

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

Штучний інтелект

Методичні вказівки до виконання лабораторних робіт для студентів денної та заочної форми навчання другого рівня вищої освіти, галузі Інформаційні технології.

ЗАТВЕРДЖЕНО

на засіданні кафедри кібербезпеки та програмного забезпечення, протокол № 1 від 26.08.2025 року

Кропивницький
2025

УДК 004.8

Штучний інтелект: Методичні вказівки до виконання лабораторних робіт для студентів денної та заочної форми навчання другого рівня вищої освіти, галузі Інформаційні технології. / М-во освіти і науки України, Центральноукр. нац. техн. ун-т; [уклад. Є.В. Мелешко, О.М. Дреєв, Р.О. Ткачук] – Кропивницький: ЦНТУ, 2025. – 49 с.

Укладачі: **Мелешко Є.В.** доцент кафедри кібербезпеки та програмного забезпечення, доктор техн. наук, доцент;
Дреєв О.М. доцент кафедри кібербезпеки та програмного забезпечення, кандидат технічних наук;
Ткачук Р.О. Senior DevOps Engineer EPAM Systems / асистент кафедри кібербезпеки та програмного забезпечення

Рецензенти: Смірнов О. А., докт. техн. наук, професор, завідувач кафедри;
Якименко Н.М., к. ф.-м. наук, доцент.

© Центральноукраїнський
національний технічний
університет, 2025

ЗМІСТ

ВСТУП.....	4
Лабораторна робота №1.....	8
Лабораторна робота №2.	10
Лабораторна робота №3.	12
Лабораторна робота №4.....	14
Лабораторна робота №5.....	16
Лабораторна робота №6.....	18
Лабораторна робота №7.....	20
Лабораторна робота №8.....	22
Лабораторна робота №9.....	24
Лабораторна робота №10.	26
Лабораторна робота №11	28
Лабораторна робота №12.....	30
Лабораторна робота №13.....	32
Лабораторна робота №14.....	34
Лабораторна робота №15.....	36
Лабораторна робота №16.....	38
Список використаної літератури.....	45

ВСТУП

Навчальна дисципліна «Штучний інтелект» призначена для формування у здобувачів другого рівня вищої освіти галузі знань «Інформаційні технології» системи теоретичних знань та практичних навичок, необхідних для розробки, оптимізації та впровадження моделей машинного навчання на пристроях обмеженої обчислювальної потужності. Курс орієнтований на повний цикл створення Edge AI-рішення: від збору даних із фізичних сенсорів і камер до розгортання та підтримки моделі безпосередньо на мікроконтролерах і одноплатних комп'ютерах.

здобувачів здатності проєктувати та програмно реалізовувати інтелектуальні системи як складові комп'ютерних систем і мереж на різних програмних платформах із застосуванням сучасних методів, мов програмування та засобів автоматизації проєктування. Дисципліна забезпечує опанування принципів побудови архітектури системного і прикладного програмного забезпечення, уміння аналізувати проблематику, формулювати задачі та обирати ефективні методи їх розв'язання з урахуванням міждисциплінарних аспектів і сучасних наукових здобутків.

Окремий акцент робиться на розробленні апаратно програмних рішень інтелектуальних систем, обґрунтуванні вибору архітектури та способів обчислювального прискорення з урахуванням ресурсних і енергетичних обмежень. Результати навчання передбачають навички пошуку й критичного оцінювання інформації та здатність чітко представляти результати розробок і досліджень у звітах, статтях і презентаціях.

Метою вивчення навчальної дисципліни є формування системного розуміння сучасних підходів до розв'язання складних завдань у галузі комп'ютерної інженерії та поглиблення підготовки здобувачів у сфері проєктування, розроблення й дослідження програмних і апаратно-програмних

систем. Дисципліна орієнтована на поєднання теоретичних засад із практичними аспектами створення сучасних комп'ютерних рішень.

У межах курсу розглядаються питання алгоритмізації, програмування, побудови архітектури комп'ютерних систем і мереж, створення системного та прикладного програмного забезпечення, а також принципи інтеграції програмних і апаратних компонентів інтелектуальних систем. Значна увага приділяється використанню сучасних наукових досягнень, інженерних методів і засобів автоматизації проектування, що забезпечують ефективну реалізацію технічних рішень у професійній діяльності.

Навчальний курс спрямований на розвиток аналітичного та дослідницького підходу, здатності здійснювати науковий пошук, критично оцінювати інформацію та приймати обґрунтовані інженерні рішення. Розглядаються аспекти реалізації проєктів з урахуванням технічних, соціальних, економічних і правових чинників, а також особливостей функціонування сучасних комп'ютерних архітектур.

У результаті опанування дисципліни здобувачі набувають узагальнених знань щодо застосування математичних, природничих та інженерних методів у комп'ютерній інженерії, отримують досвід розроблення програмних і апаратно-програмних рішень, зокрема інтелектуальних систем, із врахуванням ресурсних, енергетичних та обчислювальних обмежень. Окрему увагу приділено питанням представлення результатів професійної та дослідницької діяльності у зрозумілій і структурованій формі для різних аудиторій.

Супровідною формою навчання є лабораторні роботи, які формують практичні компетентності у сфері Edge AI.

Лабораторні роботи виконуються у такій послідовності:

- вивчення теоретичного матеріалу за темою роботи;

- підготовка апаратної та програмної складової;
- реалізація алгоритмів навчання або інференсу;
- тестування моделі в реальних умовах;
- аналіз продуктивності та якості роботи системи;
- оформлення та захист звіту.

По завершенні кожної лабораторної роботи студент готує звіт, який повинен містити:

- тему та мету роботи;
- опис апаратної конфігурації;
- постановку задачі та обґрунтування вибору методів;
- фрагменти програмного коду;
- результати тестування та метрики якості;
- аналіз продуктивності та висновки.

Оцінювання результатів виконання лабораторних робіт здійснюється з урахуванням рівня теоретичної підготовки, коректності реалізації алгоритмів, ефективності оптимізації, стабільності роботи на цільовому пристрої та якості оформлення звіту.

