

## ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

# Метод обробки зображень з використанням машинного навчання

## Кваліфікаційна робота

Виконав:  
студент гр. СПМ-23-5  
Ровенчак В.М.

Керівник:  
доц. каф. ЕОМ, к.т.н.,  
Федорченко В.М.

## Мета роботи та завдання

2

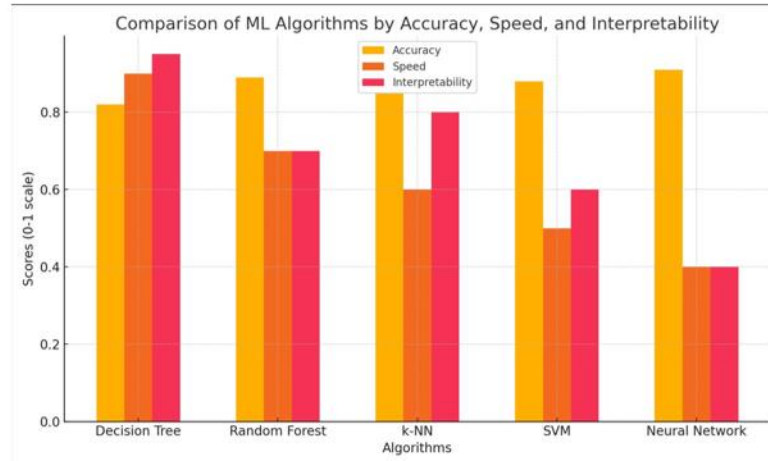
**Метою кваліфікаційної роботи** є дослідження існуючих методів та реалізація ефективного методу обробки зображень, який дозволяє з високою точністю розпізнавати ознаки пневмонії на рентгенівських знімках.

**Об'єктом дослідження** є процес обробки медичних зображень із метою виявлення ознак патологій.

### **Завдання:**

- провести аналіз сучасних методів обробки зображень та алгоритмів машинного навчання, що застосовуються для медичної діагностики;
- дослідити архітектури глибоких нейронних мереж з метою оцінки їхньої придатності для виявлення пневмонії на рентгенівських знімках;
- обрати відповідний відкритий набір медичних зображень та здійснити попередню обробку даних;
- розробити та реалізувати модель глибокого навчання для класифікації знімків на наявність або відсутність пневмонії;
- провести навчання та тестування моделі, з використанням стандартних метрик точності, чутливості, специфічності та AUC;
- здійснити візуалізацію результатів роботи моделі, включаючи правильні та хибні класифікації;
- оцінити ефективність розробленого методу та порівняти його з іншими підходами.

## Порівняльний аналіз методів машинного навчання 3



## Огляд архітектур та інструментів реалізації 4

Архітектура	Призначення	Ключові особливості
CNN	Обробка зображень, відео, просторових даних	Згорткові фільтри, локальність, ієрархічне вилучення ознак
RNN	Обробка послідовностей, часових рядів, мови	Зворотні зв'язки, пам'ять про попередні стани
GAN	Генерація нових даних, зокрема зображень	Архітектура з двома мережами: генератор і дискримінатор
Архітектура	Переваги	Недоліки
CNN	Висока точність при класифікації зображень	Не пристосована до обробки послідовностей
RNN	Добре працює з часовими залежностями	Складність тренування, проблема з градієнтами
GAN	Здатність створювати реалістичні нові дані	Важке налаштування, нестабільність під час навчання

Фреймворк	Основне призначення	Сумісність з Google Colab
TensorFlow	Глибоке навчання, розробка нейронних мереж	Повна інтеграція, попередньо встановлений
PyTorch	Глибоке навчання, наукові дослідження та візуалізація	Повна інтеграція, попередньо встановлений
OpenCV	Обробка зображень, комп'ютерний зір	Повна інтеграція, попередньо встановлений
Фреймворк	Переваги	Недоліки
TensorFlow	Підтримка мобільних та веб-платформ, TPU	Складніша відлагодження, громіздкий синтаксис
PyTorch	Гнучкість, зручність для дослідників, динамічне обчислення	Менш зручний у виробничих середовищах
OpenCV	Швидкість, простота базових операцій зображень	Не призначений для глибокого навчання

## Метод обробки медичних зображень



## Метод обробки медичних зображень

### Крок 1. Вибір і підготовка даних:

- використовується датасет ChestMNIST із колекції MedMNIST, що містить зображення у форматі 28×28 пікселів;
- дані поділяються на тренувальний, валідаційний і тестовий набори;
- здійснюється нормалізація піксельних значень, перетворення до тензорів, а також, за потреби, балансування класів.

