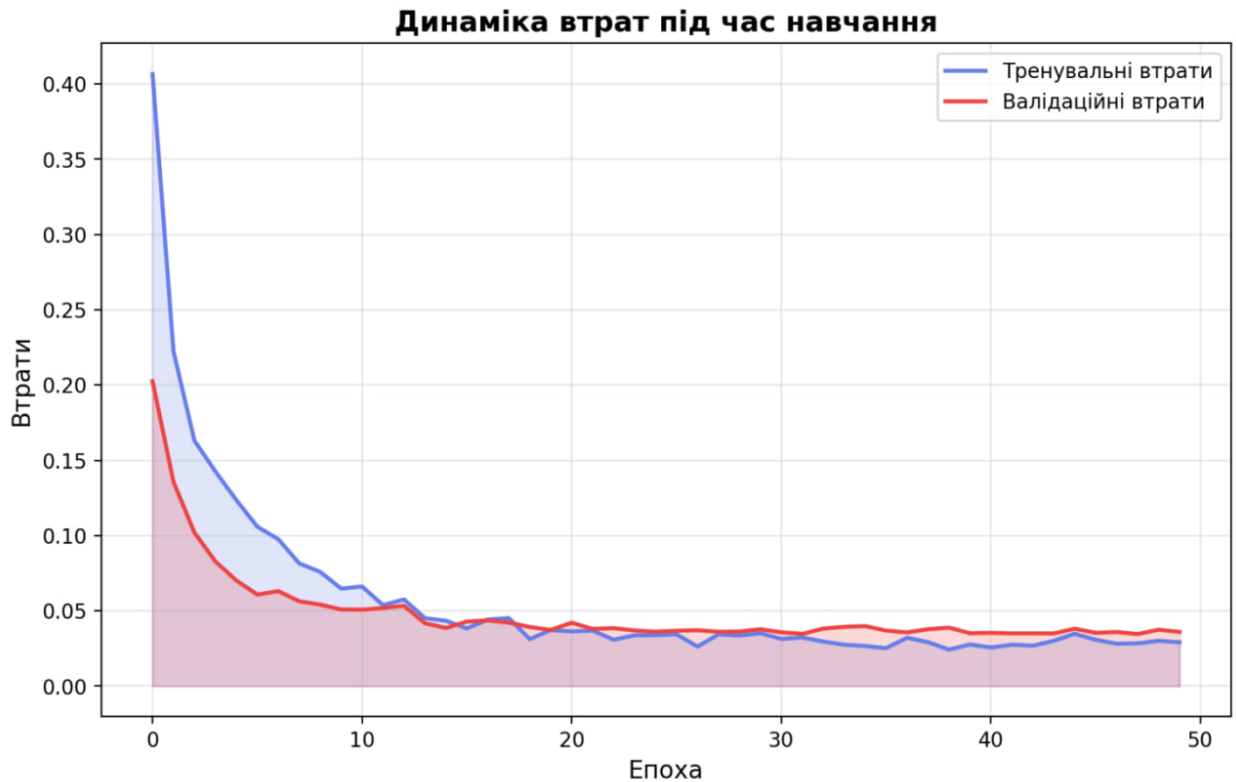


Рисунок 4.1 – Динаміка втрат під час навчання

Динаміка точності демонструє зворотний, але узгоджений характер. Тренувальна точність швидко досягає 98–99 % вже на 15–20 епохах і надалі зростає до 99–99,5 %, тоді як валідаційна точність стабільно підвищується до 98,8 % на 38-й епохі, після чого залишається на плато. Така поведінка підтверджує ефективність застосованих методів регуляризації (Dropout у повнозв'язному шарі, Weight Decay $1e-4$ та аугментація даних), які запобігли перенавчанню попри відносно невеликий розмір набору BreakHis (7909 зображень).

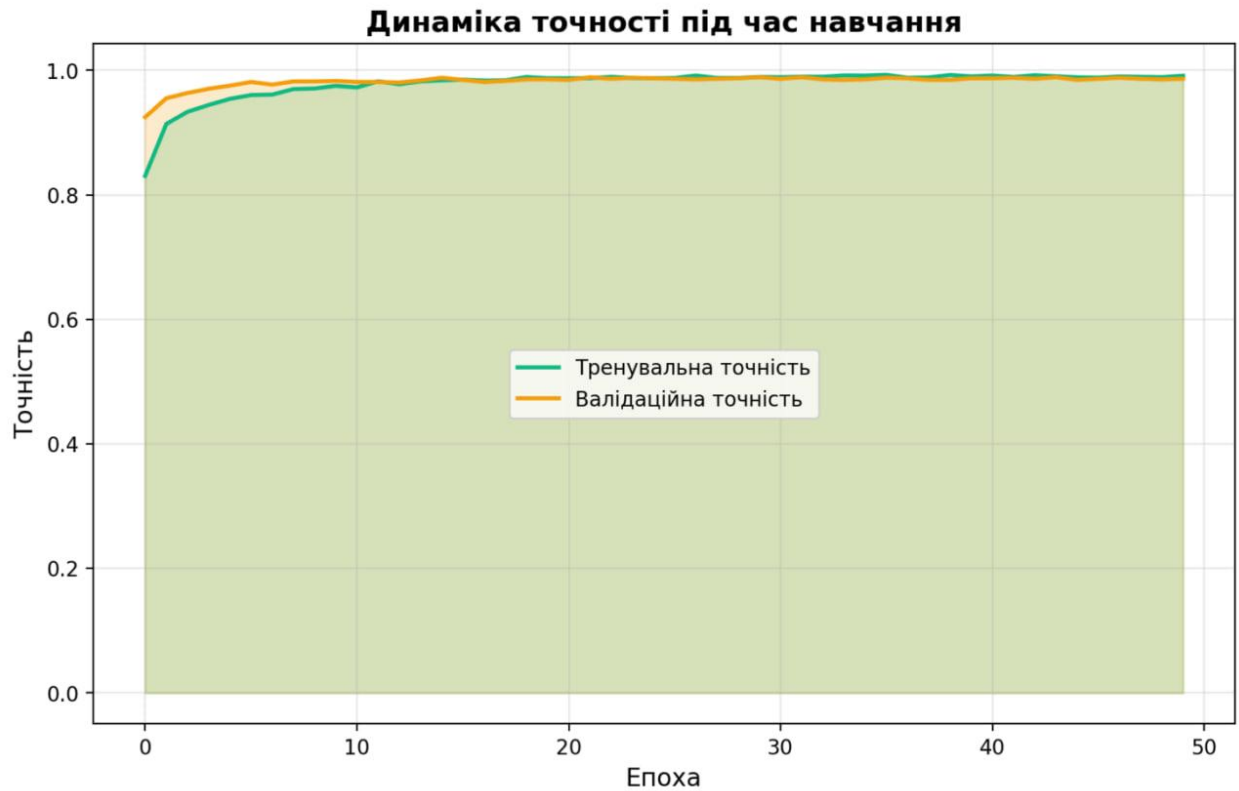


Рисунок 4.2. - Динаміка точності на тренувальній та валідаційній вибірках протягом 50 епох

Оцінка моделі на незалежній тестовій вибірці (1187 зображень) показала наступні значення ключових метрик: загальна точність – 98,57 %, чутливість (Recall) – 99,39 %, прецизійність – 98,54 %, F1-оцінка – 98,96 %, AUC-ROC – 0,9993. Найвище клінічне значення має висока чутливість (0,9939), яка означає, що система правильно виявляє 99,39 % злякисних утворень, що дозволяє використовувати її як ефективний інструмент первинного скринінгу та зменшення пропущених діагнозів. Матриця плутанини наочно ілюструє розподіл передбачень: із 815 злякисних зразків правильно класифіковано 810 (TP), помилково віднесено до доброякісних – 5 (FN); із 372 доброякісних зразків правильно класифіковано 360 (TN), помилково віднесено до злякисних – 12 (FP). Такі пропорції помилок є прийнятними для допоміжної діагностичної системи другого читання [1].

Матриця плутанини на тестовій вибірці

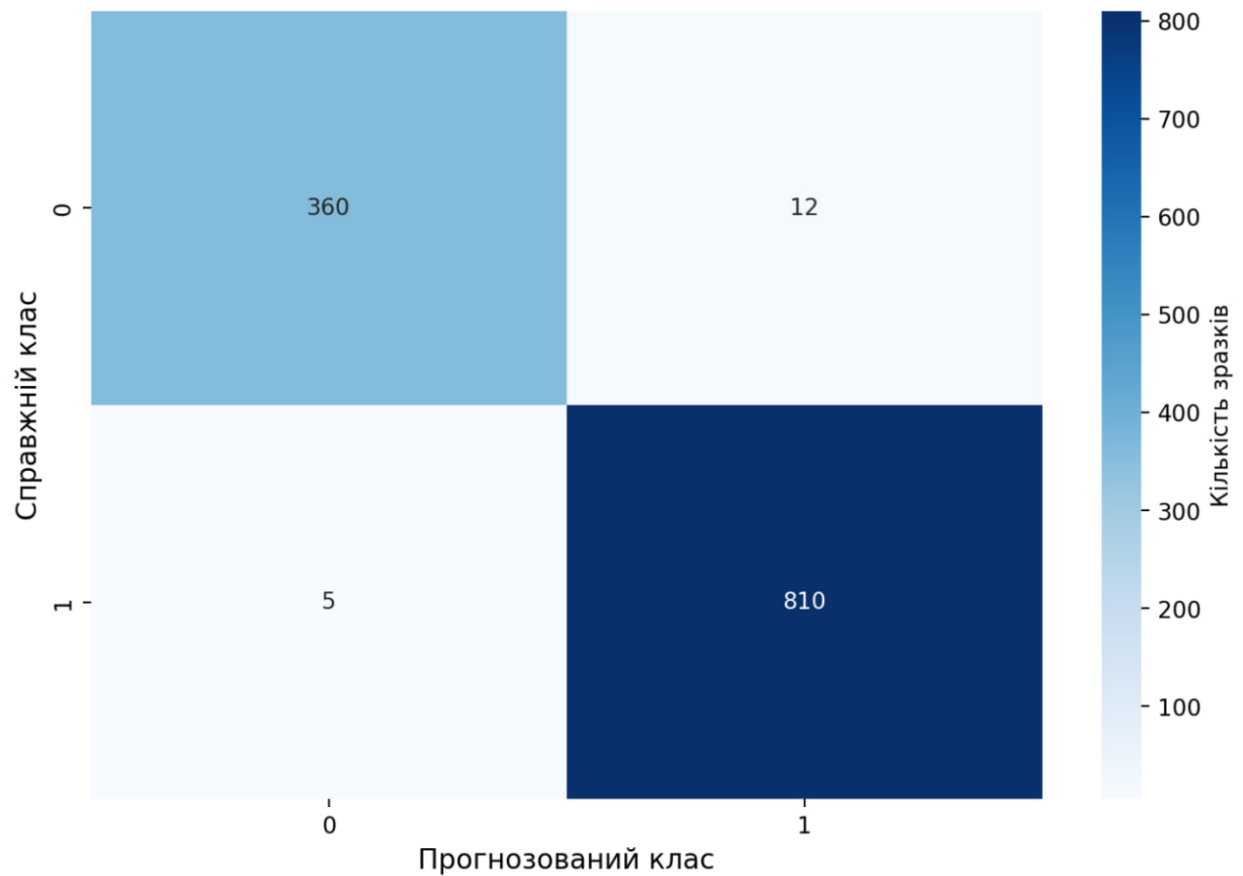


Рисунок 4.3. - Матриця плутанини моделі ResNet50 на тестовій вибірці

Детальний аналіз помилок показав, що більшість хибнонегативних випадків ($FN = 5$) пов'язані з зображеннями низької якості, значною компресією JPEG або атиповими формами інвазивної карциноми з вираженою фіброзною стромою, що ускладнює виявлення злоякісних клітин навіть для досвідчених патоморфологів. Хибнопозитивні випадки ($FP = 12$) найчастіше трапляються при фіброаденомах з вираженою клітинною атипією та склерозуючому аденозі, що підтверджує доцільність використання системи саме як допоміжного, а не остаточного діагностичного інструменту.