ЛАБОРАТОРНА РОБОТА №1

ТЕМА: ОЗНАЙОМЛЕННЯ З АПАРАТНОЮ БАЗОЮ ТА СЕРЕДОВИЩАМИ РОЗРОБКИ

МЕТА: Налаштувати робоче середовище та перевірити доступні інструменти для Edge AI.

ЗНАТИ: Базові інструменти Python/Jupyter, принципи роботи з VSCode, основи розгортання проєктів на Raspberry Pi та ESP32.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- Jupyter Notebook документація <https://jupyter.org/documentation>
- VS Code: Python та Jupyter <https://code.visualstudio.com/docs/python/python-tutorial>

Додаткова документація:

- <https://platformio.org/install/ide?install=vscode>
- <https://www.raspberrypi.com/documentation/>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується з використанням Python, Jupyter, VSCode та доступних Edge-пристроїв. Доступ дозволено локально, через віддалений Jupyter або Google Colab (за погодженням з викладачем).

ЗАВДАННЯ

Необхідно налаштувати робоче середовище та підтвердити готовність інструментів.

Підготовка до виконання

1. Перевірити доступ до локального або віддаленого Jupyter (в локальній мережі або Google Colab).
2. Налаштувати Python-середовище у VSCode та перевірити запуск ноутбука.

Перевірка інструментів

3. Створити тестовий ноутбук у Jupyter та виконати прості обчислення (NumPy, matplotlib).
4. Перевірити доступ до Raspberry Pi (SSH або локально) та виконати базові команди.
5. Ініціалізувати проєкт у PlatformIO для ESP32 та зібрати приклад (Hello World або blink).

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Скриншоти/лог виконання, підтвердження доступу до кожного засобу та короткі висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які інструменти потрібні для виконання лабораторних робіт з Edge AI?
2. У чому різниця між локальним Jupyter та Google Colab?
3. Як перевірити коректність налаштування Python у VSCode?
4. Які переваги використання PlatformIO для ESP32?
5. Як організувати доступ до Raspberry Pi у локальній мережі?
6. Що таке середовище виконання (kernel) у Jupyter?
7. Які базові пакети Python потрібні для ML?
8. Як зберігати результати лабораторних робіт для звіту?

ЛАБОРАТОРНА РОБОТА №2

ТЕМА: ЗБІР ДАНИХ З ESP32 ТА OPENML CAMERA

МЕТА: Отримати базовий датасет для подальшого навчання ML-моделей.

ЗНАТИ: Основи зчитування сенсорних даних, формати збереження та організацію датасетів.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- ESP32 документація
<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/>
- OpenML: Datasets <https://www.openml.org/search?type=data>

Додаткова документація:

- <https://platformio.org/>
- <https://pandas.pydata.org/docs/>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується з використанням ESP32 або доступних датчиків, Python та Jupyter для збереження даних.

ЗАВДАННЯ

Необхідно зібрати первинні дані (зображення або сенсорні сигнали) та підготувати структуру датасету.

Підготовка до виконання

1. Обрати тип даних: зображення, аудіо або сенсорні сигнали.
2. Налаштувати зчитування даних з ESP32 або іншого джерела.

Збір та збереження

3. Зібрати мінімум 3 класи даних або 3 режими сигналів.
4. Зберегти дані у структурованому вигляді (папки або CSV/JSON).
5. Описати метадані: частота, формат, умови збору.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис процедури збору, структуру датасету, приклади даних і висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які основні кроки збору даних для ML?
2. Як забезпечити репрезентативність датасету?
3. Чому важливо фіксувати метадані збору?
4. Які формати даних зручні для подальшого навчання?
5. Як уникнути зміщення в даних при зборі?
6. Що таке data leakage на етапі збору?
7. Як оцінити якість зібраних даних?
8. Які типові помилки під час збору даних?

ЛАБОРАТОРНА РОБОТА №3

ТЕМА: ПІДГОТОВКА ТА РОЗМІТКА ДАНИХ У PYTHON

МЕТА: Сформувати навчальну, валідаційну та тестову вибірки.

ЗНАТИ: Методи очищення, нормалізації, розбиття даних, базову візуалізацію.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- Pandas: User Guide https://pandas.pydata.org/docs/user_guide/index.html
- Scikit-learn: Model Selection https://scikit-learn.org/stable/modules/cross_validation.html

Додаткова документація:

- <https://numpy.org/doc/>
- <https://matplotlib.org/stable/users/index.html>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується у Python/Jupyter (локально або Google Colab).

ЗАВДАННЯ

Необхідно очистити та підготувати дані для навчання.

Підготовка до виконання

1. Завантажити зібраний датасет у Python.
2. Перевірити пропуски та аномалії.

Обробка та розмітка

3. Виконати нормалізацію або масштабування.

4. Розмітити дані (labels) або підтвердити наявну розмітку.
5. Виконати розбиття на train/val/test з фіксованим seed.

Візуалізація

6. Побудувати базові графіки/приклади даних для перевірки якості.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис очищення, параметри розбиття, приклади даних та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Навіщо потрібні train/val/test набори?
2. Які методи нормалізації найчастіше застосовують?
3. Чому важливо фіксувати random seed?
4. Які ризики data leakage на етапі підготовки?
5. Як оцінити якість розмітки?
6. Що робити з дисбалансом класів?
7. Як візуалізація допомагає в перевірці даних?
8. Які типові помилки при розбитті даних?

ЛАБОРАТОРНА РОБОТА №4

ТЕМА: НАВЧАННЯ БАЗОВОЇ МОДЕЛІ НА СЕРВЕРІ

МЕТА: Отримати працюючу ML-модель та зафіксувати базові метрики.

ЗНАТИ: Основи навчання моделей у TensorFlow, вибір архітектури, контроль якості.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- TensorFlow: Getting Started <https://www.tensorflow.org/learn>
- Keras: Model Training https://keras.io/guides/training_with_built_in_methods/

Додаткова документація:

– <https://www.tensorflow.org/lite>

– https://scikit-learn.org/stable/modules/model_evaluation.html

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується у Python/Jupyter на локальному сервері або Google Colab.

ЗАВДАННЯ

Необхідно навчити базову модель (MLP/CNN) та зберегти її артефакти.