### Крок 2. Побудова базової нейронної моделі:

- розробляється згортовка нейронна мережа (CNN) з кількома шарами Conv2D, MaxPooling, Dropout і Dense;
- застосовуються функції активації ReLU, а на вихідному шарі – Softmax (для бінарної класифікації – sigmoid);
- компіляція моделі з використанням оптимізатора Adam, функції втрат binary crossentropy, та метрик точності.

### Крок 3. Розширення моделі на основі ResNet:

- імплементується варіант ResNet (наприклад, ResNet18) з попередньо навченими вагами або без них;
- архітектура адаптується до розмірів вхідних зображень і кількості вихідних класів;
- забезпечується глибше вилучення ознак без втрати градієнтів (residual connections).

### Крок 4. Навчання моделі:

- навчання проводиться в кілька епох з використанням збереження найкращої моделі (ModelCheckpoint) та контролю за втратою на валідаційному наборі (EarlyStopping);
- ведеться збір історії точності й втрат для подальшої візуалізації.

### Крок 5. Оцінка результатів:

- обчислюються точність, recall, precision, F1-score;
- будується матриця неточностей, ROC-крива, PR-крива, а також обчислюється AUC;
- візуалізуються правильні та хибні передбачення моделі на прикладах зображень.

### Крок 6. Інтерпретація та висновки:

- аналізується поведінка моделі, виявляються сильні сторони та потенційні помилки.

# Структура згорткової нейронної мережі. ResNet.

7

Model: "sequential"

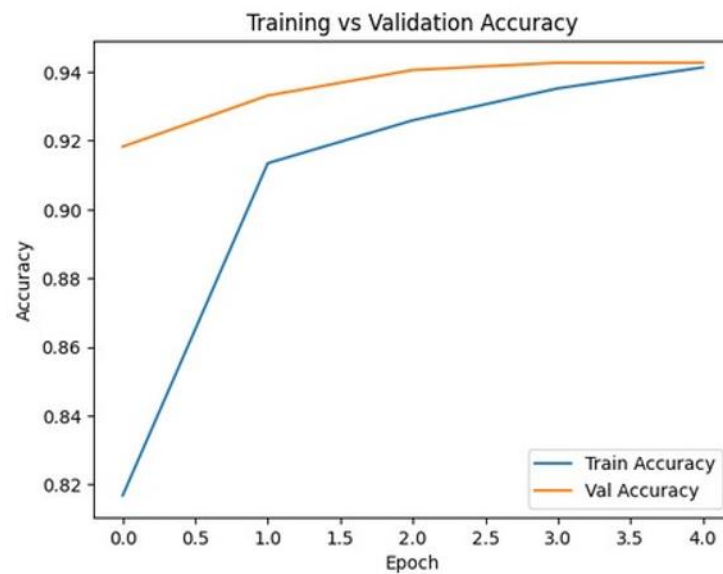
Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
conv2d_1 (Conv2D)	(None, 11, 11, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 5, 5, 64)	0
flatten (Flatten)	(None, 1600)	0
dense (Dense)	(None, 64)	102,464
dropout (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 1)	65

Total params: 121,345 (474.00 KB)  
 Trainable params: 121,345 (474.00 KB)  
 Non-trainable params: 0 (0.00 B)