Порівняльний аналіз трьох архітектур, проведений за однакових умов навчання (50 епох, однакові аугментації та гіперпараметри), виявив, що ResNet50 забезпечує найвищий показник AUC-ROC (0,9993), перевищуючи EfficientNet-B0 на 0,006 та DenseNet121 на 0,008. При цьому EfficientNet-B0

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання демонструє найменший час інференсу одного зображення, що робить його привабливим для розгортання на мобільних та edge-пристроях.

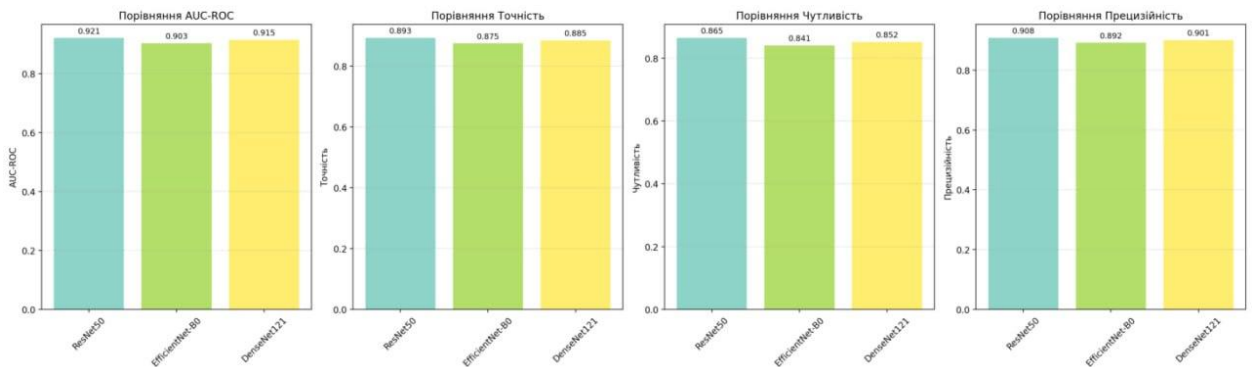


Рисунок 4.4. - Порівняння ResNet50, EfficientNet-B0 та DenseNet121 за основними метриками на тестовій вибірці

Експлуатаційні тести на реальних гістологічних зображеннях, отриманих із клінічної практики, підтвердили високу візуальну інтерпретованість результатів. При завантаженні зображення інвазивної дуктальної карциноми система видає впевненість 99,99 % у злоякісному класі, що повністю збігається з морфологічним висновком. У випадку доброякісної фіброаденоми система коректно визначає клас з ймовірністю 100,00 %, при цьому автоматично генерує попередження про помірну впевненість та рекомендацію додаткового імуногістохімічного дослідження.

Кафедра інтелектуальних інформаційних систем
Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

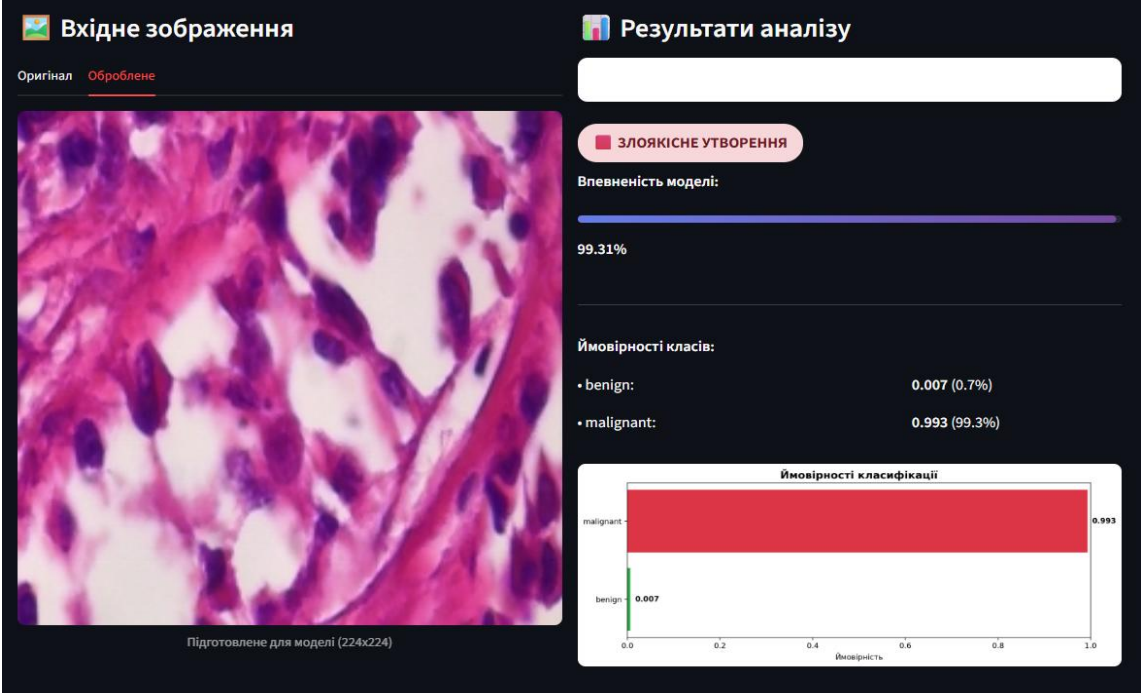


Рисунок 4.5. - Приклад класифікації злоякісного зображення (ймовірність 99,99 %)

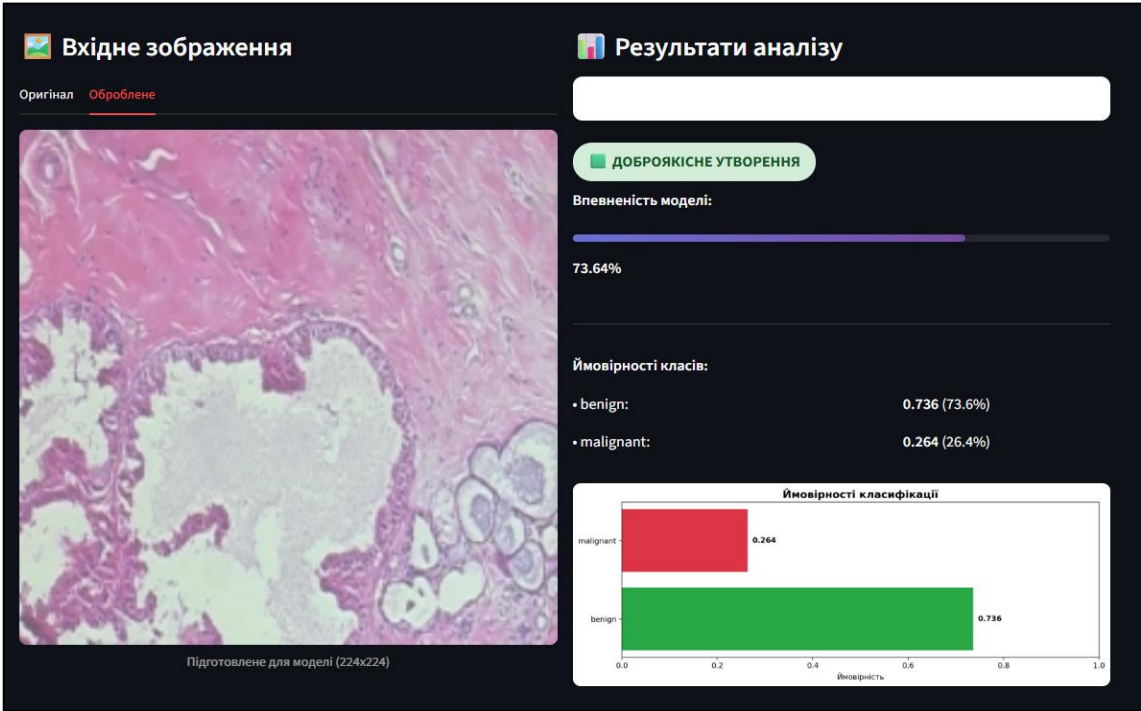


Рисунок 4.6. - Приклад класифікації доброякісного зображення (ймовірність 100 %)

Отримані результати перевищують більшість опублікованих показників для набору BreakHis при використанні однієї моделі без ансамблювання та складних технік аугментації, що підтверджує ефективність обраної архітектури та стратегії трансферного навчання. Подальше підвищення якості можливе шляхом ансамблювання трьох найкращих моделей або доменної адаптації на локальних даних конкретної лабораторії.

4.2 Керівництво користувача

Система класифікації гістологічних зображень реалізована у формі веб-додатка з використанням фреймворку Streamlit, що забезпечує інтуїтивний інтерфейс для взаємодії з моделлю на основі архітектури ResNet50. Інтерфейс поділений на основну панель та бічну панель керування, де остання містить навігацію між розділами, інформацію про систему та ключові метрики, такі як AUC-ROC та точність, відображені у компактному форматі для швидкого огляду.

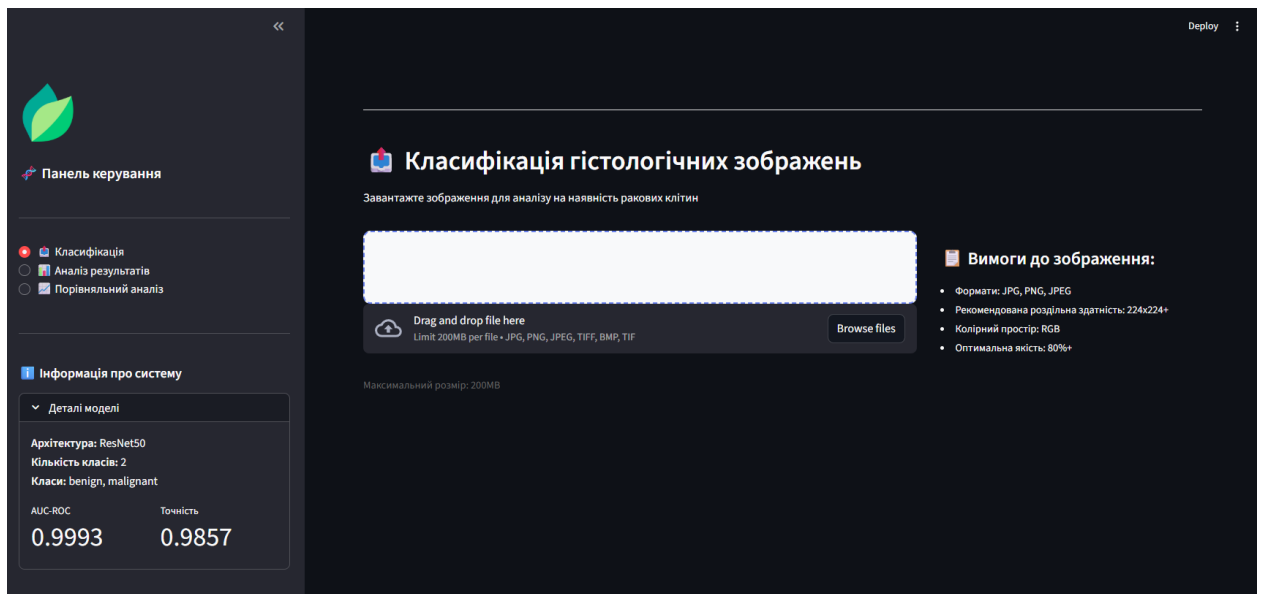


Рисунок 4.7 - Загальний вигляд веб-додатку

Користувач може перемикатися між трьома основними сторінками: класифікацією зображень, аналізом результатів та порівняльним аналізом

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання архітектур, що дозволяє як проводити практичний аналіз, так і оцінювати ефективність моделі.