Підготовка до виконання

1. Обрати архітектуру під тип даних (MLP або CNN).
2. Підготувати дані з попередньої лабораторної роботи.

Навчання

3. Налаштувати цикл навчання (epochs, batch size).
4. Зафіксувати метрики навчання та валідації.
5. Зберегти модель у форматі SavedModel або H5.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис архітектури, таблицю метрик, графіки навчання та посилання на збережену модель.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Як обрати архітектуру під тип даних?
2. Які гіперпараметри найбільше впливають на якість?
3. Що таке overfitting і як його виявити?
4. Чому важливо зберігати модель після навчання?
5. Які формати збереження підтримує TensorFlow?
6. Як інтерпретувати криві навчання?
7. Що таке baseline модель?
8. Які ризики виникають при навчанні на малому датасеті?

ЛАБОРАТОРНА РОБОТА №5

ТЕМА: ОЦІНКА ЯКОСТІ ТА ІНТЕРПРЕТАЦІЯ РЕЗУЛЬТАТІВ

МЕТА: Навчитися аналізувати метрики та виявляти типові помилки моделі.

ЗНАТИ: Метрики класифікації, confusion matrix, ROC/PR криві.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- Scikit-learn: Classification Metrics https://scikit-learn.org/stable/modules/model_evaluation.html#classification-metrics
- TensorFlow: Model Evaluation https://www.tensorflow.org/api_docs/python/tf/keras/metrics

Додаткова документація:

- https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html
- https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується у Python/Jupyter.

ЗАВДАННЯ

Необхідно оцінити якість моделі та інтерпретувати результати.

Підготовка до виконання

1. Завантажити навчену модель з попередньої роботи.
2. Підготувати тестовий набір.

Оцінка та аналіз

3. Обчислити accuracy, precision, recall, F1.
4. Побудувати confusion matrix.
5. Побудувати ROC та PR криві.
6. Зробити аналіз типових помилок.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Таблиці метрик, графіки, опис помилок та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. У чому різниця між precision і recall?
2. Навіщо потрібна confusion matrix?
3. Коли ROC може бути оманливою?
4. Які метрики чутливі до дисбалансу класів?
5. Як інтерпретувати PR криву?
6. Які типові причини помилок класифікації?
7. Як обрати поріг для класифікатора?
8. Чим відрізняється accuracy від F1?

ЛАБОРАТОРНА РОБОТА №6

ТЕМА: ОПТИМІЗАЦІЯ МОДЕЛІ ДЛЯ EDGE (КВАНТИЗАЦІЯ)

МЕТА: Зменшити розмір моделі та оцінити вплив на точність і швидкість.

ЗНАТИ: Основи квантизації, компроміси точність/продуктивність.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- TensorFlow Lite: Quantization
https://www.tensorflow.org/lite/performance/post_training_quantization
- TensorFlow Lite: Model Optimization
https://www.tensorflow.org/lite/performance/model_optimization

Додаткова документація:

- <https://www.tensorflow.org/lite/guide>
- <https://github.com/tensorflow/model-optimization>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується у Python/Jupyter.

ЗАВДАННЯ

Необхідно виконати post-training quantization і порівняти результати.

Підготовка до виконання

1. Завантажити навчену модель та підготовлені дані.

Квантизація

2. Застосувати динамічну або повну квантизацію.

3. Зберегти TFLite модель.

Порівняння

4. Порівняти точність оригінальної та квантизованої моделей.

5. Оцінити розмір і швидкість виконання (інференс).

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

– Завдання л/роботи.

– Порівняльну таблицю розміру, точності, часу інференсу та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке квантизація і навіщо вона потрібна?

2. Які типи квантизації існують?

3. Як квантизація впливає на точність?

4. Чому важливо порівнювати розмір моделі?

5. Які типові ризики при квантизації?

6. Що таке TFLite формат?

7. Як перевірити швидкодію на Edge-пристрої?

8. У яких випадках квантизація недоречна?

ЛАБОРАТОРНА РОБОТА №7

ТЕМА: РОЗГОРТАННЯ МОДЕЛІ НА RASPBERRY PI 4

МЕТА: Запустити inference на Edge-пристрої та оцінити продуктивність.

ЗНАТИ: Основи розгортання TFLite моделей, роботу з Raspberry Pi, вимірювання FPS та latency.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- TensorFlow Lite на Raspberry Pi <https://www.tensorflow.org/lite/guide/python>
- Raspberry Pi Documentation <https://www.raspberrypi.com/documentation/>

Додаткова документація:

- <https://www.tensorflow.org/lite/examples>
- <https://www.tensorflow.org/lite/performance/benchmark>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується на Raspberry Pi 4 або через віддалений доступ до нього.

ЗАВДАННЯ

Необхідно запустити TFLite модель на Raspberry Pi та виміряти продуктивність.

Підготовка до виконання

1. Перенести модель на Raspberry Pi.
2. Налаштувати Python-середовище та залежності.

Розгортання

3. Реалізувати скрипт інференсу для статичних даних або з камери.

4. Виміряти latency та FPS.

Оцінка результатів

5. Порівняти швидкість з локальним запуском (за можливості).

6. Зробити висновки щодо продуктивності.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

– Завдання л/роботи.

– Опис середовища, результати вимірювань та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які фактори впливають на latency на Edge-пристроях?
2. Чим відрізняється FPS від throughput?
3. Як оптимізувати inference на Raspberry Pi?
4. Які обмеження має Pi 4 для ML задач?
5. Навіщо вимірювати продуктивність на реальному пристрої?
6. Які інструменти використовують для бенчмаркінгу?
7. Як організувати пайплайн для відео-інференсу?
8. Які ризики при розгортанні моделі на Edge?

ЛАБОРАТОРНА РОБОТА №8

ТЕМА: РОЗГОРТАННЯ МОДЕЛІ НА ESP32 (TFLITE MICRO)

МЕТА: Запустити спрощену модель на MCU та перевірити базовий inference.