Layer (type)	Output Shape	Param #	Connected to
input_layer_1 (InputLayer)	(None, 224, 224, 3)	0	-
conv1_pad (ZeroPadding2D)	(None, 238, 238, 3)	0	input_layer_1[0]-
conv1_conv (Conv2D)	(None, 112, 112, 64)	9,472	conv1_pad[0][0]
conv1_bn (BatchNormalizati...	(None, 112, 112, 64)	256	conv1_conv[0][0]
conv1_relu (Activation)	(None, 112, 112, 64)	0	conv1_bn[0][0]
pool1_pad (ZeroPadding2D)	(None, 114, 114, 64)	0	conv1_relu[0][0]
pool1_pool (MaxPooling2D)	(None, 56, 56, 64)	0	pool1_pad[0][0]
conv2_block1_1_conv (Conv2D)	(None, 56, 56, 64)	4,160	pool1_pool[0][0]
conv2_block1_1_bn (BatchNormalizati...	(None, 56, 56, 64)	256	conv2_block1_1_c-
conv2_block1_1_relu (Activation)	(None, 56, 56, 64)	0	conv2_block1_1_b-
conv2_block1_2_conv (Conv2D)	(None, 56, 56, 64)	36,928	conv2_block1_1_r-
conv2_block1_2_bn (BatchNormalizati...	(None, 56, 56, 64)	256	conv2_block1_2_c-
conv2_block1_2_relu (Activation)	(None, 56, 56, 64)	0	conv2_block1_2_b-
conv2_block1_0_conv (Conv2D)	(None, 56, 56, 256)	16,640	pool1_pool[0][0]
conv2_block1_3_conv (Conv2D)	(None, 56, 56, 256)	16,640	conv2_block1_2_r-
conv2_block1_0_bn (BatchNormalizati...	(None, 56, 56, 256)	1,024	conv2_block1_0_c-
conv2_block1_3_bn (BatchNormalizati...	(None, 56, 56, 256)	1,024	conv2_block1_3_c-
conv2_block1_add (Add)	(None, 56, 56, 256)	0	conv2_block1_0_b- conv2_block1_3_b-
conv2_block1_out	(None, 56, 56, 256)	0	conv2_block1_add

## Зміна точності моделі під час навчання і валідації

8



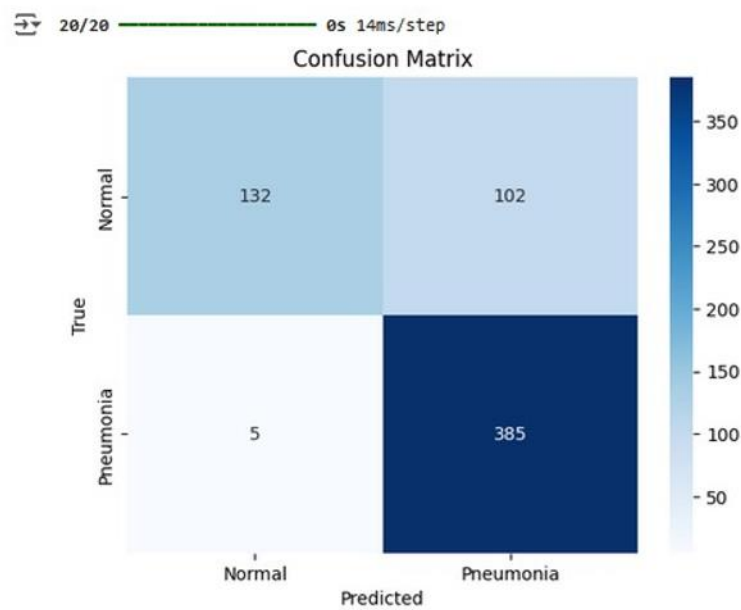
## Зміна функції втрат на тренувальному та валідаційному наборах даних