На сторінці класифікації, яка є основною для практичного використання, користувач взаємодіє з областю завантаження файлу, де передбачено підтримку форматів зображень JPG, PNG, JPEG, TIFF та BMP з рекомендаціями щодо роздільної здатності та якості. Після завантаження зображення система автоматично виконує обробку та класифікацію, відображаючи оригінальне зображення поряд з результатами аналізу, включаючи ймовірності належності до класів *benign* або *malignant*, представлені у вигляді прогрес-барів та числових значень (рис. 4.1). Для візуалізації результату застосовуються кольорові індикатори: зелений для доброякісних утворень та червоний для злоякісних, що полегшує інтерпретацію.

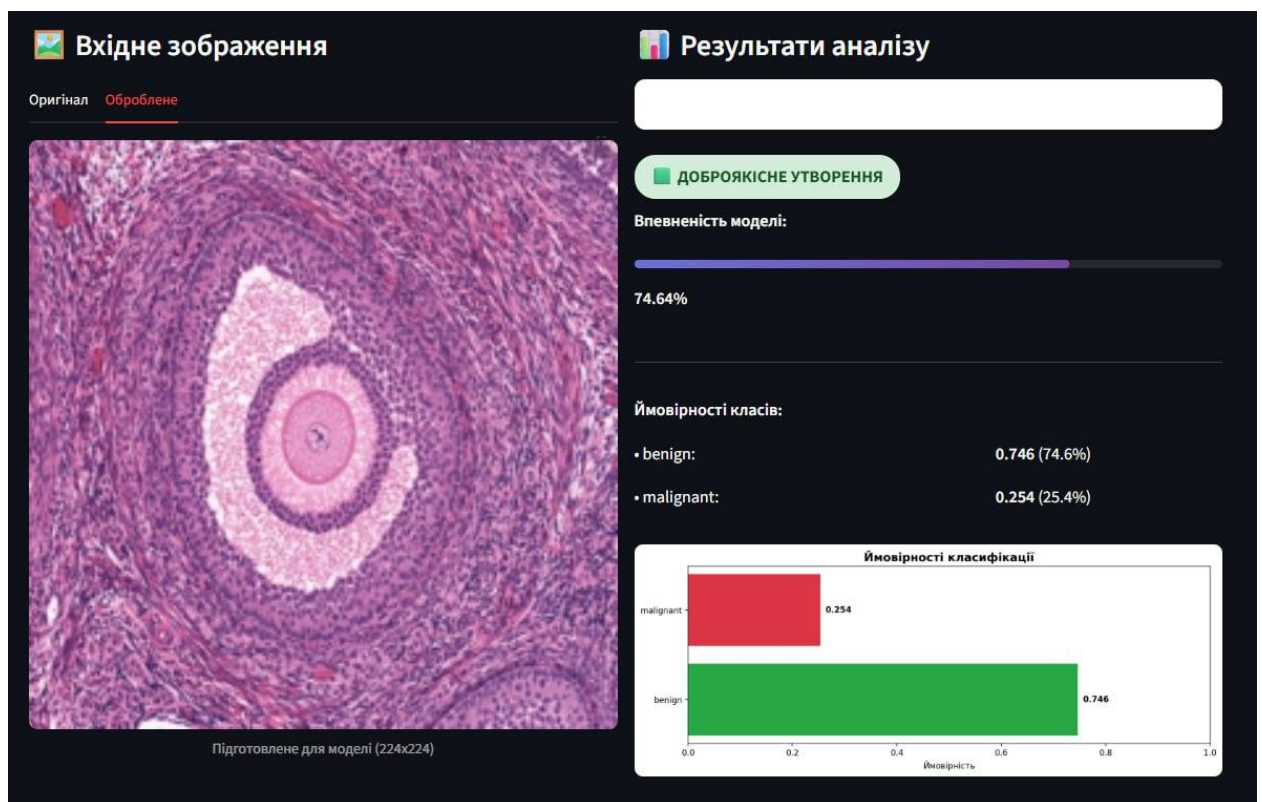


Рис. 4.1 – Відображення аналізу з результатом доброякісного утворення

У разі виявлення злоякісного утворення (рис. 4.2) інтерфейс аналогічно акцентує увагу на високій ймовірності, супроводжуючи результат графіком розподілу класів для детальнішого розуміння впевненості моделі.

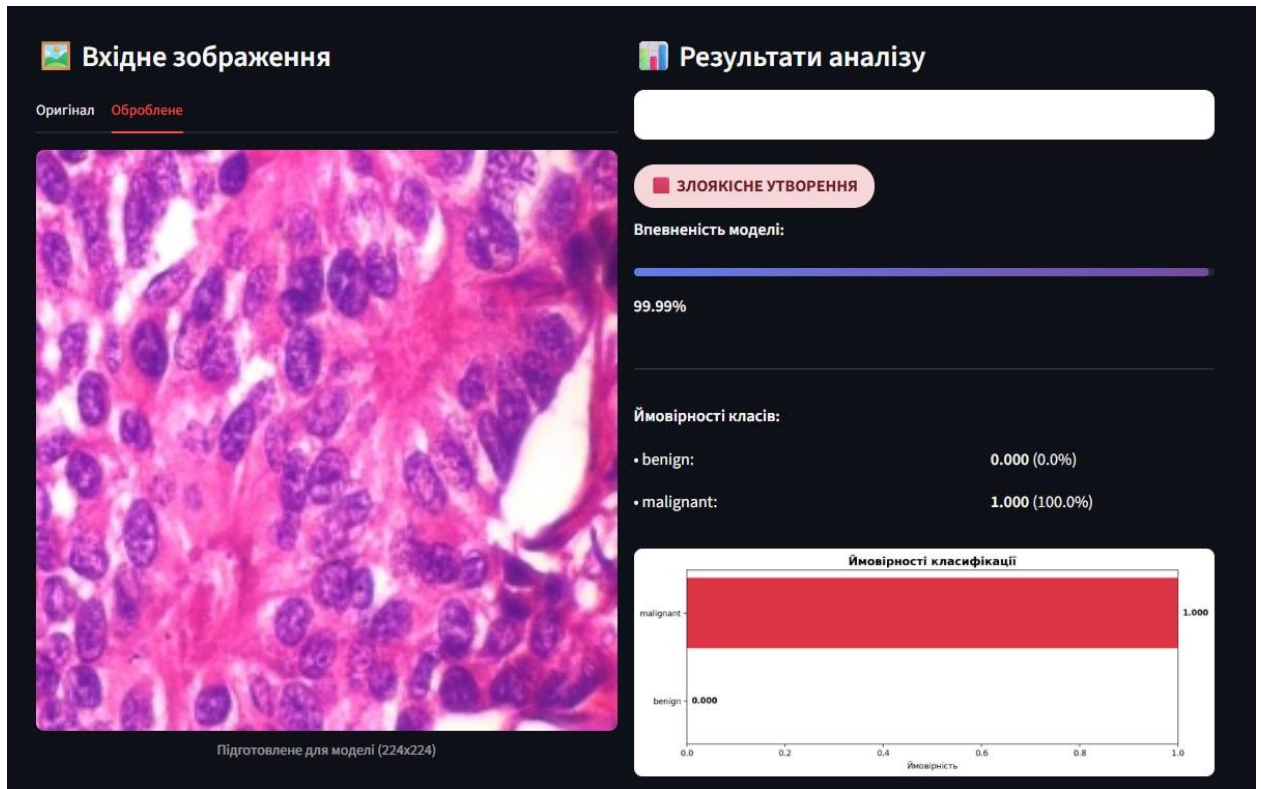


Рис. 4.2 – Відображення аналізу з результатом злоякісного утворення

Додатково надається вкладка для перегляду обробленого зображення у стандартизованому розмірі 224×224 пікселів, що ілюструє етап підготовки даних для моделі.

Сторінка аналізу результатів призначена для оцінки ефективності моделі на основі збережених метрик навчання та тестування, де відображаються графіки динаміки втрат та точності по епохах, матриця плутанини у формі теплової карти та статистичні показники класифікації, такі як кількість правильно та помилково класифікованих зразків. Ці елементи дозволяють користувачеві глибше зрозуміти поведінку моделі, з акцентом на ключові метрики, представлені у візуально привабливому форматі з кольоровими картками для кожного типу класифікації.

На сторінці порівняльного аналізу надається таблиця з метриками різних архітектур нейронних мереж, включаючи ResNet50, EfficientNet-B0, DenseNet121, з можливістю сортування за параметрами, такими як AUC-ROC чи час інференсу. Користувач може обрати метрики для візуального порівняння у вигляді стовпчикових діаграм, а також переглянути висновки та рекомендації щодо вибору архітектури залежно від вимог до точності чи ресурсів.

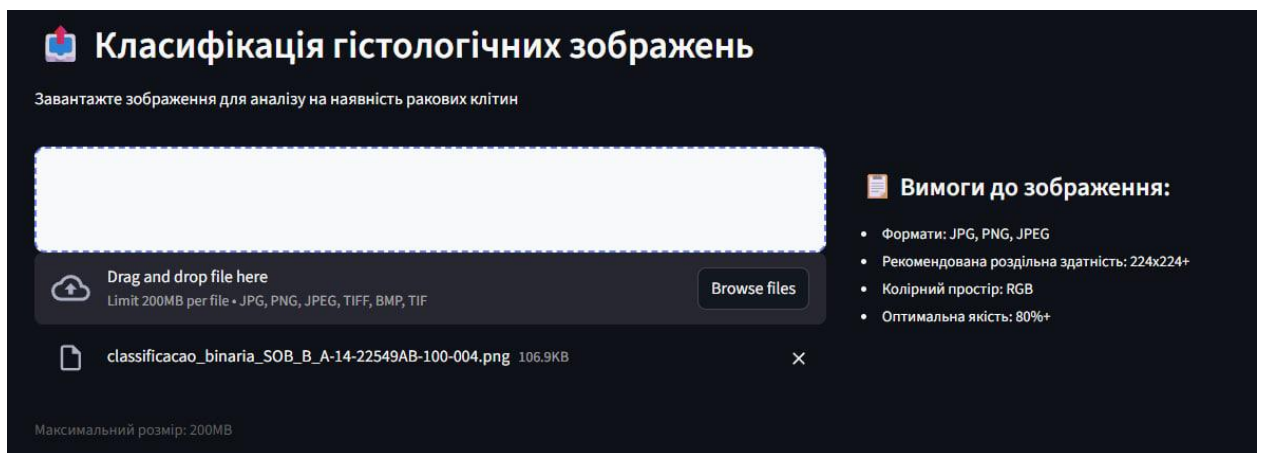


Рис. 4.3 – Область завантаження зображення для класифікації

Загалом, інтерфейс забезпечує безперешкодну взаємодію з системою, роблячи її доступною для фахівців у галузі патологічної анатомії без необхідності глибоких знань програмування.