ЗНАТИ: Основи TFLite Micro, обмеження пам'яті MCU, процес інтеграції в PlatformIO.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- TensorFlow Lite Micro <https://www.tensorflow.org/lite/microcontrollers>
- ESP32 PlatformIO
<https://docs.platformio.org/en/latest/boards/espressif32/esp32dev.html>

Додаткова документація:

- <https://github.com/tensorflow/tflite-micro>
- <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується на ESP32 з використанням PlatformIO у VSCode.

ЗАВДАННЯ

Необхідно інтегрувати TFLite Micro модель у прошивку та виконати тестовий inference.

Підготовка до виконання

1. Перетворити модель у формат TFLite Micro (квантизація за потреби).
2. Створити проєкт PlatformIO для ESP32.

Інтеграція

3. Додати модель до прошивки (C array).
4. Реалізувати тестовий inference на MCU.

Перевірка

5. Зняти лог результатів та оцінити час виконання.
6. Зробити висновки щодо обмежень MCU.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис процесу інтеграції, результати inference та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які обмеження має ESP32 для ML задач?
2. Чим відрізняється TFLite Micro від стандартного TFLite?
3. Як інтегрувати модель у прошивку?
4. Які дані доцільно використовувати для MCU inference?
5. Як оцінити споживання пам'яті моделі?
6. Які типові проблеми виникають при розгортанні на MCU?
7. Навіщо потрібна квантизація для MCU?
8. Як перевіряти коректність результатів inference?

ЛАБОРАТОРНА РОБОТА №9

ТЕМА: ІНФЕРЕНС КОМП'ЮТЕРНОГО ЗОРУ НА RASPBERRY PI 4 З КАМЕРОЮ

МЕТА: Побудувати end-to-end pipeline для задачі комп'ютерного зору на Edge.

ЗНАТИ: Базові етапи CV pipeline, роботу з камерою, вимірювання затримок.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- OpenCV Python Tutorials
https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html
- TensorFlow Lite: Image Classification
https://www.tensorflow.org/lite/examples/image_classification/overview

Додаткова документація:

- <https://www.raspberrypi.com/documentation/accessories/camera.html>
- https://docs.opencv.org/4.x/d8/dfe/classcv_1_1VideoCapture.html

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується на Raspberry Pi 4 з камерою або через віддалений доступ до нього.

ЗАВДАННЯ

Необхідно реалізувати pipeline: захоплення кадру -> передобробка -> inference -> вивід результатів.

Підготовка до виконання

1. Налаштувати камеру та перевірити захоплення кадрів.
2. Підготувати модель для класифікації або детекції.

Pipeline

3. Реалізувати передобробку (resize, normalization).
4. Запустити інференс на кожному кадрі.
5. Відобразити результат (label, confidence) та заміряти FPS.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис pipeline, метрики продуктивності, приклади результатів і висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які основні етапи CV pipeline?
2. Як розмір зображення впливає на продуктивність?
3. Які метрики використовують для оцінки CV моделей?
4. Як зменшити затримку в реальному часі?
5. Які обмеження має використання камери на Pi?
6. Чим відрізняється класифікація від детекції?
7. Як організувати буферизацію кадрів?
8. Які ризики data leakage при роботі з відео?

ЛАБОРАТОРНА РОБОТА №10

ТЕМА: РЕАЛЬНИЙ ЧАС НА ESP32: СЕНСОРНІ ДАНІ ТА КЛАСИФІКАЦІЯ

МЕТА: Реалізувати *near-real-time inference* на MCU з потоком даних.

ЗНАТИ: Буферизацію сигналів, циклічний *inference*, оцінку затримок.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- ESP32 Data Streaming
<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/adc.html>
- TensorFlow Lite Micro Examples
<https://github.com/tensorflow/tflite-micro/tree/main/tensorflow/lite/micro/examples>

Додаткова документація:

- <https://platformio.org/>
- <https://www.tensorflow.org/lite/microcontrollers>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується на ESP32 з використанням PlatformIO.

ЗАВДАННЯ

Необхідно організувати потік сенсорних даних та виконувати *inference* у циклі.

Підготовка до виконання

1. Підключити сенсор (або використовувати псевдодані).
2. Налаштувати частоту збору даних.

Реальний час

3. Реалізувати буферизацію у вікні фіксованого розміру.
4. Виконувати inference на кожному вікні.
5. Виміряти затримку та частоту оновлення результату.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис потоку даних, параметри буфера, вимірювання затримок та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке near-real-time inference?
2. Як розмір буфера впливає на затримку?
3. Які обмеження пам'яті критичні для MCU?
4. Як стабілізувати частоту зчитування даних?
5. Які типові проблеми при потоковій обробці?
6. Як виміряти затримку inference?
7. Як оптимізувати виконання на ESP32?
8. Які ризики втрати даних у потоці?

ЛАБОРАТОРНА РОБОТА №11

ТЕМА: МУЛЬТИСЕНСОРНА ІНТЕГРАЦІЯ: КАМЕРА + ДОДАТКОВІ СИГНАЛИ

МЕТА: Підвищити точність через злиття даних різних модальностей.

ЗНАТИ: Стратегії data fusion, синхронізацію потоків, базові multimodal підходи.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- Multimodal Learning Overview <https://arxiv.org/abs/1705.09406>
- TensorFlow: Functional API https://keras.io/guides/functional_api/

Додаткова документація:

- <https://pandas.pydata.org/docs/>
- <https://numpy.org/doc/>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується у Python/Jupyter, за наявності кількох джерел даних.

ЗАВДАННЯ

Необхідно синхронізувати дані з двох джерел та побудувати просту multimodal модель.

Підготовка до виконання

1. Зібрати або використати готові дані з двох модальностей.
2. Визначити спосіб синхронізації (timestamp або індексація).

Інтеграція

3. Побудувати ознаки для кожної модальності.
4. Об'єднати ознаки та навчити просту модель (MLP).

Оцінка

5. Порівняти якість unimodal та multimodal моделей.
6. Зробити висновки щодо ефективності злиття.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис синхронізації, архітектуру моделі, метрики та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке multimodal learning?
2. Як синхронізувати дані з різних джерел?
3. Які підходи data fusion найпоширеніші?
4. Коли fusion може погіршити результат?
5. Як оцінити вклад кожної модальності?
6. Які ризики невірної синхронізації?
7. Як обрати архітектуру для multimodal задачі?
8. Які типові метрики використовують для оцінки?