9

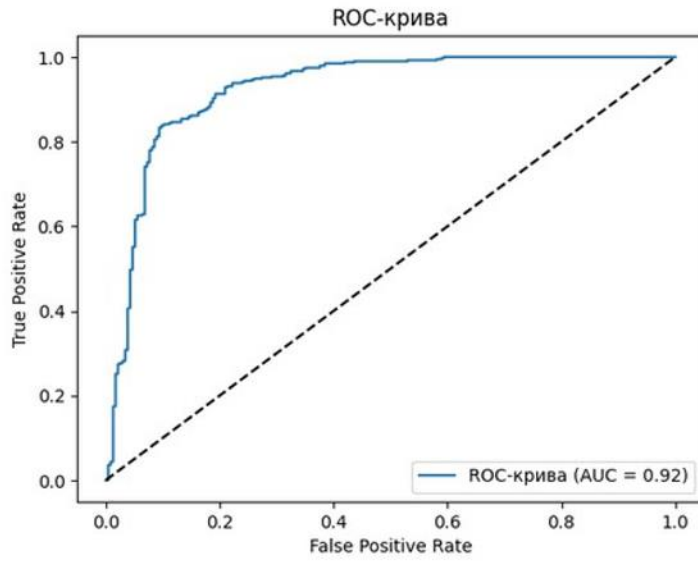


## Матриця плутанини

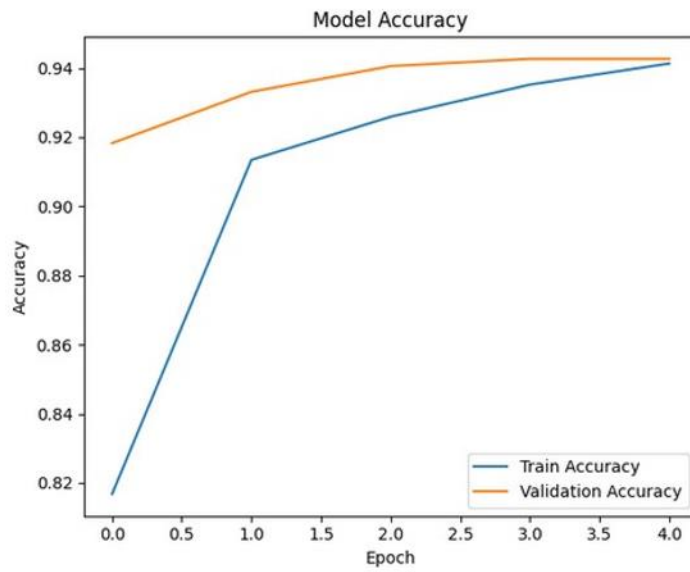
10



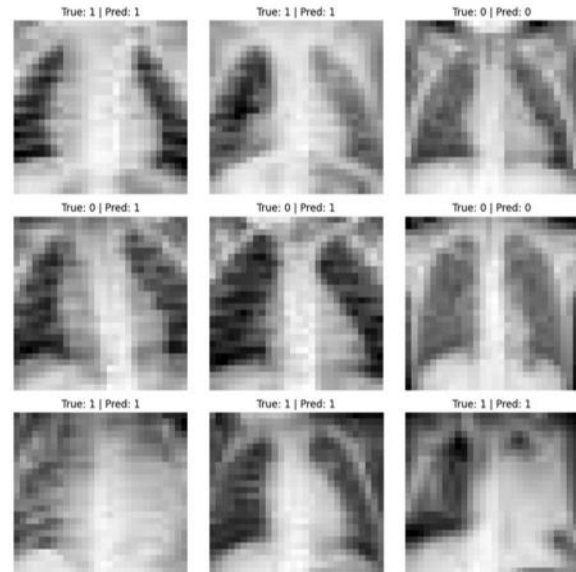
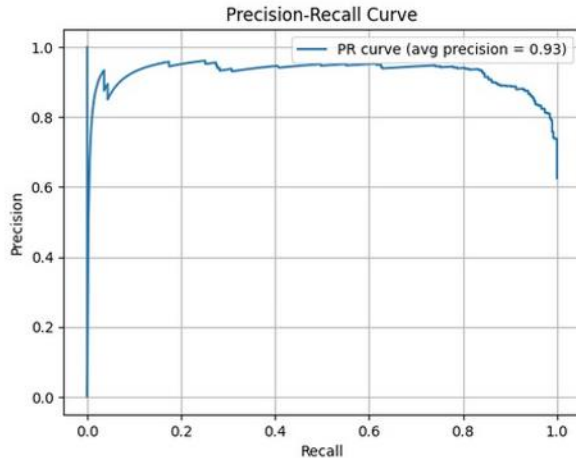
# ROC крива