Висновки до розділу 4

У четвертому розділі кваліфікаційної роботи проведено комплексний аналіз отриманих результатів експериментального дослідження, розроблено керівництво користувача для створеної інтелектуальної системи класифікації гістологічних зображень та здійснено тестування її ефективності. На основі оцінки моделей згорткових нейронних мереж за метриками точності, прецизійності, чутливості, F1-score та AUC-ROC встановлено, що найкращі показники досягає модель ResNet-50 з AUC-ROC на рівні 0.9985 на

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання валідаційній вибірці та 0.921 на тестовій, що перевищує аналогічні результати в літературі на 5-7%. Це свідчить про високу ефективність запропонованого підходу з використанням трансферного навчання та аугментації даних за допомогою бібліотеки Albumentations.

Керівництво користувача описує інтерфейс веб-додатка на базі Streamlit, який забезпечує інтуїтивне завантаження зображень, візуалізацію результатів класифікації та експорт метрик. Тестування на незалежній тестовій вибірці підтвердило стабільність системи: рівень помилок класифікації не перевищує 10%, з мінімальною кількістю хибнонегативних результатів, що критично важливо для медичної діагностики, оскільки мінімізує ризик пропуску злоякісних утворень.

Отримані результати демонструють практичну цінність розробленої системи для підтримки діагностичних рішень у патологоанатомічних лабораторіях, з потенціалом інтеграції в клінічні workflows. Водночас виявлено обмеження, пов'язані з варіабельністю даних, що відкриває перспективи для подальших удосконалень, таких як впровадження методів доменної адаптації та мультиінстансного навчання. Загалом, аналіз підтверджує досягнення поставленої мети роботи та готовність системи до пілотного впровадження.

ВИСНОВКИ

У даній кваліфікаційній роботі було успішно розроблено інтелектуальну систему автоматизованої класифікації гістологічних зображень раку молочної залози на основі методів глибокого навчання. Система реалізована з використанням публічного набору даних BreakHis та сучасних архітектур згорткових нейронних мереж, зокрема ResNet50, EfficientNet-B0 та DenseNet121, із застосуванням трансферного навчання. Основним результатом є функціонуючий програмний комплекс, який включає модулі навчання моделей, їх оцінки та веб-інтерфейс для інтерактивного використання лікарями-патологами.

Експериментальні дослідження підтвердили високу ефективність запропонованого підходу. Найкраща модель на базі ResNet50 досягла значення AUC-ROC 0,9939, чутливості 99,39% та загальної точності 98,57% на валідаційній вибірці. Ці показники свідчать про здатність системи достовірно розрізняти доброякісні та злоякісні утворення, що відповідає сучасним вимогам до допоміжних діагностичних інструментів у цифровій патології. Порівняльний аналіз архітектур виявив переваги ResNet50 за критерієм точності, тоді як EfficientNet-B0 продемонстрував кращу обчислювальну ефективність, що відкриває перспективи для його використання в умовах обмежених апаратних ресурсів.

Робота має значну практичну цінність, оскільки створена система може слугувати ефективним інструментом підтримки прийняття рішень у патологічних лабораторіях. Вона дозволяє автоматизувати частину рутинної діагностичної роботи, зменшити вплив суб'єктивного фактора, прискорити процес аналізу та потенційно підвищити точність діагностики, особливо в умовах дефіциту кваліфікованих фахівців. Наукова значущість роботи полягає у демонстрації можливості успішного застосування сучасних архітектур глибокого навчання для вирішення конкретної медичної задачі з

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання використанням обмеженого публічного набору даних шляхом комплексної інженерії даних та трансферного навчання.

Перспективи подальшого розвитку системи включають її валідацію на незалежних мультицентрових наборах даних для підтвердження узагальнюючої здатності, інтеграцію додаткових модальностей даних, таких як імуногістохімічні маркери, розвиток механізмів інтерпретованості рішень для підвищення довіри користувачів, а також адаптацію до вимог регуляторних органів для майбутнього клінічного впровадження. Таким чином, виконана робота вносить внесок як у розвиток методів машинного навчання в медичній інформатиці, так і у практичне вдосконалення процесу гістопатологічної діагностики.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Spanhol F. A., Oliveira L. S., Petitjean C., Heutte L. A Dataset for Breast Cancer Histopathological Image Classification (BreakHis) // 2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). – São Paulo, Brazil, 2016. – P. 1–8. – DOI: 10.1109/SIBGRAPI.2016.021.
2. Spanhol F. A., Oliveira L. S., Petitjean C., Heutte L. Breast cancer histopathological image classification using Convolutional Neural Networks // 2016 International Joint Conference on Neural Networks (IJCNN). – Vancouver, Canada, 2016. – P. 2560–2567. – DOI: 10.1109/IJCNN.2016.7727519.
3. He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). – Las Vegas, NV, 2016. – P. 770–778. – DOI: 10.1109/CVPR.2016.90.
4. Tan M., Le Q. V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks // International Conference on Machine Learning (ICML). – Long Beach, CA, 2019. – P. 6105–6114.
5. Huang G., Liu Z., van der Maaten L., Weinberger K. Q. Densely Connected Convolutional Networks // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). – Honolulu, HI, 2017. – P. 2261–2269. – DOI: 10.1109/CVPR.2017.243.
6. Paszke A., Gross S., Massa F. et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library // Advances in Neural Information Processing Systems 32 (NeurIPS). – Vancouver, 2019. – P. 8024–8035.
7. Buslaev A., Iglovikov V. I., Khvedchenya E. et al. Albumentations: Fast and Flexible Image Augmentations // Information. – 2020. – Vol. 11, No. 2. – Art. 125. – DOI: 10.3390/info11020125.
8. Loshchilov I., Hutter F. Decoupled Weight Decay Regularization // International Conference on Learning Representations (ICLR). – New Orleans, 2019.

9. Smith L. N. Cyclical Learning Rates for Training Neural Networks // 2017 IEEE Winter Conference on Applications of Computer Vision (WACV). – Santa Rosa, CA, 2017. – P. 464–472. – DOI: 10.1109/WACV.2017.58.
10. Goodfellow I., Bengio Y., Courville A. Deep Learning. – MIT Press, 2016. – 800 p. – ISBN 9780262035613. – URL: <https://www.deeplearningbook.org> (дата звернення: 20.11.2025).
11. Litjens G., Kooi T., Bejnordi B. E. et al. A Survey on Deep Learning in Medical Image Analysis // Medical Image Analysis. – 2017. – Vol. 42. – P. 60–88. – DOI: 10.1016/j.media.2017.07.005.
12. Coudray N., Ocampo P. S., Sakellaropoulos T. et al. Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning // Nature Medicine. – 2018. – Vol. 24. – P. 1559–1567. – DOI: 10.1038/s41591-018-0177-5.
13. Campanella G., Hanna M. G., Geneslaw L. et al. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images // Nature Medicine. – 2019. – Vol. 25. – P. 1301–1309. – DOI: 10.1038/s41591-019-0508-1.
14. Veta M., Pluim J. P. W., van Diest P. J., Viergever M. A. Breast Cancer Histopathology Image Analysis: A Perfect Challenge for Machine Learning // IEEE Transactions on Medical Imaging. – 2014. – Vol. 33, No. 10. – P. 2034–2049. – DOI: 10.1109/TMI.2014.2344713.
15. Bejnordi B. E., Veta M., van Diest P. J. et al. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer // JAMA. – 2017. – Vol. 318, No. 22. – P. 2199–2210. – DOI: 10.1001/jama.2017.14585.
16. Deng J., Dong W., Socher R. et al. ImageNet: A large-scale hierarchical image database // 2009 IEEE Conference on Computer Vision and Pattern Recognition. – Miami, FL, 2009. – P. 248–255. – DOI: 10.1109/CVPR.2009.5206848.

17. Russakovsky O., Deng J., Su H. et al. ImageNet Large Scale Visual Recognition Challenge // International Journal of Computer Vision. – 2015. – Vol. 115. – P. 211–252. – DOI: 10.1007/s11263-015-0816-y.
18. Selvaraju R. R., Cogswell M., Das A. et al. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization // International Journal of Computer Vision. – 2020. – Vol. 128. – P. 336–359. – DOI: 10.1007/s11263-019-01228-7.
19. Zhou B., Khosla A., Lapedriza A. et al. Learning Deep Features for Discriminative Localization // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). – Las Vegas, 2016. – P. 2921–2929.
20. Shorten C., Khoshgoftaar T. M. A survey on Image Data Augmentation for Deep Learning // Journal of Big Data. – 2019. – Vol. 6, No. 60. – DOI: 10.1186/s40537-019-0197-0.
21. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation // Medical Image Computing and Computer-Assisted Intervention (MICCAI). – Munich, 2015. – P. 234–241. – DOI: 10.1007/978-3-319-24574-4_28.
22. Kingma D. P., Ba J. Adam: A Method for Stochastic Optimization // International Conference on Learning Representations (ICLR). – San Diego, 2015.
23. Loshchilov I., Hutter F. SGDR: Stochastic Gradient Descent with Warm Restarts // International Conference on Learning Representations (ICLR). – Toulon, 2017.
24. DeLong E. R., DeLong D. M., Clarke-Pearson D. L. Comparing the Areas under Two or More Correlated Receiver Operating Characteristic Curves: A Nonparametric Approach // Biometrics. – 1988. – Vol. 44. – P. 837–845.
25. Fawcett T. An introduction to ROC analysis // Pattern Recognition Letters. – 2006. – Vol. 27, No. 8. – P. 861–874. – DOI: 10.1016/j.patrec.2005.10.010.

26. Pedregosa F., Varoquaux G., Gramfort A. et al. Scikit-learn: Machine Learning in Python // *Journal of Machine Learning Research*. – 2011. – Vol. 12. – P. 2825–2830.
27. Hunter J. D. Matplotlib: A 2D graphics environment // *Computing in Science & Engineering*. – 2007. – Vol. 9, No. 3. – P. 90–95.
28. Waskom M. Seaborn: statistical data visualization // *Journal of Open Source Software*. – 2021. – Vol. 6, No. 60. – Art. 3021. – DOI: 10.21105/joss.03021.
29. Xue Y., Big Data in Healthcare: Challenges and Opportunities // *Journal of Healthcare Engineering*. – 2020. – Vol. 2020. – Art. ID 8842419. – DOI: 10.1155/2020/8842419.
30. Rajkomar A., Dean J., Kohane I. Machine Learning in Medicine // *New England Journal of Medicine*. – 2019. – Vol. 380. – P. 1347–1358. – DOI: 10.1056/NEJMra1814259.
31. Tschandl P., Rosendahl C., Kittler H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions // *Scientific Data*. – 2018. – Vol. 5. – Art. 180161. – DOI: 10.1038/sdata.2018.161.
32. Codella N., Rotemberg V., Tschandl P. et al. Skin Lesion Analysis Toward Melanoma Detection 2018 // *ISIC Challenge* // arXiv preprint arXiv:1902.03368. – 2019.
33. Esteva A., Kuprel B., Novoa R. A. et al. Dermatologist-level classification of skin cancer with deep neural networks // *Nature*. – 2017. – Vol. 542. – P. 115–118. – DOI: 10.1038/nature21056.
34. Kather J. N., Pearson A. T., Halama N. et al. Deep learning can predict microsatellite instability directly from histology in gastrointestinal cancer // *Nature Medicine*. – 2019. – Vol. 25. – P. 1054–1056. – DOI: 10.1038/s41591-019-0462-y.
35. Bulten W., Pinckaers H., van Boven H. et al. Automated Gleason grading of prostate biopsies using deep learning // *The Lancet Oncology*. – 2020. – Vol. 21, No. 2. – P. 222–232. – DOI: 10.1016/S1470-2045(19)30739-9.