ЛАБОРАТОРНА РОБОТА №12

ТЕМА: ОПТИМІЗАЦІЯ ПРОДУКТИВНОСТІ НА RASPBERRY PI 4

МЕТА: Досягти заданих FPS/latency через профілювання та оптимізацію пайплайну.

ЗНАТИ: Основи профілювання, оптимізації I/O, batch-обробки, кешування.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- TensorFlow Lite Performance <https://www.tensorflow.org/lite/performance>
- Python Profiling <https://docs.python.org/3/library/profile.html>

Додаткова документація:

- <https://www.raspberrypi.com/documentation/>
- <https://www.tensorflow.org/lite/performance/benchmark>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується на Raspberry Pi 4.

ЗАВДАННЯ

Необхідно профілювати пайплайн та покращити продуктивність.

Підготовка до виконання

1. Визначити базовий сценарій інференсу (кадри або статичні дані).
2. Зафіксувати початкові метрики FPS та latency.

Оптимізація

3. Застосувати одну або кілька оптимізацій (batching, зменшення розміру, кешування).
4. Повторно виміряти метрики.

Порівняння

5. Порівняти результати до/після.
6. Описати причини покращення або деградації.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Таблицю метрик до/після, опис оптимізацій та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які основні джерела затримки в Edge-пайплайні?
2. Як профілювати Python код на Raspberry Pi?
3. Чому batching може допомогти або зашкодити?
4. Як зменшення розміру вхідних даних впливає на точність?
5. Які оптимізації дають найбільший ефект?
6. Як оцінювати стабільність FPS?
7. Які ризики надмірної оптимізації?
8. Як зафіксувати відтворюваність вимірювань?

ЛАБОРАТОРНА РОБОТА №13

ТЕМА: ІНТЕГРАЦІЯ ML В КЕРОВАНУ МАШИНКУ НА RASPBERRY PI 4

МЕТА: Реалізувати автономний сценарій керування на основі inference.

ЗНАТИ: Основи роботи з GPIO, управління приводами, логування результатів.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- Raspberry Pi GPIO
<https://www.raspberrypi.com/documentation/computers/os.html#gpio>
- TensorFlow Lite: Python API <https://www.tensorflow.org/lite/guide/python>

Додаткова документація:

- <https://gpiozero.readthedocs.io/en/stable/>
- <https://www.raspberrypi.com/documentation/>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується на Raspberry Pi 4 з підключеними приводами/датчиками (за наявності) або у симуляційному режимі.

ЗАВДАННЯ

Необхідно реалізувати цикл: отримання даних -> inference -> керування.

Підготовка до виконання

1. Описати сценарій керування (лінійне слідування, уникнення перешкод тощо).
2. Підготувати модель для inference.

Інтеграція

3. Реалізувати зв'язок inference з керуванням приводами.
4. Додати логування рішень моделі.

Оцінка

5. Провести тестування у декількох сценаріях.
6. Зробити висновки щодо стабільності керування.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис сценарію, результати тестів, логи та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Як організувати безпечне керування приводами?
2. Які вимоги до затримки у керованих системах?
3. Як логування допомагає у налагодженні?
4. Які типові помилки при інтеграції ML в керування?
5. Як оцінити стабільність роботи системи?
6. Які ризики виникають при некоректних prediction?
7. Як тестувати керовану систему без ризику пошкоджень?
8. Які підходи до fail-safe логіки?

ЛАБОРАТОРНА РОБОТА №14

ТЕМА: НАДІЙНІСТЬ І ТЕСТУВАННЯ EDGE AI

МЕТА: Перевірити стійкість моделі до шуму та збоїв.

ЗНАТИ: Підходи до fault injection, сценарне тестування, базові метрики надійності.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- Robustness in ML (overview) <https://arxiv.org/abs/1902.02751>
- TensorFlow Model Evaluation <https://www.tensorflow.org/guide/evaluate>

Додаткова документація:

- https://scikit-learn.org/stable/modules/model_evaluation.html
- <https://numpy.org/doc/>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується у Python/Jupyter або на Edge-пристрої.

ЗАВДАННЯ

Необхідно оцінити стійкість моделі до шуму та часткових збоїв.

Підготовка до виконання

1. Обрати модель та тестовий набір.
2. Визначити типи шуму або збоїв.

Тестування

3. Додати шум до даних та повторити inference.
4. Провести fault injection (наприклад, пропуски даних).

5. Порівняти метрики з базовими результатами.

Оцінка

6. Визначити межі стійкості.

7. Запропонувати способи підвищення надійності.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

– Завдання л/роботи.

– Опис сценаріїв тестування, таблиці метрик та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Що таке fault injection?

2. Які типи шуму найчастіше використовують у тестах?

3. Як оцінити стійкість моделі?

4. Які метрики чутливі до деградації якості?

5. Як відрізнити системний збій від випадкового шуму?

6. Які підходи підвищують надійність моделей?

7. Чому важливо тестувати на реальному пристрої?

8. Як документувати результати тестів?

ЛАБОРАТОРНА РОБОТА №15

ТЕМА: ОТА-ООНОВЛЕННЯ МОДЕЛЕЙ

МЕТА: Налаштувати цикл оновлення та перевірити rollback.

ЗНАТИ: Основи ОТА, контроль версій, сумісність артефактів.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- ESP32 OTA Updates
<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/system/ota.html>
- MLOps Model Versioning <https://mlflow.org/docs/latest/model-registry.html>

Додаткова документація:

- <https://docs.platformio.org/en/latest/platforms/espressif32.html>
- <https://www.tensorflow.org/lite/guide>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується на ESP32 або Raspberry Pi (за можливості), з використанням Python для керування артефактами.

ЗАВДАННЯ

Необхідно виконати оновлення моделі та перевірити коректний rollback.

Підготовка до виконання

1. Підготувати дві версії моделі (v1, v2).
2. Визначити критерії сумісності.

Оновлення

3. Реалізувати механізм оновлення (OTA або ручне завантаження).
4. Перевірити роботу моделі після оновлення.