# Точність моделі



# Precision-Recall. Візуалізація результатів класифікації



## Апробація результатів

<sup>1</sup>Kharkiv National University of Radio Electronics, Kharkiv, Ukraine

### METHODS OF DATA MINING USING MACHINE LEARNING

**Abstract.** Relevance. In the context of the continuous growth of information volumes across various fields of the economy, science, and technology, the problem of effective processing and analysis of large-scale data has become increasingly urgent. Traditional analytical methods are no longer capable of providing fast and accurate extraction of useful knowledge and patterns from massive information flows. The response to this challenge lies in the methods of data mining, which are based on modern machine learning technologies. These methods enable the automatic discovery of hidden patterns, the generation of accurate predictions, and the support of data-driven decision-making in real time. Given the rapid development of digitalization, artificial intelligence, and the need for prompt decision-making in a competitive environment, the relevance of developing and improving data mining methods is growing steadily. The object of research is process of data mining using machine learning methods, namely the set of algorithms, models, tools, and approaches that ensure the detection of hidden patterns, anomalies, and structures in large volumes of heterogeneous information. Purpose of the article. This study explores contemporary approaches to intelligent data analysis based on machine learning techniques and assesses their effectiveness across a range of application domains. The article aims to provide a structured overview of state-of-the-art algorithms and to evaluate their respective advantages and limitations in processing large-scale and high-dimensional datasets. Research results. A systematic analysis of key data mining methods based on machine learning algorithms was carried out. It was found that the most effective approaches for processing large and heterogeneous datasets include classification, clustering, regression analysis, and dimensionality reduction techniques. Deep neural networks demonstrated effectiveness when applied to unstructured data such as text, images, and time series. The study revealed that the appropriate choice of algorithm depends not only on the data type but also on the specific nature of the task. A comparative assessment of tools showed that the Python ecosystem offers the greatest flexibility, while AutoML platforms simplify model deployment for users with limited programming experience. The research also included a review of recent publications that confirm the practical value of machine learning in real-world use cases. Overall, the findings indicate that machine learning is a driving force behind the evolution of data mining methods, enabling accurate, scalable, and adaptive data processing in the context of modern digital transformation. Conclusions. Machine learning has significantly expanded the capabilities of intelligent data analysis by enabling the automatic detection of patterns, forecasting, and decision-making based on large volumes of information. The study demonstrates the effectiveness of various algorithms in tasks such as classification, clustering, regression, and deep learning. Python-based tools and cloud platforms have been identified as the most convenient environments for implementing analytical models. A promising direction lies in the development of explainable AI and hybrid approaches that combine algorithmic precision with domain-specific expertise. Keywords: intelligent data analysis, machine learning, classification, clustering, regression, deep learning, neural networks, dimensionality reduction, analytical tools, Data Mining, CRISP-DM, AutoML, Big Data, Python.

#### Introduction

Intelligent data analysis (Data Mining) [1] is a significant area within modern information technologies, aimed at the automated extraction of meaningful patterns, knowledge, and deep insights from large volumes of both structured and unstructured data. In today's world, data is generated at an unprecedented rate – from online platforms and social networks to financial transactions, medical records, and industrial sensors. In this context of information overload, traditional analytical methods are losing their effectiveness and are increasingly being replaced by intelligent approaches. Machine learning plays a central role as the driving force behind intelligent data analysis. Its ability to

of digital data, the need for more complex, self-learning algorithms has become apparent. Machine learning has evolved from simple models into advanced deep neural networks capable of performing multi-level information processing with high accuracy.

Unlike traditional statistical methods, which are oriented toward hypothesis testing and formal modeling, machine learning can work with large-scale datasets without the need for predefined assumptions. It can identify patterns within chaotic, heterogeneous, and high-dimensional data – a crucial feature in the era of Big Data. Moreover, intelligent data analysis based on machine learning is inherently interdisciplinary, integrating knowledge from computer science, statistics, mathematics, linguistics, medicine, economics, psy-

machine learning has become highly relevant, as evidenced by the growing number of specialized reviews and systematic studies across various fields. This section highlights the most influential publications in finance, education, healthcare, and multimedia, as well as methodological papers focused on imbalanced data processing and open science initiatives.

A significant contribution in this area is presented in [2], a comprehensive review on fraud detection methods based on data mining and machine learning techniques. The paper covers multiple applied domains, such as banking, insurance, telecommunications, and e-commerce. The authors emphasize that traditional fraud detection algorithms, despite their past effectiveness, are increasingly incapable of addressing the challenges posed by today's high-speed and dynamic environments. The review proposes a generalized classification of approaches, including both classical techniques and modern ensemble or hybrid architectures that combine multiple models to improve accuracy when working with imbalanced datasets. Special attention is paid to model interpretability, which is critical in financial sectors where decision-making must be transparent and accountable. In this context, the researchers highlight the role of explainable AI as a direction capable of integrating the computational power of modern algorithms with the need for human oversight and auditing. The paper also stresses the growing importance of adaptive and context-aware models capable of operating with limited, fragmented, or unreliable data in the era of increasingly complex fraud schemes.