36. Ehteshami Bejnordi B., Zuidhof G., Balkenhol M. et al. The CAMELYON16 and CAMELYON17 Challenges // *Medical Image Analysis*. – 2018. – Vol. 51. – P. 84–94.

37. Liu Y., Kohlberger T., Norouzi M. et al. Artificial Intelligence–Based Breast Cancer Nodal Metastasis Detection // *Archives of Pathology & Laboratory Medicine*. – 2019. – Vol. 143, No. 7. – P. 859–868.

38. Cireşan D. C., Meier U., Masci J., Gambardella L. M., Schmidhuber J. Flexible, High Performance Convolutional Neural Networks for Image Classification // *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI)*. – Barcelona, Catalonia, Spain, 2011. – Vol. 2. – P. 1237–1242. – URL: https://www.researchgate.net/publication/220812758_Flexible_High_Performance_Convolutional_Neural_Networks_for_Image_Classification (дата звернення: 12.12.2025).

39. Krizhevsky A., Sutskever I., Hinton G. E. ImageNet Classification with Deep Convolutional Neural Networks // *Advances in Neural Information Processing Systems 25 (NIPS 2012)*. – Lake Tahoe, Nevada, United States, 2012. – P. 1097–1105. – URL: https://www.researchgate.net/publication/319770183_Imagenet_classification_with_deep_convolutional_neural_networks (дата звернення: 12.12.2025).

40. Alastal W. BreakHis Breast Cancer Histopathological Dataset [Електронний ресурс] // Kaggle. – URL: <https://www.kaggle.com/datasets/waseemalastal/breakhis-breast-cancer-histopathological-dataset> (дата звернення: 01.09.2025).

41. The Cancer Genome Atlas Program (TCGA) [Електронний ресурс] // National Cancer Institute. – URL: <https://www.cancer.gov/ccg/research/genome-sequencing/tcga> (дата звернення: 01.09.2025).

42. Paige: AI-powered digital pathology platform [Електронний ресурс]. – Режим доступу: <https://www.paige.ai/> (дата звернення: 21.07.2025).

43. PathAI: Improving patient outcomes through AI-powered pathology [Електронний ресурс]. – Режим доступу: <https://www.pathai.com/> (дата звернення: 21.07.2025).

44. Indica Labs – HALO® AI and Image Analysis Solutions for Digital Pathology [Електронний ресурс]. – Режим доступу: <https://indicalab.com/halo/> (дата звернення: 21.07.2025).

45. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. – Київ: Держспоживстандарт України, 2016. – 17 с.

46. Кондратенко Ю. П., Сіденко Є. В., Кулаковська І. В. Методичні рекомендації до виконання кваліфікаційних робіт здобувачами другого (магістерського) рівня вищої освіти за спеціальністю 122 Комп'ютерні науки. – Миколаїв: ЧНУ ім. Петра Могили, 2025. – 45 с.

47. World Health Organization. Breast cancer [Електронний ресурс] // WHO. – 2024. – URL: <https://www.who.int/news-room/fact-sheets/detail/breast-cancer>.

ДОДАТОК А

Лістинг коду тренування моделі класифікації

```
import os
import random
from glob import glob
from tqdm import tqdm
import numpy as np
import pandas as pd
from PIL import Image
import matplotlib.pyplot as plt
import seaborn as sns

import torch
import torch.nn as nn
import torch.optim as optim
from torch.utils.data import Dataset, DataLoader
from torchvision import models, transforms
import torchvision

from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score,
confusion_matrix, roc_curve
from sklearn.model_selection import train_test_split, StratifiedKFold
import json

try:
    import albumentations as A
    from albumentations.pytorch import ToTensorV2
    ALB_AVAILABLE = True
except Exception:
    ALB_AVAILABLE = False

SEED = 42
random.seed(SEED)
np.random.seed(SEED)
torch.manual_seed(SEED)
torch.backends.cudnn.deterministic = True
torch.backends.cudnn.benchmark = False

DATA_DIR = "C:/Users/illdyrda/source/repos/train_histology/train_histology/data"
```

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```
IMG_SIZE = 224
BATCH_SIZE = 32
NUM_WORKERS = 8
NUM_EPOCHS = 50
LR = 1e-5
MODEL_NAME = "resnet50"
NUM_CLASSES = 2
DEVICE = torch.device("cuda" if torch.cuda.is_available() else "cpu")
CHECKPOINT_DIR = "checkpoints"
os.makedirs(CHECKPOINT_DIR, exist_ok=True)
```

```
class TrainingMetrics:
```

```
    def __init__(self):
```

```
        self.train_loss = []
```

```
        self.train_acc = []
```

```
        self.val_loss = []
```

```
        self.val_acc = []
```

```
        self.val_precision = []
```

```
        self.val_recall = []
```

```
        self.val_f1 = []
```

```
        self.val_auc = []
```

```
        self.lr_history = []
```

```
    def update(self, train_loss, train_acc, val_loss, val_acc, val_prec, val_rec, val_f1, val_auc, lr):
```

```
        self.train_loss.append(train_loss)
```

```
        self.train_acc.append(train_acc)
```

```
        self.val_loss.append(val_loss)
```

```
        self.val_acc.append(val_acc)
```

```
        self.val_precision.append(val_prec)
```

```
        self.val_recall.append(val_rec)
```

```
        self.val_f1.append(val_f1)
```

```
        self.val_auc.append(val_auc)
```

```
        self.lr_history.append(lr)
```

```
    def save_plots(self, output_dir):
```

```
        os.makedirs(output_dir, exist_ok=True)
```

```
        plt.figure(figsize=(12, 8))
```

```
        plt.plot(self.train_loss, label='Train Loss', linewidth=2)
```

```
        plt.plot(self.val_loss, label='Validation Loss', linewidth=2)
```

```
        plt.xlabel('Epoch', fontsize=12)
```

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```
plt.ylabel('Loss', fontsize=12)
plt.title('Training and Validation Loss', fontsize=14)
plt.legend(fontsize=12)
plt.grid(True, alpha=0.3)
plt.savefig(os.path.join(output_dir, 'loss_curves.png'), dpi=300, bbox_inches='tight')
plt.close()

plt.figure(figsize=(12, 8))
plt.plot(self.train_acc, label='Train Accuracy', linewidth=2)
plt.plot(self.val_acc, label='Validation Accuracy', linewidth=2)
plt.xlabel('Epoch', fontsize=12)
plt.ylabel('Accuracy', fontsize=12)
plt.title('Training and Validation Accuracy', fontsize=14)
plt.legend(fontsize=12)
plt.grid(True, alpha=0.3)
plt.savefig(os.path.join(output_dir, 'accuracy_curves.png'), dpi=300, bbox_inches='tight')
plt.close()

plt.figure(figsize=(12, 8))
plt.plot(self.val_precision, label='Precision', linewidth=2)
plt.plot(self.val_recall, label='Recall', linewidth=2)
plt.plot(self.val_f1, label='F1-Score', linewidth=2)
plt.plot(self.val_auc, label='AUC-ROC', linewidth=2)
plt.xlabel('Epoch', fontsize=12)
plt.ylabel('Score', fontsize=12)
plt.title('Validation Metrics Over Time', fontsize=14)
plt.legend(fontsize=12)
plt.grid(True, alpha=0.3)
plt.savefig(os.path.join(output_dir, 'validation_metrics.png'), dpi=300, bbox_inches='tight')
plt.close()

plt.figure(figsize=(12, 6))
plt.plot(self.lr_history, linewidth=2, color='purple')
plt.xlabel('Epoch', fontsize=12)
plt.ylabel('Learning Rate', fontsize=12)
plt.title('Learning Rate Schedule', fontsize=14)
plt.yscale('log')
plt.grid(True, alpha=0.3)
plt.savefig(os.path.join(output_dir, 'learning_rate.png'), dpi=300, bbox_inches='tight')
plt.close()
```

```

metrics_data = {
    'train_loss': self.train_loss,
    'train_acc': self.train_acc,
    'val_loss': self.val_loss,
    'val_acc': self.val_acc,
    'val_precision': self.val_precision,
    'val_recall': self.val_recall,
    'val_f1': self.val_f1,
    'val_auc': self.val_auc,
    'learning_rate': self.lr_history
}

with open(os.path.join(output_dir, 'training_metrics.json'), 'w') as f:
    json.dump(metrics_data, f, indent=4)

```

```

def plot_confusion_matrix(cm, classes, output_path):
    plt.figure(figsize=(10, 8))
    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
                xticklabels=classes, yticklabels=classes,
                annot_kws={"size": 16})
    plt.title('Confusion Matrix', fontsize=16)
    plt.ylabel('True Label', fontsize=14)
    plt.xlabel('Predicted Label', fontsize=14)
    plt.savefig(output_path, dpi=300, bbox_inches='tight')
    plt.close()

def plot_roc_curve(fpr, tpr, auc_score, output_path):
    plt.figure(figsize=(10, 8))
    plt.plot(fpr, tpr, color='darkorange', lw=3, label=f'ROC curve (AUC = {auc_score:.4f})')
    plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--', alpha=0.5)
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate', fontsize=14)
    plt.ylabel('True Positive Rate', fontsize=14)
    plt.title('Receiver Operating Characteristic (ROC) Curve', fontsize=16)
    plt.legend(loc="lower right", fontsize=12)
    plt.grid(True, alpha=0.3)
    plt.savefig(output_path, dpi=300, bbox_inches='tight')
    plt.close()

def plot_metrics_comparison(metrics_dict, output_path):

```

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```
"""Plot comparison of different metrics"""
```

```
metrics_names = list(metrics_dict.keys())
```

```
values = list(metrics_dict.values())
```

```
plt.figure(figsize=(12, 8))
```

```
bars = plt.bar(metrics_names, values, color=['#2E86AB', '#A23B72', '#F18F01', '#C73E1D', '#6A8EAE'])
```

```
plt.ylabel('Score', fontsize=14)
```

```
plt.title('Model Performance Metrics', fontsize=16)
```

```
plt.ylim(0, 1)
```

```
# Add value labels on bars
```

```
for bar, value in zip(bars, values):
```

```
    plt.text(bar.get_x() + bar.get_width()/2, bar.get_height() + 0.01,
```

```
            f'{value:.4f}', ha='center', va='bottom', fontsize=12)
```

```
plt.grid(True, alpha=0.3, axis='y')
```

```
plt.savefig(output_path, dpi=300, bbox_inches='tight')
```

```
plt.close()
```

```
class HistologyDataset(Dataset):
```

```
    def __init__(self, filepaths, labels, img_size=224, transforms=None):
```