Rollback

5. Імітувати збій та повернутися до v1.
6. Задokumentувати результати.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Опис процесу оновлення, результати тестів і висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Чому OTA важливе для Edge AI?
2. Які ризики оновлення моделей у виробництві?
3. Як перевірити сумісність нової моделі?
4. Що таке rollback і як його реалізувати?
5. Які метрики потрібно контролювати після оновлення?
6. Як організувати версіонування моделей?
7. Які типові помилки в OTA процесі?
8. Як документувати оновлення для аудиту?

ЛАБОРАТОРНА РОБОТА №16

ТЕМА: ПІДСУМКОВИЙ МІНІПРОЄКТ

МЕТА: Інтегрувати повний Edge AI цикл у межах одного проєкту.

ЗНАТИ: Повний цикл Edge AI: збір даних, навчання, оптимізація, розгортання.

ТЕОРЕТИЧНІ ВІДОМОСТІ

У зв'язку з великим обсягом інформації використовувати електронну документацію (погоджувати з лектором).

Основна документація:

- Edge AI Overview <https://www.tensorflow.org/lite>
- ML Project Checklist
<https://developers.google.com/machine-learning/guides/rules-of-ml>

Додаткова документація:

- <https://jupyter.org/documentation>
- <https://platformio.org/>

УМОВИ ВИКОНАННЯ Л/Р

Лабораторна робота виконується з використанням доступних засобів: Python/Jupyter, Raspberry Pi або ESP32, VSCode та Google Colab (за потреби).

ЗАВДАННЯ

Необхідно реалізувати повний Edge AI цикл для обраної задачі.

Підготовка до виконання

1. Обрати прикладну задачу (CV, аудіо, сенсори).
2. Скласти короткий план експерименту.

Реалізація

3. Зібрати або підготувати дані.
4. Навчити модель та оцінити якість.
5. Оптимізувати модель для Edge.
6. Розгорнути модель на пристрої.

Демонстрація

7. Показати роботу системи у реальному сценарії.
8. Підготувати коротку презентацію результатів.

ОФОРМЛЕННЯ ЗВІТУ

У звіт лабораторної роботи помістити:

- Завдання л/роботи.
- Повний опис циклу, метрики, артефакти, результати демонстрації та висновки.

Лабораторну роботу оформляти за загальноприйнятою формою (ДСТУ) та аналогічно до інших предметів кафедри.

КОНТРОЛЬНІ ЗАПИТАННЯ

1. Які етапи включає повний цикл Edge AI?
2. Як обрати задачу під доступні ресурси?
3. Як оцінити успішність проєкту?
4. Які типові ризики під час інтеграції?
5. Як забезпечити відтворюваність результатів?
6. Які артефакти слід зберігати після проєкту?
7. Як обґрунтувати вибір оптимізацій?
8. Які критерії демонстрації результатів?

Контроль знань

Експрес-опитування, письмові контрольні завдання, поточний контроль у

Види контролю: поточний, підсумковий.

Методи контролю: спостереження за освітньою діяльністю здобувачів, усне опитування.

Форма підсумкового контролю: екзамен.

Контроль знань і вмінь здобувачів (поточний і підсумковий) здійснюється згідно з кредитною трансферно-накопичувальною системою організації освітнього процесу. Рейтинг здобувача вищої освіти визначається за 100 бальною шкалою: складається з рейтингу із поточної академічної активності впродовж семестру, для оцінювання якої призначається 100 балів (1й семестр по 50 балів за кожен змістовний модуль, 2й семестр по 30 балів за кожен змістовний модуль та 40 балів за екзамен - оцінка здобувача вищої освіти).

Оцінювання здійснюється відповідно до вимог Положення про організацію освітнього процесу в ЦНТУ (розділ 8, стор. 28-30, 32-33; апеляція – стор. 34-35) за такими **критеріями оцінювання академічних досягнень** здобувачів вищої освіти:

- Вчасність,
- ініціативна дослідна робота в межах СРС,
- активність, ініціативність на заняттях.

При оцінюванні знань студентів використовуються такі засоби контролю: усне опитування перед допуском до виконання лабораторної роботи – здійснюється на їх початку; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної роботи згідно з робочою програмою дисципліни.

Оцінка, яка виставляється за лабораторне заняття, складається з таких елементів: усне опитування студентів перед допуском до виконання лабораторної роботи; знання теоретичного матеріалу з теми; якість оформлення звіту; вміння

студента обґрунтувати прийняті конструктивні рішення; покращувати результат після зворотного зв'язку; своєчасний захист лабораторної роботи. Для виконання програми дисципліни студент повинен отримати оцінки за всі лабораторні роботи.

Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на наступному після виконання роботи занятті. За несвоєчасний захист лабораторної роботи з неповажної причини студент за позитивну відповідь отримує оцінку «задовільно».

Пропущене лабораторне заняття студент повинен відпрацювати в лабораторіях кафедри у встановлений викладачем термін, але не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Відповідність шкали оцінювання ЄКТС національній системі оцінювання визначена у розділі 8 Положення про організацію освітнього процесу в ЦНТУ: таблиця 1, стор. 33 www.kntu.kr.ua/file/content/424/polozhennia-pro-orhanizatsiyu-osvitnoho-protsesu-v-tsntu.pdf

Розподіл балів, які отримують студенти

1-й семестр

Поточне тестування та самостійна робота							Сума балів за рубіжні контролі	Підсумковий контроль (залік)
I рубіж				II рубіж				
T1	T2	T3	T4	T5	T6	T7	100	100
12	12	12	14	16	16	18		
50				50				

2-й семестр

Поточне тестування та самостійна робота							Сума балів за рубіжні контролі	Підсумковий контроль (екзамен)
I рубіж				II рубіж				
T1	T2	T3	T4	T5	T6	T7	60	100
7	7	8	8	10	10	10		
30				30				

Шкала оцінювання знань

Шкала оцінювання: національна та ЄКТС

Сума балів за всіма видами навчальної діяльності	Оцінка ЄКТС	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90-100	A	відмінно	зараховано
82-89	B	добре	
74-81	C		
64-73	D	задовільно	
60-63	E		
35-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

Критерії оцінки іспиту:

оцінку «відмінно» (90-100 балів, А) заслуговує студент, який:

- всебічно, систематично і глибоко володіє навчально-програмовим матеріалом;
- вміє самостійно виконувати завдання, передбачені програмою, використовує набуті знання і вміння у нестандартних ситуаціях;
- засвоїв основну і ознайомлений з додатковою літературою, яка рекомендована програмою;
- засвоїв взаємозв'язок основних понять дисципліни та усвідомлює їх значення для професії, яку він набуває;
- вільно висловлює власні думки, самостійно оцінює різноманітні життєві явища і факти, виявляючи особистісну позицію;
- самостійно визначає окремі цілі власної навчальної діяльності, виявив творчі здібності і використовує їх при вивченні навчально-програмового матеріалу, проявив нахил до наукової роботи.