Another influential work is the updated systematic review [3], focused on the use of machine learning in education. The authors trace the evolution of Educational Data Mining and Learning Analytics over the past decade, analyzing both the growth in publication metrics and the methodological maturity of the field. The paper presents educational analytics as an emerging scientific discipline that integrates pedagogy, artificial intelligence, psychology, sociology, and statistics. It explores the technological pipeline for knowledge discovery in educational data – from data collection and preprocessing to interpretation and practical implementation. Numerous examples are provided, including the use of classification models for learning outcome prediction, learning style analysis, and recommendation systems in MOOCs. The paper also examines the appli-

per evaluates the performance of traditional statistical methods and modern machine learning algorithms in identifying anomalies. It covers supervised, unsupervised, and deep learning methods, assessing their ability to handle high-dimensional financial data and detect complex fraud patterns. Ensemble models are found to offer a strong balance between detection accuracy and interpretability. However, challenges remain, particularly regarding class imbalance and model generalization. The study proposes hybrid approaches that combine domain knowledge with data-driven analysis and emphasizes the potential of explainable AI to enhance transparency and trust in automated financial anomaly detection systems. The results significantly expand the understanding of machine learning in financial analytics and suggest promising directions for improving anomaly detection efficiency and reliability.

A notable interdisciplinary example is a study dedicated to the use of machine learning in cardiology. The publication [5] presents an extensive meta-analysis of forty studies focused on automated prediction of coronary artery disease using clinical and demographic data. Ensemble algorithms – combining logistic regression, decision trees, gradient boosting, and neural networks – emerge as the most effective models. The study emphasizes evaluation using medically relevant metrics such as sensitivity, specificity, and accuracy, highlighting the importance of not only statistical performance but also the applicability of models within medical systems, where false positives or negatives can have serious consequences. The authors draw attention to the limitations of publicly available medical datasets and recommend local model validation across demographic groups. Interpretable algorithms and hybrid systems that integrate medical expertise with the flexibility of machine learning are proposed as future directions.

In [6], the role of data mining methods – particularly text mining – in digital forensics is examined. The authors emphasize the growing complexity of digital sources and devices involved in investigations, which renders manual analysis ineffective. Machine learning models and pattern recognition algorithms are shown to be essential for uncovering hidden digital evidence that traditional methods might overlook. The paper demonstrates that text analysis techniques, as a subset of data mining, are critical to improving the precision and efficiency of digital forensic workflows. The

D. Diachenko, M. Prokorchuk, V. Rovenchak, A. Frolov. Methods of data mining using machine learning. Системи управління, навігації та зв'язку, вип.4. Полтава, 2025.

## Висновки

15

Було розроблено, реалізовано та апробовано метод обробки медичних зображень на основі глибокого машинного навчання, зокрема із застосуванням згорткових нейронних мереж. Метою роботи було вивчення можливостей автоматизованого аналізу зображень органів грудної клітки для виявлення патологій (зокрема, пневмонії) з використанням сучасних нейронних архітектур.

У теоретичній частині було проведено систематизований огляд методів обробки зображень, що охоплював як класичні підходи (фільтрація, контурний аналіз, сегментація), так і сучасні алгоритми на основі машинного навчання. Особливу увагу приділено принципам роботи глибоких згорткових мереж, їх архітектурам (CNN, ResNet, U-Net) та етапам підготовки даних для навчання моделей. У контексті медичних застосувань було розглянуто особливості класифікації, сегментації та пояснення рішень моделей.

Практична частина дослідження включала реалізацію класифікаційної моделі для виявлення пневмонії на основі набору зображень ChestMNIST із колекції MedMNIST. Було побудовано базову згорткову нейромережу, а також апробовано модифікацію з використанням ResNet-архітектури, що дозволило досягти високих значень точності (понад 94%) та чутливості. Було реалізовано повний цикл: від завантаження й попередньої обробки даних до оцінки результатів за допомогою матриці неточностей, ROC-кривої та PR-кривої. Окрема увага була присвячена візуалізації результатів і поясненню поведінки моделі на прикладі конкретних зображень.