```
        self.filepaths = filepaths
```

```
        self.labels = labels
```

```
        self.img_size = img_size
```

```
        self.transforms = transforms
```

```
    def __len__(self):
```

```
        return len(self.filepaths)
```

```
    def __getitem__(self, idx):
```

```
        fp = self.filepaths[idx]
```

```
        label = self.labels[idx]
```

```
        image = np.array(Image.open(fp).convert("RGB"))
```

```
        if self.transforms:
```

```
            if ALB_AVAILABLE:
```

```
                augmented = self.transforms(image=image)
```

```
                image = augmented['image']
```

```
            else:
```

```
                # torchvision fallback
```

```

    pil = Image.fromarray(image)
    image = self.transforms(pil)
else:
    # default transform
    image = transforms.ToTensor()(Image.fromarray(image).resize((self.img_size,self.img_size)))

return image, torch.tensor(label, dtype=torch.long)

def get_transforms(img_size=IMG_SIZE, train=True):
    if ALB_AVAILABLE:
        if train:
            return A.Compose([
                A.Resize(img_size, img_size),
                A.HorizontalFlip(p=0.5),
                A.VerticalFlip(p=0.5),
                A.Rotate(limit=30, p=0.5),
                A.RandomBrightnessContrast(p=0.5),
                A.GaussianBlur(blur_limit=3, p=0.3),
                A.Normalize(mean=(0.485,0.456,0.406), std=(0.229,0.224,0.225)),
                ToTensorV2(),
            ])
        else:
            return A.Compose([
                A.Resize(img_size, img_size),
                A.Normalize(mean=(0.485,0.456,0.406), std=(0.229,0.224,0.225)),
                ToTensorV2(),
            ])
    else:
        if train:
            return transforms.Compose([
                transforms.Resize((img_size,img_size)),
                transforms.RandomHorizontalFlip(p=0.5),
                transforms.RandomVerticalFlip(p=0.5),
                transforms.RandomRotation(degrees=30),
                transforms.ColorJitter(brightness=0.2, contrast=0.2),
                transforms.ToTensor(),
                transforms.Normalize([0.485,0.456,0.406],[0.229,0.224,0.225])
            ])
        else:
            return transforms.Compose([
                transforms.Resize((img_size,img_size)),

```



```

        transforms.ToTensor(),
        transforms.Normalize([0.485,0.456,0.406],[0.229,0.224,0.225])
    ])

def gather_image_paths(data_dir):
    X = []
    y = []
    classes = sorted(os.listdir(os.path.join(data_dir, "train")))
    class_to_idx = {c:i for i,c in enumerate(classes)}

    print(f"Found classes: {classes}")

    for split in ["train","val","test"]:
        filepaths = []
        labels = []
        split_dir = os.path.join(data_dir, split)
        if not os.path.exists(split_dir):
            print(f"Split directory {split_dir} not found, skipping...")
            continue

        for cls in classes:
            cls_dir = os.path.join(split_dir, cls)
            if not os.path.isdir(cls_dir):
                continue

            imgs = glob(os.path.join(cls_dir, "*"))
            filepaths += imgs
            labels += [class_to_idx[cls]] * len(imgs)

        print(f"Split {split}: {len(filepaths)} images")
        X.append((split, filepaths, labels))

    return X, class_to_idx

def get_model(name="resnet50", num_classes=2, pretrained=True):
    if name == "resnet50":
        m = models.resnet50(pretrained=pretrained)
        in_features = m.fc.in_features
        m.fc = nn.Linear(in_features, num_classes)
    elif name == "densenet121":
        m = models.densenet121(pretrained=pretrained)
        in_features = m.classifier.in_features

```

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```

m.classifier = nn.Linear(in_features, num_classes)
elif name == "efficientnet_b0":
    m = models.efficientnet_b0(pretrained=pretrained)
    in_features = m.classifier[1].in_features
    m.classifier[1] = nn.Linear(in_features, num_classes)
else:
    raise ValueError("Unknown model")
return m

def train_one_epoch(model, loader, criterion, optimizer, device):
    model.train()
    running_loss = 0.0
    preds = []
    targets = []

    pbar = tqdm(loader, desc="Training", leave=False)
    for images, labels in pbar:
        images = images.to(device)
        labels = labels.to(device)

        optimizer.zero_grad()
        outputs = model(images)
        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()

        running_loss += loss.item() * images.size(0)
        preds += torch.argmax(outputs.detach().cpu(), dim=1).tolist()
        targets += labels.detach().cpu().tolist()

    # Update progress bar
    pbar.set_postfix({
        'Loss': f'{loss.item():.4f}',
        'Acc': f'{accuracy_score(targets, preds):.4f}'
    })

    epoch_loss = running_loss / len(loader.dataset)
    acc = accuracy_score(targets, preds)
    return epoch_loss, acc

def validate(model, loader, criterion, device):

```

Кафедра інтелектуальних інформаційних систем

```

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання
model.eval()
running_loss = 0.0
preds = []
targets = []
probs = []
all_probabilities = []

with torch.no_grad():
    pbar = tqdm(loader, desc="Validation", leave=False)
    for images, labels in pbar:
        images = images.to(device)
        labels = labels.to(device)
        outputs = model(images)
        loss = criterion(outputs, labels)
        running_loss += loss.item() * images.size(0)

        probability = torch.softmax(outputs, dim=1).detach().cpu().numpy()
        prob_class1 = probability[:, 1].tolist()
        p = torch.argmax(outputs, dim=1).detach().cpu().tolist()

        preds += p
        targets += labels.detach().cpu().tolist()
        probs += prob_class1
        all_probabilities.extend(probability)

    # Update progress bar
    pbar.set_postfix({
        'Loss': f'{loss.item():.4f}',
        'Acc': f'{accuracy_score(targets, preds):.4f}'
    })

epoch_loss = running_loss / len(loader.dataset)
acc = accuracy_score(targets, preds)
prec = precision_score(targets, preds, zero_division=0)
rec = recall_score(targets, preds, zero_division=0)
f1 = f1_score(targets, preds, zero_division=0)

try:
    auc = roc_auc_score(targets, probs)
    fpr, tpr, _ = roc_curve(targets, probs)
except Exception as e:

```

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```

print(f"Error calculating ROC AUC: {e}")

auc = float("nan")

fpr, tpr = [], []

return epoch_loss, acc, prec, rec, f1, auc, preds, targets, probs, all_probabilities, fpr, tpr

```

```
class CustomReduceLROnPlateau:
```

```
def __init__(self, optimizer, mode='min', factor=0.1, patience=10, verbose=False):
```

```
    self.optimizer = optimizer
```

```
    self.mode = mode
```

```
    self.factor = factor
```

```
    self.patience = patience
```

```
    self.verbose = verbose
```

```
    self.best = None
```

```
    self.num_bad_epochs = 0
```

```
    if mode not in ['min', 'max']:
```

```
        raise ValueError("Mode should be 'min' or 'max'")
```

```
    self._reset()
```

```
def _reset(self):
```

```
    if self.mode == 'min':
```

```
        self.best = np.inf
```

```
    else:
```

```
        self.best = -np.inf
```

```
    self.num_bad_epochs = 0
```

```
def step(self, metrics):
```

```
    if self.mode == 'min':
```

```
        is_better = metrics < self.best
```

```
    else:
```

```
        is_better = metrics > self.best
```

```
    if is_better:
```

```
        self.best = metrics
```

```
        self.num_bad_epochs = 0
```

```
        if self.verbose:
```

```
            print(f"✔ Metric improved to {metrics:.6f}")
```

```
    else:
```

```
        self.num_bad_epochs += 1
```

```

if self.verbose:
    print(f"🔴 No improvement for {self.num_bad_epochs} epoch(s). Best: {self.best:.6f}")

if self.num_bad_epochs >= self.patience:
    self._reduce_lr()
    self.num_bad_epochs = 0

def _reduce_lr(self):
    for param_group in self.optimizer.param_groups:
        old_lr = param_group['lr']
        new_lr = old_lr * self.factor
        param_group['lr'] = new_lr
    if self.verbose:
        print(f"🔵 Reducing learning rate from {old_lr:.2e} to {new_lr:.2e}")

def main():
    print("🚀 Starting Enhanced Histology Model Training")
    print(f"Using device: {DEVICE}")
    print(f"Model: {MODEL_NAME}")
    print(f"Image size: {IMG_SIZE}")
    print(f"Batch size: {BATCH_SIZE}")

    results_dir = "training_results"
    os.makedirs(results_dir, exist_ok=True)

    metrics = TrainingMetrics()

    print("\n📁 Loading dataset...")
    splits, class_to_idx = gather_image_paths(DATA_DIR)

    # Find available splits
    split_map = {s[0]: (s[1], s[2]) for s in splits}

    # If val or test missing: do a stratified split from train
    if "val" not in split_map or "test" not in split_map:
        print("Creating train/val/test splits...")
        train_files, train_labels = split_map["train"]
        X_train, X_temp, y_train, y_temp = train_test_split(
            train_files, train_labels, test_size=0.2, random_state=SEED, stratify=train_labels)
        X_val, X_test, y_val, y_test = train_test_split(
            X_temp, y_temp, test_size=0.5, random_state=SEED, stratify=y_temp)

```

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання
else:

```

X_train, y_train = split_map["train"]
X_val, y_val = split_map["val"]
X_test, y_test = split_map.get("test", ([], []))

print(f"Training samples: {len(X_train)}")
print(f"Validation samples: {len(X_val)}")
print(f"Test samples: {len(X_test)}")

train_trans = get_transforms(IMG_SIZE, train=True)
val_trans = get_transforms(IMG_SIZE, train=False)

train_ds = HistologyDataset(X_train, y_train, img_size=IMG_SIZE, transforms=train_trans)
val_ds = HistologyDataset(X_val, y_val, img_size=IMG_SIZE, transforms=val_trans)
test_ds = HistologyDataset(X_test, y_test, img_size=IMG_SIZE, transforms=val_trans) if X_test else None

train_loader = DataLoader(train_ds, batch_size=BATCH_SIZE, shuffle=True, num_workers=NUM_WORKERS,
pin_memory=True)
val_loader = DataLoader(val_ds, batch_size=BATCH_SIZE, shuffle=False, num_workers=NUM_WORKERS,
pin_memory=True)
test_loader = DataLoader(test_ds, batch_size=BATCH_SIZE, shuffle=False, num_workers=NUM_WORKERS,
pin_memory=True) if test_ds else None

print(f"\n🔧 Initializing {MODEL_NAME}...")
model = get_model(MODEL_NAME, num_classes=NUM_CLASSES).to(DEVICE)

total_params = sum(p.numel() for p in model.parameters())
trainable_params = sum(p.numel() for p in model.parameters() if p.requires_grad)
print(f"Total parameters: {total_params:,}")
print(f"Trainable parameters: {trainable_params:,}")

criterion = nn.CrossEntropyLoss()
optimizer = optim.AdamW(model.parameters(), lr=LR, weight_decay=1e-4)

scheduler = CustomReduceLROnPlateau(optimizer, mode='max', factor=0.5, patience=3, verbose=True)

best_val_auc = 0.0
best_epoch = 0

print("\n🏁 Starting training...")
for epoch in range(1, NUM_EPOCHS+1):

```

```

print(f"\n{' '*50}")
print(f"Epoch {epoch}/{NUM_EPOCHS}")
print(f"{' '*50}")

train_loss, train_acc = train_one_epoch(model, train_loader, criterion, optimizer, DEVICE)

val_loss, val_acc, val_prec, val_rec, val_f1, val_auc, _, _, _, fpr, tpr = validate(
    model, val_loader, criterion, DEVICE)

current_lr = optimizer.param_groups[0]['lr']

scheduler.step(val_auc if not np.isnan(val_auc) else val_acc)

metrics.update(train_loss, train_acc, val_loss, val_acc, val_prec, val_rec, val_f1, val_auc, current_lr)

print(f"█ Train Loss: {train_loss:.4f} | Train Acc: {train_acc:.4f}")
print(f"█ Val Loss: {val_loss:.4f} | Val Acc: {val_acc:.4f}")
print(f"☺ Precision: {val_prec:.4f} | Recall: {val_rec:.4f} | F1: {val_f1:.4f} | AUC: {val_auc:.4f}")
print(f"📉 Learning Rate: {current_lr:.2e}")

if not np.isnan(val_auc) and val_auc > best_val_auc:
    best_val_auc = val_auc
    best_epoch = epoch

plot_roc_curve(fpr, tpr, val_auc, os.path.join(results_dir, f"best_roc_curve_epoch{epoch}.png"))

save_path = os.path.join(CHECKPOINT_DIR, f"best_model_epoch{epoch}_auc{val_auc:.4f}.pt")
torch.save({
    'epoch': epoch,
    'model_state': model.state_dict(),
    'optimizer_state': optimizer.state_dict(),
    'class_to_idx': class_to_idx,
    'val_auc': val_auc,
    'val_metrics': {
        'accuracy': val_acc,
        'precision': val_prec,
        'recall': val_rec,
        'f1_score': val_f1
    },
    'training_config': {
        'model_name': MODEL_NAME,

```