оцінку «добре» (82-89 балів, В) – заслуговує студент, який:

- повністю опанував і вільно (самостійно) володіє навчально-програмовим матеріалом, в тому числі застосовує його на практиці, має системні знання достатньому обсязі відповідно до навчально-програмового матеріалу, аргументовано використовує їх у різних ситуаціях;

- має здатність до самостійного пошуку інформації, а також до аналізу, постановки і розв'язування проблем професійного спрямування;

- під час відповіді допустив деякі неточності, які самостійно виправляє, добирає переконливі аргументи на підтвердження вивченого матеріалу;

оцінку «добре» (74-81 бал, С) заслуговує студент, який:

- в загальному роботу виконав, але відповідає на екзамені з певною кількістю помилок;

- вміє порівнювати, узагальнювати, систематизувати інформацію під керівництвом викладача, в цілому самостійно застосовувати на практиці, контролювати власну діяльність;

- опанував навчально-програмовий матеріал, успішно виконав завдання, передбачені програмою, засвоїв основну літературу, яка рекомендована програмою;

оцінку «задовільно» (64-73 бали, D) – заслуговує студент, який:

- знає основний навчально-програмовий матеріал в обсязі, необхідному для подальшого навчання і використання його у майбутній професії;

- виконує завдання, але при рішенні допускає значну кількість помилок;

- ознайомлений з основною літературою, яка рекомендована програмою;

- допускає на заняттях чи екзамені помилки при виконанні завдань, але під керівництвом викладача знаходить шляхи їх усунення.

оцінку «задовільно» (60-63 бали, E) – заслуговує студент, який:

- володіє основним навчально-програмовим матеріалом в обсязі, необхідному для подальшого навчання і використання його у майбутній професії, а виконання завдань задовольняє мінімальні критерії. Знання мають репродуктивний характер.

оцінка «незадовільно» (35-59 балів, FX) – виставляється студенту, який:
виявив суттєві прогалини в знаннях основного програмового матеріалу,
допустив принципові помилки у виконанні передбачених програмою завдань.

оцінку «незадовільно» (35 балів, F) – виставляється студенту, який:

- володіє навчальним матеріалом тільки на рівні елементарного розпізнавання і відтворення окремих фактів або не володіє зовсім;
- допускає грубі помилки при виконанні завдань, передбачених програмою;
- не може продовжувати навчання і не готовий до професійної діяльності після закінчення університету без повторного вивчення даної дисципліни.

При виставленні оцінки враховуються результати навчальної роботи студента протягом семестру

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

Базова

1. Weidman S. Deep Learning from Scratch: Building with Python from First Principles. – O'Reilly. – 252 p.
2. Luger, George F. Artificial Intelligence: Principles and Practice. Springer, 2024. ISBN 3031574362.
3. Hurbans R. Grokking Artificial Intelligence Algorithms. – Manning, 2020. – 631 p.
4. Teofili T. Deep Learning for Search. – Manning, 2019. – 695 p.
5. Kotu V., Deshpande B. Data Science: Concepts and Practice. – Elsevier Science, 2018. – 953 p.
6. Читальний зал № 1 (ЦНТУ) Фісун М. Т., Кравець І. О., Казмірчук П. П., Ніколенко С. Г.. Інтелектуальний аналіз даних: практикум : навч. посіб. Львів : Новий Світ 2000, 2023. 161 с. ISBN 978-966-418-287-7.
7. Читальний зал № 1 (ЦНТУ) Висоцька В. А., Досин Д. Г., Микіч Х. І. та ін. Методи та засоби функціонування систем підтримки прийняття рішень на основі онтологій : монографія. Львів : Новий Світ 2000, 2019. 236 с. ISBN 978-617-7519-47-7.
8. Читальний зал № 1 (ЦНТУ) Нікольський Ю. В., Пасічник В. В., Щербина Ю. М.. Системи штучного інтелекту : навч. посіб. 2-ге вид., випр. та доп. Львів : Магнолія, 279 с. ISBN 978-617-57-40-11-4.

Допоміжна

9. Кавун С.В., Смірнов О.А., Сорбат І.В., Мелешко Є.В., Коваленко О.В. Системи штучного інтелекту // Навчальний посібник – Кіровоград: КНТУ 2013. – 335 с.