Отримані результати підтвердили гіпотезу щодо ефективності використання глибоких згорткових мереж для класифікації медичних зображень. Запропонований підхід значно перевершив класичні методи машинного навчання як за точністю, так і за адаптивністю до складних, неоднорідних даних. Порівняння з аналогами продемонструвало перевагу розробленої моделі як за якістю класифікації, так і за її узагальнюючою здатністю.

## ДОДАТОК Б

### Програмний код

#### Б.1 Встановлення бібліотек, завантаження MedMNIST

```

import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D,
Flatten, Dense, Dropout
from sklearn.metrics import classification_report,
confusion_matrix
import seaborn as sns

!pip install -q medmnist
from medmnist import INFO
from medmnist import PneumoniaMNIST
import torchvision.transforms as transforms

# Налаштування
data_flag = 'pneumoniamnist'
info = INFO[data_flag]
DataClass = getattr(__import__('medmnist'),
info['python_class'])

# Завантаження
train_dataset = DataClass(split='train',
transform=transforms.ToTensor(), download=True)
test_dataset = DataClass(split='test',
transform=transforms.ToTensor(), download=True)

X_train = train_dataset.imgs / 255.0
y_train = train_dataset.labels.flatten()
X_test = test_dataset.imgs / 255.0
y_test = test_dataset.labels.flatten()

X_train = X_train[... , np.newaxis]
X_test = X_test[... , np.newaxis]

```

#### Б.2 Побудова моделі CNN, навчання моделі, візуалізація результатів

##### навчання

```

model = Sequential([
    Conv2D(32, (3,3), activation='relu', input_shape=(28, 28,

```

```

1)),
    MaxPooling2D(2,2),
    Conv2D(64, (3,3), activation='relu'),
    MaxPooling2D(2,2),
    Flatten(),
    Dense(64, activation='relu'),
    Dropout(0.3),
    Dense(1, activation='sigmoid')
])

model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])

model.summary()
history = model.fit(X_train, y_train, epochs=5,
                    validation_split=0.2)
plt.plot(history.history['accuracy'], label='Train Accuracy')
plt.plot(history.history['val_accuracy'], label='Val Accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.title('Training vs Validation Accuracy')
plt.legend()
plt.show()

plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Val Loss')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.title('Training vs Validation Loss')
plt.legend()
plt.show()

```

### Б.3 Оцінка моделі, ROC-крива, класифікаційний звіт, приклади класифікації зображень

```

plt.plot(history.history['accuracy'], label='Train Accuracy')
plt.plot(history.history['val_accuracy'], label='Val Accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.title('Training vs Validation Accuracy')
plt.legend()
plt.show()

plt.plot(history.history['loss'], label='Train Loss')
plt.plot(history.history['val_loss'], label='Val Loss')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.title('Training vs Validation Loss')
plt.legend()
plt.show()

```

```

loss, acc = model.evaluate(X_test, y_test)
print(f"Test Accuracy: {acc*100:.2f}%")
y_pred = model.predict(X_test) > 0.5
cm = confusion_matrix(y_test, y_pred)

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
xticklabels=['Normal', 'Pneumonia'], yticklabels=['Normal',
'Pneumonia'])
plt.xlabel('Predicted')
plt.ylabel('True')
plt.title('Confusion Matrix')
plt.show()
from sklearn.metrics import roc_curve, auc

fpr, tpr, _ = roc_curve(y_test, model.predict(X_test))
roc_auc = auc(fpr, tpr)

plt.figure()
plt.plot(fpr, tpr, label=f'ROC-крива (AUC = {roc_auc:.2f})')
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC-крива')
plt.legend(loc="lower right")
plt.show()
print(classification_report(y_test, y_pred,
target_names=['Normal', 'Pneumonia']))
import random

indices = random.sample(range(len(X_test)), 9)
plt.figure(figsize=(10, 10))
for i, idx in enumerate(indices):
    plt.subplot(3, 3, i + 1)
    plt.imshow(X_test[idx].squeeze(), cmap='gray')
    plt.title(f"True: {y_test[idx]} | Pred: {int(y_pred[idx])}")
    plt.axis('off')
plt.tight_layout()
plt.show()

lt.show()

```