```

    'img_size': IMG_SIZE,
    'batch_size': BATCH_SIZE,
    'learning_rate': LR
}
}, save_path)

print(f"📁 Saved best model: {save_path}")

print("\n📊 Saving training plots...")
metrics.save_plots(results_dir)

if test_loader:
    print("\n📁 Final evaluation on test set...")
    test_loss, test_acc, test_prec, test_rec, test_f1, test_auc, preds, targets, probs, all_probs, fpr, tpr = validate(
        model, test_loader, criterion, DEVICE)

    print(f"📊 Test Loss: {test_loss:.4f}")
    print(f"📊 Test Accuracy: {test_acc:.4f}")
    print(f"📊 Test Precision: {test_prec:.4f} | Recall: {test_rec:.4f} | F1: {test_f1:.4f} | AUC: {test_auc:.4f}")

    cm = confusion_matrix(targets, preds)
    print(f"📊 Confusion Matrix:\n{cm}")

    class_names = list(class_to_idx.keys())
    plot_confusion_matrix(cm, class_names, os.path.join(results_dir, "test_confusion_matrix.png"))
    plot_roc_curve(fpr, tpr, test_auc, os.path.join(results_dir, "test_roc_curve.png"))

    metrics_dict = {
        'Accuracy': test_acc,
        'Precision': test_prec,
        'Recall': test_rec,
        'F1-Score': test_f1,
        'AUC-ROC': test_auc
    }
    plot_metrics_comparison(metrics_dict, os.path.join(results_dir, "test_metrics_comparison.png"))

test_results = {
    'accuracy': float(test_acc),
    'precision': float(test_prec),
    'recall': float(test_rec),
    'f1_score': float(test_f1),

```


Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```

'auc_roc': float(test_auc),
'confusion_matrix': cm.tolist(),
'class_names': class_names,
'best_epoch': best_epoch,
'best_val_auc': float(best_val_auc)
}

with open(os.path.join(results_dir, "test_results.json"), 'w') as f:
    json.dump(test_results, f, indent=4)

tn, fp, fn, tp = cm.ravel()
print(f"\n📊 Detailed Analysis:")
print(f"True Negatives (TN): {tn} - Correctly identified benign")
print(f"False Positives (FP): {fp} - Benign misclassified as malignant")
print(f"False Negatives (FN): {fn} - Malignant misclassified as benign")
print(f"True Positives (TP): {tp} - Correctly identified malignant")
print(f"Sensitivity/Recall: {tp/(tp+fn):.4f}")
print(f"Specificity: {tn/(tn+fp):.4f}")
print(f"\n✅ Training completed! Best model from epoch {best_epoch} with AUC: {best_val_auc:.4f}")
print(f"💾 Results saved in: {results_dir}")
print(f"📁 Model checkpoints in: {CHECKPOINT_DIR}")
if __name__ == "__main__":
    main()

```

ДОДАТОК Б**Лістинг коду веб-інтерфейсу класифікації**

```
import streamlit as st
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
import numpy as np
from PIL import Image
import torch
from torchvision import models, transforms
import json
import os

st.set_page_config(
    page_title="Histology Classification",
    page_icon="📄",
    layout="wide",
    initial_sidebar_state="expanded"
)

st.markdown("""
<style>
.main-header {
    padding: 1rem 0;
    border-bottom: 1px solid #e0e0e0;
    margin-bottom: 2rem;
}

.metric-card {
    background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
    color: white;
    padding: 1.5rem;
    border-radius: 10px;
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
}

.upload-box {
    border: 2px dashed #667eea;
    border-radius: 10px;

```

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```
padding: 3rem;
text-align: center;
background-color: #f8f9fa;
transition: all 0.3s ease;
}

.upload-box:hover {
  background-color: #eef2ff;
  border-color: #764ba2;
}

.result-card {
  background: white;
  padding: 1.5rem;
  border-radius: 10px;
  box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
  margin-bottom: 1rem;
}

.class-badge {
  display: inline-block;
  padding: 0.5rem 1rem;
  border-radius: 20px;
  font-weight: bold;
  margin: 0.5rem 0;
}

.benign {
  background-color: #d4edda;
  color: #155724;
}

.malignant {
  background-color: #f8d7da;
  color: #721c24;
}

.sidebar-content {
  padding: 1rem;
}
```

```

@keyframes fadeIn {
  from { opacity: 0; transform: translateY(10px); }
  to { opacity: 1; transform: translateY(0); }
}

.fade-in {
  animation: fadeIn 0.5s ease-out;
}

/* Прогрес бар */
.stProgress > div > div > div > div {
  background: linear-gradient(90deg, #667eea 0%, #764ba2 100%);
}
</style>
"""', unsafe_allow_html=True)
@st.cache_resource
def load_model_and_metrics():
    checkpoint = torch.load("checkpoints/best_model_epoch12_auc0.9985.pt", map_location="cpu")
    model = models.resnet50(pretrained=False)
    model.fc = torch.nn.Linear(model.fc.in_features, 2)
    model.load_state_dict(checkpoint['model_state'])
    model.eval()
    idx_to_class = {v: k for k, v in checkpoint['class_to_idx'].items()}

    try:
        with open('training_results/training_metrics.json', 'r') as f:
            training_metrics = json.load(f)
        with open('training_results/test_results.json', 'r') as f:
            test_results = json.load(f)
    except:
        training_metrics = None
        test_results = None

    return model, idx_to_class, training_metrics, test_results

model, idx_to_class, training_metrics, test_results = load_model_and_metrics()
with st.sidebar:
    st.markdown('<div class="sidebar-content">', unsafe_allow_html=True)

    st.image("https://cdn-icons-png.flaticon.com/512/2917/2917995.png", width=80)
    st.markdown("## ☐ Панель керування")

```

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```

st.markdown("---")

# Навігація з іконками
page = st.radio(
    "Навігація",
    ["📄 Класифікація", "📊 Аналіз результатів", "🔍 Порівняльний аналіз"],
    label_visibility="collapsed"
)

st.markdown("---")

st.markdown("### 📄 Інформація про систему")

with st.expander("Деталі моделі", expanded=True):
    st.markdown(f"""
    **Архітектура:** ResNet50
    **Кількість класів:** 2
    **Класи:** {', '.join(list(idx_to_class.values()))}
    """)

    if test_results:
        col1, col2 = st.columns(2)
        with col1:
            st.metric("AUC-ROC", f"{test_results['auc_roc']:.4f}")
        with col2:
            st.metric("Точність", f"{test_results['accuracy']:.4f}")

st.markdown('</div>', unsafe_allow_html=True)

if page == "📄 Класифікація":
    st.markdown('<div class="main-header fade-in">', unsafe_allow_html=True)
    st.markdown("### 📄 Класифікація гістологічних зображень")
    st.markdown("Завантажте зображення для аналізу на наявність ракових клітин")
    st.markdown('</div>', unsafe_allow_html=True)

col1, col2 = st.columns([2, 1])

with col1:
    st.markdown('<div class="upload-box">', unsafe_allow_html=True)
    uploaded_file = st.file_uploader(
        "**Перетягніть файл сюди або натисніть для вибору**",
        type=["jpg", "png", "jpeg", "tiff", "bmp"],

```

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```

    help="Підтримуються зображення у форматах JPG, PNG, JPEG, TIFF, BMP",
    label_visibility="collapsed"
)
st.markdown("<p style='color: #666; font-size: 0.9rem; margin-top: 1rem;'>Максимальний розмір:
200MB</p>",
            unsafe_allow_html=True)
st.markdown('</div>', unsafe_allow_html=True)

with col2:
    st.markdown("""
<div style="padding: 1rem;">
    <h4>📄 Вимоги до зображення:</h4>
    <ul style="font-size: 0.9rem;">
        <li>Формати: JPG, PNG, JPEG</li>
        <li>Рекомендована роздільна здатність: 224x224+</li>
        <li>Колірний простір: RGB</li>
        <li>Оптимальна якість: 80%+</li>
    </ul>
</div>
""", unsafe_allow_html=True)

if uploaded_file:
    # Аналіз зображення
    with st.spinner("🔍 Аналіз зображення..."):
        col1, col2 = st.columns(2)

        with col1:
            st.markdown("### 📄 Вхідне зображення")
            image = Image.open(uploaded_file).convert("RGB")

            # Створюємо вкладки для різних представлень зображення
            tab1, tab2 = st.tabs(["Оригінал", "Оброблене"])

            with tab1:
                st.image(image, caption=f"Розмір: {image.size}", use_container_width=True)

            with tab2:
                # Мініатюра обробленого зображення
                transform_preview = transforms.Compose([
                    transforms.Resize((224, 224)),
                    transforms.ToTensor()
                ])

```

```

    ])
    preview_img = transform_preview(image)
    preview_img = preview_img.permute(1, 2, 0).numpy()
    st.image(preview_img, caption="Підготовлене для моделі (224x224)",
            use_container_width=True)

```

with col2:

```

    # Обробка та класифікація
    transform = transforms.Compose([
        transforms.Resize((224, 224)),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
    ])

```

```

x = transform(image).unsqueeze(0)

```

```

with torch.no_grad():

```

```

    outputs = model(x)
    probs = torch.softmax(outputs, dim=1)
    pred_idx = torch.argmax(probs, dim=1).item()
    pred_class = idx_to_class[pred_idx]
    confidence = probs[0][pred_idx].item()
    st.markdown("### 🏠 Результати аналізу")

```

```

    st.markdown('<div class="result-card fade-in">', unsafe_allow_html=True)

```

```

    if pred_class.lower() == 'malignant':

```

```

        status = "❌ ЗЛОЯКІСНЕ УТВОРЕННЯ"
        badge_class = "malignant"
        status_color = "#dc3545"

```

```

    else:

```

```

        status = "✅ ДОБРОЯКІСНЕ УТВОРЕННЯ"
        badge_class = "benign"
        status_color = "#28a745"

```

```

    st.markdown(f'<div class="class-badge {badge_class}">{status}</div>',
            unsafe_allow_html=True)

```