10. Інженерія програмного забезпечення: Навчальний посібник / О.А. Смірнов, О.В. Коваленко, Є.В. Мелешко, Л.В. Константинова, А.С. Кожанова // Кіровоград: КНТУ 2013. – 335 с.
11. Fenner M. Machine Learning with Python for Everyone (Addison-Wesley Data & Analytics Series) 1st Edition, Kindle Edition. - Addison-Wesley Professional, 2019. – 586 p.
12. Mollick, Ethan. Co-Intelligence: Living and Working with AI. Portfolio, 2024. ISBN 059371671X.
13. Harari, Yuval Noah. Nexus: A Brief History of Information Networks from the Stone Age to AI. Random House Publishing Group, 2024. ISBN 059373422X.
14. Hao, Karen. Empire of AI: Dreams and Nightmares in Sam Altman's OpenAI. Penguin Press, 2025. ISBN 0593657500.
15. Lennox, John C. 2084 and the AI Revolution, Updated and Expanded Edition: How Artificial Intelligence Informs Our Future. Zondervan, 2024. ISBN 0310166640.
16. Poole, David L., Mackworth, Alan K. Artificial Intelligence: Foundations of Computational Agents. 3rd ed. Cambridge University Press, 2023. ISBN 1009258192.
17. Dagur, Arvind, Agarwal, Sohit, Shukla, Dharendra Kumar, Ali, Shabir, Sharma, Sandhya (eds.). Artificial Intelligence and Sustainable Innovation. Volume 1. CRC Press, 2026. ISBN 1041101996.
18. Martinez, David R., Kifle, Bruke M. Artificial Intelligence: A Systems Approach from Architecture Principles to Deployment. The MIT Press, 2024. ISBN 0262048981.
19. Eitel Porter, Ray, Dongha, Paul, Vogel, Miriam. Governing the Machine: How to Navigate the Risks of AI and Unlock Its True Potential. Bloomsbury Business, 2025. ISBN 139942629X.
20. Lavista Ferres, Juan M. Degrees of Change: What AI Means for Education and the Next Generation. Wiley, 2026. ISBN 1394413068.
21. Graley, J. F. Understanding Artificial Intelligence Volume One: A Beginners Guide to the World of Artificial Intelligence. Books of Understand, 2024. ASIN B0DHGRWY3C.

22. Tsoukalas, Lefteri H. Fuzzy Logic: Applications in Artificial Intelligence, Big Data, and Machine Learning. McGraw Hill, 2023. ISBN 1264675917.
23. Iansiti, Marco, Lakhani, Karim R. Competing in the Age of AI: Strategy and Leadership When Algorithms and Networks Run the World. Harvard Business Review Press, 2020. ISBN 1633697622.
24. Wilkins, Neil. Artificial Intelligence: What You Need to Know about Machine Learning, Robotics, Deep Learning, Recommender Systems, Internet of Things, Neural Networks, Reinforcement Learning, and Our Future. Bravex Publications, 2019. ISBN 1647481694.
25. Smith, Robert Elliott. Rage Inside the Machine: The Prejudice of Algorithms, and How to Stop the Internet Making Bigots of Us All. Bloomsbury Business, 2019. ISBN 1472963881.
26. Theodoridis, Sergios, Koutroumbas, Konstantinos. Pattern Recognition. Academic Press, 2008. ISBN 1597492728.
27. Chen, Stella. Artificial Intelligence: Navigating the Depths and Heights of Artificial Intelligence, its Origins, Innovations, Ethical Implications, and Boundless Potential for Tomorrow's World. Kindle edition, 2024. ASIN B0CSTRD7YJ.

Наукові публікації

28. Ulichev O., Meleshko Ye., Smirnov O., Khokh V., Goncharenko Iu. Method of Choosing Objects for Informational Influence in Social Networks during Information Campaign Based on the Analytic Hierarchy Process // CEUR-WS, Vol 2588, Lviv, Ukraine. – 2019. – P. 215-227 (ISSN 16130073) (SCOPUS) – URL: <https://www.scopus.com/record/display.uri?eid=2-s2.0-85083203878&origin=resultslist>
29. Mohammed A.S., Meleshko, Y., Balaji S.B., Semenov S. Collaborative Filtering Method with the use of Production Rules Proceedings of ICCIKE, Amity University DubaiDubai; United Arab Emirates. –2019. – c. 387-391 (SCOPUS) (Web of Science)

URL:<https://www.scopus.com/record/display.uri?eid=2-s2.0-85080933551&origin=resultslist>

30. Meleshko Ye., Drieiev O., Drieieva H. Method of identification bot profiles based on neural networks in recommendation systems // Advanced Information Systems. – 2020. – Vol. 4, No. 2 – С. 24-28. **(фахове видання)** – URL: <http://ais.khpi.edu.ua/article/view/2522-9052.2020.2.05> **(Google Scholar)**

31. Meleshko Ye. Method of generating recommendations lists with considering activity indexes of users in a recommendation system // Науковий журнал Сучасні інформаційні системи. – Харків: НТУ "ХПИ", 2019. – Т. 3, № 1. – С. 43-47. **(фахове видання)** – URL: <http://ais.khpi.edu.ua/article/view/2522-9052.2019.1.07>

32. Meleshko Ye. Method of collaborative filtration based on associative networks of users similarity // Науковий журнал Сучасні інформаційні системи. – Харків: НТУ "ХПИ", 2018. - Т. 2, № 4. - С. 55-59. **(фахове видання)** – URL: <http://ais.khpi.edu.ua/article/view/2522-9052.2018.4.09>

33. Meleshko Ye., Yakymenko M., Semenov S. A Method of Detecting Bot Networks Based on Graph Clustering in the Recommendation System of Social Network // CEUR-WS, Vol 2870, Lviv, Ukraine. – 2021. – P. 1249-1261 (ISSN 16130073) **(SCOPUS)** **(Web of Science)** – URL: <https://www.scopus.com/record/display.uri?eid=2-s2.0-85107217511&origin=resultslist>

34. Мелешко Є.В. Методи кластеризації графів соціальних мереж для побудови рекомендаційних систем // Збірник наукових праць Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2019. – Т. 2 (54). – С. 129-134. **(фахове видання)** – URL: <http://journals.nupp.edu.ua/sunz/article/view/1421>

Інформаційні ресурси

35. Курс «Штучний інтелект» на сервері дистанційної освіти ЦНТУ. – URL: <https://moodle.kntu.kr.ua/course/view.php?id=1084>

36. <https://prometheus.org.ua/> – українська платформа безкоштовних онлайн-курсів
37. <https://www.tensorflow.org/> – An end-to-end open source machine learning platform
38. <http://leenissen.dk/> – Fast Artificial Neural Network Library
39. <https://www.codeproject.com/> – колективний блог з новинами та навчальними статтями про інформаційні технології та програмування.
40. <http://stackoverflow.com/> – система питань і відповідей для професійних програмістів та новачків у програмуванні.
41. <https://dou.ua/> – український веб-сайт з елементами колективного блогу, створений для розповсюдження новин, аналітичних статей та свіжої інформації пов'язаної із інформаційними технологіями.
42. <http://www.algomation.com/> – це платформа для перегляду, обміну і створення візуалізацій алгоритмів.