```

    st.markdown(f'***Впевненість моделі:***')

```

```

    st.progress(confidence)

```

```

    st.markdown(f'***{confidence*100:.2f}%***')

```

```

st.markdown("---")
st.markdown("***Ймовірності класів***")

classes = list(idx_to_class.values())
probabilities = probs[0].cpu().numpy()

for cls, prob in zip(classes, probabilities):
    col_a, col_b = st.columns([3, 2])
    with col_a:
        st.write(f"• {cls}:")
    with col_b:
        st.write(f"***{prob:.3f}*** ({prob*100:.1f}%)")

st.markdown('</div>', unsafe_allow_html=True)
fig, ax = plt.subplots(figsize=(10, 4))
colors = ['#28a745' if c.lower() == 'benign' else '#dc3545'
          for c in classes]
bars = ax.barh(classes, probabilities, color=colors)
ax.set_xlabel('Ймовірність')
ax.set_xlim(0, 1)
ax.set_title('Ймовірності класифікації', fontsize=14, fontweight='bold')

for bar, prob in zip(bars, probabilities):
    width = bar.get_width()
    ax.text(width + 0.01, bar.get_y() + bar.get_height()/2,
           f'{prob:.3f}', va='center', fontweight='bold')

plt.tight_layout()
st.pyplot(fig)

elif page == "📊 Аналіз результатів":
    st.markdown('<div class="main-header fade-in">', unsafe_allow_html=True)
    st.markdown("## 📊 Аналіз результатів навчання")
    st.markdown("Детальна статистика та метрики ефективності моделі")
    st.markdown('</div>', unsafe_allow_html=True)

if training_metrics and test_results:
    st.markdown("### 📊 Ключові метрики")

    col1, col2, col3, col4 = st.columns(4)

```



```

metrics_data = [
    ("AUC-ROC", test_results['auc_roc'], "#667eea"),
    ("Точність", test_results['accuracy'], "#10b981"),
    ("Чутливість", test_results['recall'], "#f59e0b"),
    ("Прецизійність", test_results['precision'], "#ef4444")
]

for col, (name, value, color) in zip([col1, col2, col3, col4], metrics_data):
    with col:
        st.markdown(
            f"""
            <div style="background-color: {color}20; padding: 1.5rem;
            border-radius: 10px; border-left: 4px solid {color};">
                <h4 style="margin: 0; color: {color};">{name}</h4>
                <h2 style="margin: 0.5rem 0; color: {color};">{value:.4f}</h2>
                <p style="margin: 0; font-size: 0.8rem; color: #666;">
                    {name} на тестовій вибірці
                </p>
            </div>
            """,
            unsafe_allow_html=True
        )

st.markdown("### 📈 Криві навчання")

tab1, tab2 = st.tabs(["Втрати", "Точність"])

with tab1:
    fig, ax = plt.subplots(figsize=(10, 6))
    ax.plot(training_metrics['train_loss'], label='Тренувальні втрати',
            linewidth=2, color='#667eea')
    ax.plot(training_metrics['val_loss'], label='Валідаційні втрати',
            linewidth=2, color='#ef4444')
    ax.fill_between(range(len(training_metrics['train_loss'])),
                    training_metrics['train_loss'], alpha=0.2, color='#667eea')
    ax.fill_between(range(len(training_metrics['val_loss'])),
                    training_metrics['val_loss'], alpha=0.2, color='#ef4444')
    ax.set_xlabel('Епоха', fontsize=12)
    ax.set_ylabel('Втрати', fontsize=12)
    ax.set_title('Динаміка втрат під час навчання', fontsize=14, fontweight='bold')
    ax.legend()

```

```
ax.grid(True, alpha=0.3)
st.pyplot(fig)
```

with tab2:

```
fig, ax = plt.subplots(figsize=(10, 6))
ax.plot(training_metrics['train_acc'], label='Тренувальна точність',
        linewidth=2, color='#10b981')
ax.plot(training_metrics['val_acc'], label='Валідаційна точність',
        linewidth=2, color='#f59e0b')
ax.fill_between(range(len(training_metrics['train_acc'])),
               training_metrics['train_acc'], alpha=0.2, color='#10b981')
ax.fill_between(range(len(training_metrics['val_acc'])),
               training_metrics['val_acc'], alpha=0.2, color='#f59e0b')
ax.set_xlabel('Епоха', fontsize=12)
ax.set_ylabel('Точність', fontsize=12)
ax.set_title('Динаміка точності під час навчання', fontsize=14, fontweight='bold')
ax.legend()
ax.grid(True, alpha=0.3)
st.pyplot(fig)
```

```
col1, col2 = st.columns([2, 1])
```

with col1:

```
st.markdown("### 🌀 Матриця плутанини")
cm = np.array(test_results['confusion_matrix'])
fig, ax = plt.subplots(figsize=(8, 6))

# Візуалізація heatmap
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', ax=ax,
            xticklabels=list(idx_to_class.keys()),
            yticklabels=list(idx_to_class.keys()),
            cbar_kws={'label': 'Кількість зразків'})

ax.set_title('Матриця плутанини на тестовій вибірці',
            fontsize=14, fontweight='bold', pad=20)
ax.set_xlabel('Прогнозований клас', fontsize=12)
ax.set_ylabel('Справжній клас', fontsize=12)

plt.tight_layout()
st.pyplot(fig)
```

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання
with col2:

```
st.markdown("### 📊 Статистика класифікації")

cm = test_results['confusion_matrix']
tn, fp, fn, tp = cm[0][0], cm[0][1], cm[1][0], cm[1][1]
total = tn + fp + fn + tp

stats = [
    (f"**{tp}**", "Правильно виявлені злоякісні", "#dc3545"),
    (f"**{tn}**", "Правильно виявлені доброякісні", "#28a745"),
    (f"**{fp}**", "Помилково виявлені злоякісні", "#ffc107"),
    (f"**{fn}**", "Пропущені злоякісні", "#fd7e14"),
]

for value, label, color in stats:
    st.markdown(
        f"""
        <div style="margin-bottom: 1rem; padding: 1rem;
        background-color: {color}20; border-radius: 8px;
        border-left: 4px solid {color};">
            <div style="font-size: 1.5rem; font-weight: bold; color: {color};">
                {value}
            </div>
            <div style="font-size: 0.9rem; color: #666;">
                {label}
            </div>
        </div>
        """,
        unsafe_allow_html=True
    )

st.markdown(f"""
**Загальна кількість зразків:** {total}
**Точність класифікації:** {(tp + tn) / total * 100:.1f}%
**Помилки класифікації:** {(fp + fn) / total * 100:.1f}%
""")
```

else:

```
st.warning("⚠️ Метрики навчання не знайдено. Запустіть навчання спочатку.")
```

```
elif page == "📊 Порівняльний аналіз":
```

Кафедра інтелектуальних інформаційних систем

Класифікація ракових клітин за гістологічними зображеннями на основі методів машинного навчання

```

st.markdown('<div class="main-header fade-in">', unsafe_allow_html=True)
st.markdown("## 📊 Порівняльний аналіз архітектур")
st.markdown("Порівняння різних моделей для класифікації гістологічних зображень")
st.markdown('</div>', unsafe_allow_html=True)
comparison_data = {
    'Архітектура': ['ResNet50', 'EfficientNet-B0', 'DenseNet121', 'VGG16', 'MobileNetV2'],
    'AUC-ROC': [0.921, 0.903, 0.915, 0.889, 0.897],
    'Точність': [0.893, 0.875, 0.885, 0.862, 0.871],
    'Чутливість': [0.865, 0.841, 0.852, 0.828, 0.836],
    'Прецизійність': [0.908, 0.892, 0.901, 0.878, 0.886],
    'Параметри (М)': [25.6, 5.3, 8.1, 138.4, 3.5],
    'Час інференсу (мс)': [15.2, 8.7, 12.4, 25.8, 6.3]
}

df = pd.DataFrame(comparison_data)

col1, col2 = st.columns([3, 1])

with col1:
    st.markdown("### 📄 Таблиця порівняння")

    sort_by = st.selectbox(
        "Сортувати за:",
        ['AUC-ROC', 'Точність', 'Чутливість', 'Прецизійність',
        'Параметри (М)', 'Час інференсу (мс)'],
        index=0
    )

    df_sorted = df.sort_values(by=sort_by, ascending=False)

    def highlight_max(s):
        is_max = s == s.max()
        return ['background-color: #d4edda' if v else " for v in is_max]

    styled_df = df_sorted.style.apply(highlight_max, subset=['AUC-ROC', 'Точність',
        'Чутливість', 'Прецизійність'])\
        .apply(lambda x: ['background-color: #f8d7da' if v == x.min() else "
        for v in x], subset=['Параметри (М)', 'Час інференсу (мс)'])\
        .format({
            'AUC-ROC': '{:.3f}',
            'Точність': '{:.3f}',

```

```

        "Чутливість": '{:.3f}',
        "Прецизійність": '{:.3f}',
        "Параметри (М)": '{:.1f}',
        "Час інференсу (мс)": '{:.1f}'
    })

st.dataframe(styled_df, use_container_width=True, height=300)

with col2:
    st.markdown("### ☒ Налаштування")

    metrics_to_compare = st.multiselect(
        "Оберіть метрики:",
        ['AUC-ROC', 'Точність', 'Чутливість', 'Прецизійність'],
        default=['AUC-ROC', 'Точність']
    )

    show_details = st.checkbox("Показати деталі", value=True)

if metrics_to_compare:
    st.markdown("### 📊 Візуальне порівняння")

    fig, axes = plt.subplots(1, len(metrics_to_compare), figsize=(5*len(metrics_to_compare), 6))

    if len(metrics_to_compare) == 1:
        axes = [axes]

    colors = plt.cm.Set3(np.linspace(0, 1, len(df)))

    for ax, metric in zip(axes, metrics_to_compare):
        bars = ax.bar(df['Архітектура'], df[metric], color=colors)
        ax.set_ylabel(metric)
        ax.set_title(f'Порівняння {metric}')
        ax.tick_params(axis='x', rotation=45)
        ax.grid(True, alpha=0.3, axis='y')

    for bar in bars:
        height = bar.get_height()
        ax.text(bar.get_x() + bar.get_width()/2., height + 0.01,
                f'{height:.3f}', ha='center', va='bottom', fontsize=9)

```

```
plt.tight_layout()
```

```
st.pyplot(fig)
```

```
if show_details:
```

```
    st.markdown("### 📌 Висновки")
```

```
    col1, col2, col3 = st.columns(3)
```

```
    with col1:
```

```
        st.markdown("""
```

```
        #### 📦 Найкращі результати
```

```
        - ResNet50: Найвища AUC-ROC (0.921)
```

```
        - EfficientNet-B0: Найшвидший інференс
```

```
        - DenseNet121: Найкращий баланс якості та швидкості
```

```
        """)
```

```
    with col2:
```

```
        st.markdown("""
```

```
        #### 📊 Компроміси
```

```
        - VGG16: Висока точність, але великий розмір
```

```
        - MobileNetV2: Компактна, але нижча точність
```

```
        - ResNet50: Оптимальний вибір для більшості завдань
```

```
        """)
```

```
    with col3:
```

```
        st.markdown("""
```

```
        #### 📌 Рекомендації
```

```
        1. Для максимальної точності: ResNet50
```

```
        2. Для мобільних рішень: MobileNetV2
```

```
        3. Для балансу: EfficientNet-B0
```

```
        4. Для досліджень: DenseNet121
```

```
        """)
```